

# संगणक ऑपरेटर आणि प्रोग्रामिंग असिस्टंट COMPUTER OPERATOR AND PROGRAMMING ASSISTANT

NSQF स्तर - 3

## ट्रेड थिअरी TRADE THEORY

क्षेत्र : आयटी आणि आयटीईएस

SECTOR : IT & ITES

(संशोधित अभ्यास क्रमानुसार जुलै 2022 - 1200 तास)  
(As per revised syllabus July 2022 - 1200 hrs)



Directorate General of Training

डायरेक्टोरेट जनरल ऑफ ट्रेनिंग  
कौशल्य विकास आणि उद्यमशीलता मंत्रालय  
भारत सरकार



नॅशनल इंस्ट्रक्शनल  
मीडिया इन्स्टिट्यूट, चेन्नई

पोस्ट बॉक्स क्र. 3142, CTA कॅम्पस, गिंडी, चेन्नई - 600 032

क्षेत्र : आयटी आणि आयटीईएस  
कालावधी : 1 वर्ष  
ट्रेड : संगणक ऑपरेटर आणि प्रोग्रामिंग असिस्टंट - ट्रेड थिअरी - NSQF स्तर - 3 (संशोधित 2022)

द्वारे विकसित आणि प्रकाशित



नॅशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट

पोस्ट बॉक्स क्र. 3142, CTA कॅम्पस,

गिंडी, चेन्नई - 600 032

भारत

ईमेल : [chennai-nimi@nic.in](mailto:chennai-nimi@nic.in)

संकेतस्थळ : [www.nimi.gov.in](http://www.nimi.gov.in)

कॉपीराइट © 2022 नॅशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट, चेन्नई

पहिली आवृत्ती : मार्च, 2023

प्रती: 1,000

Rs./-

सर्व हक्क राखीव.

या प्रकाशनाचा कोणताही भाग नॅशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट, चेन्नई यांच्या लिखित परवानगीशिवाय फोटोकॉपी, रेकॉर्डिंग किंवा कोणत्याही माहितीचे संचयन आणि पुनर्प्राप्ती प्रणालीसह कोणत्याही स्वरूपात किंवा इलेक्ट्रॉनिक किंवा यांत्रिक पद्धतीने पुनरुत्पादित किंवा प्रसारित केले जाऊ शकत नाही.

## अग्रलेख

राष्ट्रीय कौशल्य विकास धोरणाचा एक भाग म्हणून त्यांना नोकऱ्या सुरक्षित करण्यात मदत करण्यासाठी भारत सरकारने 2020 पर्यंत 30 कोटी लोकांना कौशल्ये प्रदान करण्याचे महत्वाकांक्षी लक्ष्य ठेवले आहे, प्रत्येक चार भारतीयांपैकी एक. विशेषतः कुशल मनुष्यबळ उपलब्ध करून देण्याच्या दृष्टीने औद्योगिक प्रशिक्षण संस्था (ITIs) या प्रक्रियेत महत्त्वाची भूमिका बजावतात. हे लक्षात घेऊन, आणि प्रशिक्षणार्थीना सध्याच्या उद्योगाशी संबंधित कौशल्य प्रशिक्षण देण्यासाठी, ITI अभ्यासक्रम अलीकडेच विविध भागधारकांचा समावेश असलेल्या मॅटॉर कौन्सिलच्या मदतीने अद्ययावत करण्यात आला आहे. उद्योग, उद्योजक, शिक्षणतज्ज्ञ आणि आयटीआयचे प्रतिनिधी.

नॅशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट (NIMI), चेन्नईने आता सुधारित अभ्यासक्रमाला अनुसरून शैक्षणिक साहित्य आणले आहे. आयटी आणि आयटीईएस क्षेत्रातील **संगणक ऑपरेटर आणि प्रोग्रामिंग असिस्टंट - ट्रेड थिअरी** NSQF स्तर - 3 (संशोधित 2022). NSQF स्तर - 3 ट्रेड प्रॅक्टिकल प्रशिक्षणार्थीना आंतरराष्ट्रीय समतुल्य मानक मिळविण्यात मदत करेल जिथे त्यांची कौशल्य प्रवीणता आणि योग्यता जगभरात योग्यरित्या ओळखली जाईल आणि यामुळे पूर्वीच्या शिक्षणाच्या ओळखीची व्याप्ती देखील वाढेल. NSQF स्तर - 3 प्रशिक्षणार्थीना आयुष्यभर शिक्षण आणि कौशल्य विकासाला प्रोत्साहन देण्याची संधी देखील मिळेल. मला शंका नाही की NSQF स्तर - 3 सह ITI चे प्रशिक्षक आणि प्रशिक्षणार्थी, आणि सर्व भागधारकांना या IMPs चा जास्तीत जास्त फायदा होईल आणि NIMI चे प्रयत्न देशातील व्यावसायिक प्रशिक्षणाची गुणवत्ता सुधारण्यासाठी खूप पुढे जाईल.

NIMI चे कार्यकारी संचालक आणि कर्मचारी आणि मीडिया डेव्हलपमेंट कमिटीचे सदस्य हे प्रकाशन प्रकाशित करण्यासाठी त्यांच्या योगदानाबद्दल कौतुक पात्र आहेत.

जय हिंद

अतिरिक्त सचिव/महासंचालक (प्रशिक्षण)  
**कौशल्य विकास आणि उद्यमशीलता मंत्रालय**  
भारत सरकार.

नवी दिल्ली - 110 001

## प्रस्तावना

नॅशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट (NIMI) ची स्थापना 1986 मध्ये चेन्नई येथे तत्कालीन रोजगार आणि प्रशिक्षण महासंचालनालय (D.G.E & T), श्रम आणि रोजगार मंत्रालय, (आता कौशल्य विकास आणि उद्योजकता मंत्रालयाच्या अंतर्गत) भारत सरकार, तांत्रिक सह. सरकारकडून मदत फेडरल रिपब्लिक ऑफ जर्मनीचे. कारागीर आणि शिकाऊ प्रशिक्षण योजनेतर्गत विहित अभ्यासक्रमानुसार (NSQF LEVEL - 3) विविध ट्रेड्ससाठी शैक्षणिक साहित्य विकसित करणे आणि प्रदान करणे हे या संस्थेचे प्रमुख उद्दिष्ट आहे.

भारतातील NCVT/NAC अंतर्गत व्यावसायिक प्रशिक्षणाचे मुख्य उद्दिष्ट लक्षात घेऊन ही शिकवणी सामग्री तयार केली गेली आहे, जी एखाद्या व्यक्तीला नोकरी करण्यासाठी कौशल्यांमध्ये प्रभुत्व मिळवण्यास मदत करणे आहे. निर्देशात्मक साहित्य इन्स्ट्रक्शनल मीडिया पॅकेजेस (IMPs) स्वरूपात तयार केले जाते. IMP मध्ये थिअरी बुक, प्रॅक्टिकल बुक, टेस्ट आणि असाइनमेंट बुक, इन्स्ट्रक्टर गाइड, ऑडिओ व्हिड्युअल एड (वॉल चार्ट आणि पारदर्शकता) आणि इतर सपोर्ट मटेरियल असतात.

ट्रेड प्रॅक्टिकल पुस्तकात प्रशिक्षणार्थींनी कार्यशाळेत पूर्ण करावयाच्या व्यायामांची मालिका असते. हे व्यायाम विहित अभ्यासक्रमातील सर्व कौशल्ये समाविष्ट आहेत याची खात्री करण्यासाठी डिझाइन केलेले आहेत. ट्रेड थिअरी पुस्तक प्रशिक्षणार्थींना नोकरी करण्यास सक्षम करण्यासाठी आवश्यक संबंधित सैद्धांतिक ज्ञान प्रदान करते. चाचणी आणि असाइनमेंट्स प्रशिक्षकाला प्रशिक्षणार्थींच्या कामगिरीच्या मूल्यमापनासाठी असाइनमेंट देण्यास सक्षम करतील. वॉल तक्ते आणि पारदर्शकता अद्वितीय आहेत, कारण ते केवळ प्रशिक्षकाला विषय प्रभावीपणे मांडण्यासाठीच मदत करत नाहीत तर प्रशिक्षणार्थींच्या आकलनाचे मूल्यांकन करण्यासही मदत करतात. प्रशिक्षक मार्गदर्शक प्रशिक्षकाला त्याच्या सूचनांचे वेळापत्रक, कच्च्या मालाची आवश्यकता, दैनंदिन धडे आणि प्रात्यक्षिकांचे नियोजन करण्यास सक्षम करते. कौशल्ये उत्पादनक्षम रीतीने पार पाडण्यासाठी या निर्देशात्मक सामग्रीमधील व्यायामाच्या QR कोडमध्ये निर्देशात्मक व्हिडिओ एम्बेड केले आहेत जेणेकरून व्यायामामध्ये दिलेल्या प्रक्रियात्मक व्यावहारिक पायऱ्यांसह कौशल्य शिक्षण एकत्रित करता येईल. उपदेशात्मक व्हिडिओ व्यावहारिक प्रशिक्षणाच्या दर्जाची गुणवत्ता सुधारतील आणि प्रशिक्षणार्थींना लक्ष केंद्रित करण्यास आणि कौशल्य अखंडपणे पार पाडण्यास प्रवृत्त करतील.

IMPs प्रभावी कार्यसंघ कार्यासाठी विकसित करणे आवश्यक असलेल्या जटिल कौशल्यांशी देखील संबंधित आहे. अभ्यासक्रमात विहित केल्यानुसार संलग्न व्यापारातील महत्त्वाच्या कौशल्य क्षेत्रांचा समावेश करण्याचीही आवश्यक काळजी घेण्यात आली आहे.

संस्थेमध्ये संपूर्ण सूचनात्मक मीडिया पॅकेजची उपलब्धता प्रशिक्षक आणि व्यवस्थापन दोघांनाही प्रभावी प्रशिक्षण देण्यास मदत करते.

IMPs हे NIMI चे कर्मचारी सदस्य आणि सार्वजनिक आणि खाजगी क्षेत्रातील उद्योग, प्रशिक्षण महासंचालनालय (DGT), सरकारी आणि खाजगी ITIs अंतर्गत विविध प्रशिक्षण संस्थांमधून खास काढलेल्या माध्यम विकास समित्यांच्या सदस्यांच्या सामूहिक प्रयत्नांचे परिणाम आहेत.

NIMI विविध राज्य सरकारांचे रोजगार आणि प्रशिक्षण संचालक, सार्वजनिक आणि खाजगी क्षेत्रातील उद्योगांचे प्रशिक्षण विभाग, DGT आणि DGT फील्ड इन्स्टिट्यूटचे अधिकारी, प्रूफ रीडर, वैयक्तिक मीडिया डेव्हलपर आणि त्यांचे मनःपूर्वक आभार व्यक्त करण्यासाठी या संधीचा लाभ घेऊ इच्छित आहे. समन्वयक, परंतु ज्यांच्या सक्रिय समर्थनासाठी NIMI हे साहित्य आणू शकले नसते.

## आभार

नेशनल इंस्ट्रक्शनल मीडिया इन्स्टिट्यूट (NIMI) खालील माध्यम विकासक आणि त्यांच्या प्रायोजक संस्थांनी हे निर्देशात्मक साहित्य आणण्यासाठी दिलेल्या सहकार्य आणि योगदानाबद्दल आभार मानते. (व्यापार सिद्धांत) च्या व्यापारासाठी कप (NSQF स्तर - 3) (संशोधित 2022) अंतर्गत आयटी आणि आयटीईएसआयटीआयसाठी क्षेत्र.

### माध्यम विकास समिती सदस्य

- |                      |  |
|----------------------|--|
| श्रीमती. एम. भानुमती | - सहाय्यक प्रशिक्षण अधिकारी,<br>सरकारी ITI, अंबचूर.    |
| श्रीमती. व्ही. रेवती | - सहाय्यक प्रशिक्षण अधिकारी,<br>सरकारी ITI, कोईम्बतूर. |

### निमी समन्वयक

- |                      |                                    |
|----------------------|------------------------------------|
| श्री.निर्माल्य नाथ   | - उप संचालक,<br>NIMI, चेन्नई - 32. |
| श्री.जी. मायकेल जॉनी | - मॅनेजर,<br>NIMI, चेन्नई - 32.    |
| श्रीमती बी. रेवती    | - JTA (DTP)<br>NIMI, चेन्नई - 32.  |

NIMI डेटा एंट्री, CAD, DTP ऑपरेटर्सचे या निर्देशात्मक साहित्याच्या विकासाच्या प्रक्रियेत उत्कृष्ट आणि समर्पित सेवांसाठी त्यांचे कौतुक नोंदवते.

या निर्देशात्मक साहित्याच्या विकासासाठी योगदान देणाऱ्या इतर सर्व NIMI कर्मचाऱ्यांनी केलेल्या अमूल्य प्रयत्नांची NIMI आभार मानते.

हे निर्देशात्मक साहित्य विकसित करण्यासाठी प्रत्यक्ष किंवा अप्रत्यक्षपणे मदत करणाऱ्या प्रत्येकाचे NIMI आभारी आहे.

# परिचय

ट्रेड प्रॅक्टिकल मॅन्युअल कार्यशाळेत वापरण्याचा हेतू आहे. त्यामध्ये प्रशिक्षणार्थींनी अभ्यासक्रमादरम्यान पूर्ण करायच्या अनेक व्यावहारिक व्यायामांचा समावेश आहेकपव्यायाम करण्यास मदत करण्यासाठी सूचना/माहितीद्वारे व्यापार पूरक आणि समर्थित. हे व्यायाम सर्व कौशल्ये NSQF स्तर - 3 चे पालन करतात याची खात्री करण्यासाठी डिझाइन केलेले आहेत

- |  |   |
|--|---|
| मॉड्यूल 1 - सुरक्षित कार्य पद्धती  | मॉड्यूल 24 - ऑडिओ आणि व्हिडिओ घटक व्यवस्थापित करा   |
| मॉड्यूल 2 - एक डेस्कटॉप पीसी एकत्र करा   | मॉड्यूल 25 - संक्रमण आणि ॲनिमेशन व्यवस्थापित करा  |
| मॉड्यूल 3 - विंडोज ऑपरेटिंग सिस्टम वापरणे  | मॉड्यूल 26 - सहयोग व्यवस्थापित करा  |
| मॉड्यूल 4 - संगणक मूलभूत आणि सॉफ्टवेअर स्थापना                                       | मॉड्यूल 27 - MySQL  |
| मॉड्यूल 5 - डॉस कमांड लाइन इंटरफेस   | मॉड्यूल 28 - वर प्रात्यक्षिक - केरी   |
| मॉड्यूल 6 - उबंटू लिनक्स ऑपरेटिंग सिस्टम स्थापित करा आणि मूलभूत लिनक्स कमांड्स चालवा | मॉड्यूल 29 - वर प्रात्यक्षिक - कार्यावर प्रात्यक्षिक  |
| मॉड्यूल 7 - वर्ड प्रोसेसिंग सॉफ्टवेअर  | मॉड्यूल 30 - संगणक नेटवर्क सेट-अप आणि कॉन्फिगर करा  |
| मॉड्यूल 8 - वापरणे - दस्तऐवज फॉर्मॅट करा   | मॉड्यूल 31 - आम्हाला साथी स्थिर वेब पृष्ठे तयार करा-HTML टॅग  |
| मॉड्यूल 9 - सारण्या आणि सूची व्यवस्थापित करा   | मॉड्यूल 32 - JavaScript HTML पृष्ठांमध्ये JavaScript एम्बेड करा   |
| मॉड्यूल 10 - संदर्भ  | मॉड्यूल 33 - डेटा व्हिज्युअलायझेशन किंवा विश्लेषण एक्सेल वापरून   |
| मॉड्यूल 11 - तयार करा आणि व्यवस्थापित करा - ग्राफिक घटक व्यवस्थापित करा              | मॉड्यूल 34 - उत्पादने आणि सेवा ओळखण्यासाठी ई-कॉमर्स साइट्स ब्राउझ करा                                   |
| मॉड्यूल 12 - दस्तऐवज सहयोग विभाग व्यवस्थापित करा                                     | मॉड्यूल 35 - व्हायरस, स्पायवेअर आणि इतर दुर्भावनापूर्ण कोडपासून माहिती, संगणक आणि नेटवर्कचे संरक्षण करा |
| मॉड्यूल 13 - मेलिंग व्यवस्थापित करा  | मॉड्यूल 36 - क्लाउड कॉम्प्युटिंग  |
| मॉड्यूल 14 - स्प्रेड शीट ऍप्लिकेशन, वर्कशीट्स आणि वर्कबुक व्यवस्थापित करा            | मॉड्यूल 37 - अनुप्रयोग विकसित करा आणि कार्य करा अनुप्रयोग विकास जीवन चक्र                               |
| मॉड्यूल 15 - डेटा सेल आणि रेंज व्यवस्थापित करा                                       | मॉड्यूल 38 - निवडक मॉड्यूल I - कार्यक्रम- मिंग ते 42 Python मध्ये                                       |
| मॉड्यूल 16 - टेबल आणि टेबल डेटा व्यवस्थापित करा                                      | मॉड्यूल 38 - वैकल्पिक मॉड्यूल II - प्रोग्रामिंग ते 42 JAVA मध्ये  |
| मॉड्यूल 17 - सूत्रे वापरून ऑपरेटन्स करा आणि फंक्शन्स                                 |   |
| मॉड्यूल 18 - चार्ट्स व्यवस्थापित करा   |   |
| मॉड्यूल 19 - पिक्चोटेबल्स व्यवस्थापित करा  |   |
| मॉड्यूल 20 - पॉवर पॉइंट प्रेझेंटेशन्स  |   |
| मॉड्यूल 21 - प्रेझेंटेशनचे स्वरूप येथे द   |   |
| मॉड्यूल 22 - टेबल्स आणि बुलेट केलेले मजकूर व्यवस्थापित करा                           |   |
| मॉड्यूल 23 - ग्राफिक घटक व्यवस्थापित करा   |   |

शॉप फ्लोअरमधील कौशल्य प्रशिक्षण हे काही व्यावहारिक

प्रकल्पाभोवती केंद्रित असलेल्या व्यावहारिक व्यायामांच्या मालिकेद्वारे नियोजित आहे. तथापि, अशी काही उदाहरणे आहेत जिथे वैयक्तिक व्यायाम प्रकल्पाचा भाग बनत नाही.

प्रॅक्टिकल मॅन्युअल विकसित करताना प्रत्येक व्यायाम तयार करण्याचा प्रामाणिक प्रयत्न केला गेला जो सरासरीपेक्षा कमी प्रशिक्षणार्थींना देखील समजण्यास आणि पार पाडण्यास सोपा असेल. तथापि, विकास संघाने हे मान्य केले की आणखी सुधारणेला वाव आहे. NIMI, मॅन्युअलमध्ये सुधारणा करण्यासाठी अनुभवी प्रशिक्षण शिक्षकांच्या सूचनांची अपेक्षा करते.

## व्यापार सिद्धांत

ट्रेड थिअरीच्या मॅन्युअलमध्ये अभ्यासक्रमासाठी सैद्धांतिक माहिती असतेकपव्यापार. ट्रेड प्रॅक्टिकलच्या मॅन्युअलमध्ये समाविष्ट असलेल्या व्यावहारिक व्यायामानुसार सामग्री अनुक्रमित केली जाते. शक्य तितक्या प्रमाणात प्रत्येक व्यायामामध्ये समाविष्ट असलेल्या कौशल्यासह सैद्धांतिक पैलूंचा संबंध जोडण्याचा प्रयत्न केला गेला आहे. प्रशिक्षणार्थींना कौशल्ये पार पाडण्यासाठी आकलन क्षमता विकसित करण्यास मदत करण्यासाठी हा सह-संबंध राखला जातो.

ट्रेड प्रॅक्टिकलच्या मॅन्युअलमध्ये समाविष्ट असलेल्या संबंधित व्यायामासह व्यापार सिद्धांत शिकवला आणि शिकला पाहिजे. या मॅन्युअलच्या प्रत्येक शीटमध्ये संबंधित व्यावहारिक व्यायामाचे संकेत दिले आहेत.

## सामग्री

एक्सरसाईस क्र.	धड्याचे शीर्षक	शिकत आहे परिणाम	पृष्ठ क्र.
	<b>मॉड्यूल 1 : सुरक्षित कार्य मेथड्स (Safe Working Practices)</b>		
1.1.01&02	विदूत सुरक्षा (Electrical safety)	1	1
1.1.03&04	सुरक्षा सराव - अग्निशामक (Safety practice - fire extinguishers)		2
	<b>मॉड्यूल 2 : एक डेस्कटॉप पीसी एकत्र (असेम्बल ) करा (Assemble a Desktop PC)</b>		
1.2.05&06	संगणकाचा परिचय (Introduction to computers)	1	7
	<b>मॉड्यूल 3 : विंडोज ऑपरेटिंग सिस्टम वापरणे (Using Windows Operating System)</b>		
1.3.07-13	CPU आर्किटेक्चर आणि मदरबोर्डचा परिचय (Introduction to CPU architecture and motherboard)	1	17
	<b>मॉड्यूल 4 : संगणक बेसिक आणि सॉफ्टवेअर इन्स्टॉलेशन (Computer Basics &amp; Software Installation)</b>		
1.4.14	BIOS सेटिंग्ज आणि त्यांचे बदल पहा (View the BIOS settings and their modifications)		23
1.4.15	विंडोज ऑपरेटिंग सिस्टम इंस्टाल करा (Install Windows operating system)		32
1.4.16	हार्ड डिस्कचे फॉर्मेटिंग करा आणि पार्टिशन तयार करा (Format hard disk and create partition)	1	31
1.4.17	OS इंस्टॉलेशन दरम्यान सामान्य हार्डवेअर आणि सॉफ्टवेअर समस्या ओळखा आणि दुरुस्त करा (Identify and rectify common hardware and software issues during OS installation)		39
1.4.18	विंडोजसाठी आवश्यक ॲप्लिकेशन सॉफ्टवेअर इन्स्टॉल करा उदा. ऑफिस पॅकेज, पीडीएफ रीडर, मीडिया प्लेयर इ (Install necessary application software for Windows i.e. Office Package, PDF Reader, Media Player etc)		43
1.4.19	ब्ल्यूटूथ आणि वाय-फाय सेटिंग्ज कॉन्फिगर करा (Configure Bluetooth and Wi-Fi settings)		44
1.4.20&21	डीव्हीडी, सीडी आणि बर्निंग डीव्हीडी (DVDs, CDs and burning DVDs)		47
	<b>मॉड्यूल 5 : DOS कमांड लाइन इंटरफेस (DOS Command Line Interface)</b>		
1.5.22	डिरेक्टरी लिस्ट साठी बेसिक DOS कमांड वापरा (Use basic DOS commands for directory listing)	1	49
1.5.23	DOS कमांड वापरून फाइल्स आणि फोल्डर्स मॅनेज करा (Manage files and folders using DOS commands)		52
	<b>मॉड्यूल 6 : उबंटू लिनक्स ऑपरेटिंग सिस्टम इंस्टाल करा आणि बेसिक लिनक्स कमांड कार्यान्वित करा (Install Ubuntu Linux operating system and execute basic Linux commands)</b>		
1.6.24&25	लिनक्स ऑपरेटिंग सिस्टमचा परिचय (Introduction to Linux operating system)	1	56
1.6.26-29	आज्ञा आणि विविध एडिटर हाताळणे (Handling commands and various editors)		60

एक्सरसाईस क्र.	धड्याचे शीर्षक	शिकत आहे परिणाम	पृष्ठ क्र.
1.7.30-33	<b>मॉड्यूल 7 : वर्ड प्रोसेसिंग सॉफ्टवेअर वापरणे (Using Word Processing Software)</b> एमएस वर्ड 2010 थिअरी (MS WORD 2010 THEORY)	2	85
1.8.34-36	<b>मॉड्यूल 8 : डॉक्युमेंट्स फॉर्मॅट करा (Format documents)</b> टेक्स्ट आणि प्याराग्राफ इन्सर्ट, फॉर्मॅटिंग करा, डॉक्युमेंट्स विपार्ट तयार करा आणि कॉन्फिगर करा (Insert, format text and paragraphs, Create and configure document sections)	2	94
1.9.37-39	<b>मॉड्यूल 9 : टेबल आणि याद्या मॅनेज करा (Manage Tables and Lists)</b> टेबल तयार करा, बदला (Create, modify tables)	2	96
1.10.40&41	<b>मॉड्यूल 10 : रेफरेन्स तयार करा आणि मॅनेज करा (Create and Manage References)</b> रेफरेन्स एलिमेंट आणि सारण्या तयार करा आणि मॅनेज करा (Create and manage reference elements and tables)	2	98
1.11.42-45	<b>मॉड्यूल 11 : ग्राफिक एलिमेंट मॅनेज करा (Manage Graphic Elements)</b> चित्रे आणि टेक्स्ट बॉक्स इन्सर्ट, फॉर्मॅटिंग करा (Insert, format illustrations and text boxes)	2	100
1.12.46&47	<b>मॉड्यूल 12 : डॉक्युमेंट्स कोलायब्रेशन मॅनेज करा (Manage Document Collaboration)</b> कमेंट बदल ट्रॅकिंग आणि मेलिंग मॅनेज करा (Manage comments change tracking and mailings)	2	102
1.13.48	<b>मॉड्यूल 13 : मेलिंग मॅनेज करा (Manage Mailings)</b> मेल मर्ज करा (Perform mail merge)	2	106
1.14.49-54	<b>मॉड्यूल 14 : स्प्रेड शीट ऍप्लिकेशन वर्कशीट्स आणि वर्कबुक्स मॅनेज करा (Spread Sheet Application, Manage Worksheets and Workbooks)</b> MS Excel मध्ये फाइल्स उघडा (Open files in MS Excel)	3	110
1.15.55-57	<b>मॉड्यूल 15 : डेटा सेल आणि रेन्ज मॅनेज करा (Manage Data Cells and Ranges)</b> डेटा हाताळा (Manipulate data)	3	112
1.16.58-60	<b>मॉड्यूल 16 : टेबल आणि टेबल डेटा मॅनेज करा (Manage Tables and Table Data)</b> टेबल तयार करा आणि फॉर्मॅटिंग करा (Create and format tables)	3	114
1.17.61-63	<b>मॉड्यूल 17 : सूत्रे आणि फंक्शन वापरून ऑपरेशन्स करा (Perform Operations using Formulas and Functions)</b> MS-Excel 2010 मधील फंक्शन आणि सूत्रे (Functions and formulas in MS-Excel 2010)	3	115



एक्सरसाईस क्र.	धड्याचे शीर्षक	शिकत आहे परिणाम	पृष्ठ क्र.
1.18.64-66	<b>मॉड्यूल 18 : मेलिंग मॅनेज करा (Manage Mailings)</b> चार्ट मॅनेज करा (Manage Charts)	3	120
1.19.67	<b>मॉड्यूल 19 : पिव्होट टेबल मॅनेज करा (Manage Pivot Tables)</b> पिव्होट टेबल तयार करा (Create Pivot Tables)	3	122
1.20.68-74	<b>मॉड्यूल 20 : पॉवर पॉइंट प्रेजेंटेशन (Power Point Presentations)</b> एमएस पॉवरपॉइंट प्रेजेंटेशनमध्ये फाइल्स उघडा (Open files in MS PowerPoint Presentations)	4	123
1.21.75-77	<b>मॉड्यूल 21 : फॉर्मेट प्रेजेंटेशन (Format Presentations)</b> इन्सर्ट , टेक्स्ट आणि प्याराग्राफ फॉर्मॅटिंग करा (Insert, Format text and paragraphs)	4	124
1.22.78-80	<b>मॉड्यूल 22 : टेबल आणि बुलेट केलेला टेक्स्ट मॅनेज करा (Manage Tables and Bulleted Text)</b> टेबल तयार करा, टेबल्स मॉडिफाई करा, बुलेट केलेला टेक्स्ट सुधारा (Create tables, modify tables, modify bulleted text)	4	125
1.23.81-83	<b>मॉड्यूल 23 : ग्राफिक एलिमेंट मॅनेज करा (Manage Graphic Elements)</b> चित्रे, फॉर्मेट चित्रे आणि टेक्स्ट बॉक्स इन्सर्ट (Insert illustrations, Format illustrations and text boxes)	4	126
1.24.84&85	<b>मॉड्यूल 24 : ऑडिओ आणि व्हिडिओ एलिमेंट मॅनेज करा (Manage Audio &amp; Video Elements)</b> ऑडिओ आणि व्हिडिओ एलिमेंट (Audio & Video Elements)	4	128
1.25.86&87	<b>मॉड्यूल 25 : ट्रांजिशन आणि ॲनिमेशन मॅनेज करा (Manage Transitions and Animations)</b> स्लाइड ट्रांजिशन आणि ॲनिमेशन ऑड करा (Add slide transitions & animations)	4	129
1.26.88	<b>मॉड्यूल 26 : कोलायब्रेशन मॅनेज करा (Manage Collaboration)</b> कमेंट ऑड करा आणि मॅनेज करा (Add and manage comments)	4	130
1.27.89-91	<b>मॉड्यूल 27 : MySQL वर प्रात्यक्षिक (Demonstrate on MySQL)</b> MySQL मध्ये डेटाबेस इंस्टाल करा, समस्यानिवारण करा, तयार करा आणि वापरा (Install, Troubleshoot, Create and Use of database in MySQL)	5	131
1.27.92-96	नॉर्मलायजेशन नियम, विविध डेटाटाइप, डेटा इंटिग्रिटी, डीडीएल, डीएमएल आणि डीसीएल स्टेटमेंट वापरून डेटाबेस डिझाइन करणे प्रायमरी की आणि फॉरेन की (Designing database using normalization rules, various datatypes, data integrity, DDL, DML & DCL Statements Enforcing Primary key and Foreign key)		133
1.28.97&98	<b>मॉड्यूल 28 : प्रश्नवर प्रात्यक्षिक दाखवा (Demonstrate on Queries)</b> केरी इन्सर्ट आणि डिलेट करा केरी अपडेट करा (Insert and delete queries Update queries)	5	135

एक्सरसाईस क्र.	धड्याचे शीर्षक	शिकत आहे परिणाम	पृष्ठ क्र.
1.29.99&100	<b>मॉड्यूल 29 : फंक्शन दाखवा (Demonstrate on Functions)</b> नंबर , date आणि कॅरक्टर फंक्शन्स वापरून, सब क्वेरी करून ग्रुप करा (Using the Number, Date and Character functions, group by having, sub query)	5	136
1.30.101-112	<b>मॉड्यूल 30 : संगणक नेटवर्क सेट अप आणि कॉन्फिगर करा (Set-up &amp; Configure a Computer Network)</b> संगणकाला नेटवर्कशी कनेक्ट करा आणि डिव्हाइसेस उदा. प्रिंटर, फाइल्स, फोल्डर्स आणि ड्राइव्ह शेअर करा (Connect a computer to a network and share Devices i.e. Printers, files, folders and drives)	6	138
1.31.113 1.31.114	<b>मॉड्यूल 31 : HTML टॅग वापरून साधी स्टेटिक वेब पेजेज तयार करा (Create Simple Static Web Pages using HTML Tags)</b> वेब डिझायनिंग (Web designing) CMS आणि वेब ऑथरिंग टूल्सचा परिचय (Introduction to CMS and web authoring tools)	7	147 169
1.32.115 1.32.116&117 1.32.118 1.32.119 1.32.120 1.32.121	<b>मॉड्यूल 32 : JavaScript HTML पेज मध्ये JavaScript एम्बेड करा (JavaScript Embed JavaScript in HTML Pages)</b> JavaScript समजून घेणे (Understanding JavaScript) JavaScript व्हेरिएबल आणि डेटा टाइप वापरणे (Using JavaScript Variable and data types) जावा स्क्रिप्ट मधील स्टेटमेंट्स, लूप आणि पॉपअप बॉक्स कंट्रोल करा (Control statements, Loops and Popup boxes in JavaScript) JavaScript मध्ये अरे (Arrays in JavaScript) JavaScript वापरून डायनॅमिक HTML पेजेज विकसित करा (Develop dynamic HTML pages using JavaScript) IIS वापरून वेब प्रकल्प डिप्लॉय करा (Deploy web project using IIS)	8	171 176 183 189 197 213
1.33.122-129	<b>मॉड्यूल 33 : डेटा व्हिज्युअलायझेशन किंवा एक्सेल वापरून विश्लेषण (Data Visualization or Analysis using Excel)</b> एडवान्स फॉर्मूला आणि मॅक्रो तयार करा (Create advanced formulas and macros)	9	221
1.34.130-132 1.34.133 1.34.134-137	<b>मॉड्यूल 34 : प्रोडक्ट आणि सर्विसेज साठी ई-कॉमर्स साइट्स ब्राउझ करा (Browse E-Commerce Sites to Identify Products &amp; Services)</b> ई-कॉमर्स साइट्स ब्राउझ करा (E-Commerce scope and benefits) ई-कॉमर्स साइटवर व्यवहार करा (Undertake transactions on an e-commerce site) ई कॉमर्स सुरक्षा समस्या आणि पेमेंट गेटवे (E-Commerce Security issues and Payment Gateways)	10	228 230 231

एक्सरसाईस क्र.	धड्याचे शीर्षक	शिकत आहे परिणाम	पृष्ठ क्र.
1.35.138	<b>मॉड्यूल 35 : व्हायरस, स्पायवेअर आणि इतर मालिशियस कोडपासून इनफार्मेशन , संगणक आणि नेटवर्कचे प्रोटेक्शन करा (Protect Information, Computers and Networks from Viruses, Spyware and other Malicious Code)</b> इनफार्मेशन सुरक्षा आणि धोक्यांचे ओवरविव्ह (Overview of information security and threats)	11	235
1.35.139	प्रायव्हसी (प्रायव्हसी)प्रोटेक्शन आणि आयटी कायदा (Privacy Protection and IT Act)		267
1.36.140-143	<b>मॉड्यूल 36 : क्लाउड कॉम्प्युटिंग (Cloud Computing)</b> क्लाउड सर्विस सह कार्य करणे (Working with Cloud Services)	12	272
1.37.144&145	<b>मॉड्यूल 37 : एप्लिकेशन विकसित करा आणि एप्लिकेशन डेव्हलपमेंट लाइफ साइकल करा (Develop an application and perform the Application Development Life Cycle)</b> एप्लिकेशन डेव्हलपमेंट लाइफ साइकल टप्पे ओळखा (Identify Phases of the Application Development Life Cycle)	12	277
1.38.01&02	<b>मॉड्यूल 38 to 42 : इलेक्टिव्ह मॉड्यूल I - Java मध्ये प्रोग्रामिंग (Elective Module I - Programming in Python)</b> ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग आणि जावा भाषा (Programming language (Python) Use Python from command line)		282
1.39.03-05	डेटा टाइप आणि ऑपरेटर वापरून ऑपरेशन्स करा (Perform Operations using Data Types and Operators)	13	285
1.40.06&07	डि सीजन आणि लूपसह कंट्रोल फ्लो फ्लो करा (Control Flow with Decisions and Loops)		290
1.41.08&09	डॉक्युमेंट्स आणि स्ट्रक्चर कोड (Document and Structure Code)		294
1.42.10&11	मॉड्युल्स आणि टूल्स वापरून ऑपरेशन्स करा (Perform Operations Using Modules and Tools)		313
1.38.01-03	<b>मॉड्यूल 38 to 42: इलेक्टिव्ह मॉड्यूल II - Java मध्ये प्रोग्रामिंग (Elective Module II Programming in JAVA)</b> ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग आणि जावा भाषा (Object Oriented Programming and JAVA Language)		318
1.39.04-07	JAVA प्रोग्राम लिहिण्याचे प्रात्यक्षिक दाखवा (Demonstrate writing JAVA programs)		323
1.40.08-16	JAVA प्रोग्राम फ्लो कंट्रोल (JAVA Program Flow Control)	14	337
1.41.17-23	JAVA क्लास , ओव्हरलोडिंग आणि इन्हेरिटन्स (JAVA Classes, Overloading and Inheritance)		342
1.42.24-30	JAVA मधील ऑब्सट्रक्ट क्लास आणि इंटरफेस (Abstract Classes and Interfaces in JAVA)		349

## शिकणे / मूल्यांकन करण्यायोग्य परिणाम

हे पुस्तक पूर्ण झाल्यावर तुम्ही सक्षम व्हाल

क्र. सं.	शिकण्याचा परिणाम	संदर्भ उदा. क्र.
1	Install and setup operating system and related software in a computer following safety precautions. (Mapped NOS: SSC/N3022)	1.1.01 - 1.6.29
2	Create, format, and edit document using word processing application software.(Mapped NOS: SSC/N3022)	1.7.30 - 1.13.48
3	Create, format, edit and develop a workbook by using spreadsheet application software. (Mapped NOS: SSC/N3022)	1.14.49 - 1.19.67
4	Create and customize slides for presentation. (Mapped NOS: SSC/N3022)	1.20.68 - 1.26.88
5	Create and manage database file by using MySQL. (NOS: SSC/N9401)	1.27.89 - 1.29.100
6	Install, setup/configure, troubleshoot, and secure computer network including Internet. (Mapped NOS: SSC/N3022)	1.30.101 - 1.30.112
7	Develop web pages using HTML and CSS. (Mapped NOS: SSC/N0503, SSC/N0501)	1.31.113 - 1.31.114
8	Develop web pages using JavaScript. (Mapped NOS: SSC/N0503, SSC/N0501)	1.32.115 - 1.32.121
9	Create workbooks with advanced formulas, macros, charts, pivot tables and demonstrate ability to use Power tools. (NOS: SSC/N9402)	1.33.122 - 1.33.129
10	Browse, select, and transact using E- commerce websites (NOS: SSC/N9403)	1.34.130 - 1.34.137
11	Secure information from Internet by using cyber security concept. (NOS: SSC/N9404)	1.35.138 - 1.35.139
12	Explain Cloud concepts & services and Describe Application Development Life Cycle. (NOS: SSC/N9405)	1.36.140 - 1.37.145
13	Write programs using Python language. (NOS: SSC/N9406)	1.38.01 - 1.42.11
14	Writing programs using JAVA. (SSC/N9407)	1.38.01 - 1.42.30

### SYLLABUS

Duration	Reference Learning Outcome	Professional Skills (Trade Practical) with Indicative hours	Professional Knowledge (Trade Theory)
Professional Skill- 94 Hrs; Professional Knowledge - 32 Hrs	Install and setup operating system and related software in a computer following safety precautions. (Mapped NOS: SSC/N3022)	<p><b>Safe working practices (10 Hrs)</b></p> <ol style="list-style-type: none"> <li>1. Visit COPA Lab. of the institutes and locate the electrical connections with computer system setup. (3 Hrs)</li> <li>2. Identifying safety symbols and hazard identification. (3 Hrs)</li> <li>3. Practice safe methods of fire fighting in case of electrical fire. (2 Hrs)</li> <li>4. Use of fire extinguishers. (2Hrs)</li> </ol> <p><b>Assemble a Desktop PC (8 hrs)</b></p> <ol style="list-style-type: none"> <li>5. Identify computer peripherals and internal components of a desktop computer. (4 Hrs)</li> <li>6. Assemble components of desktop computer. (4 Hrs)</li> </ol> <p><b>Using Windows Operating Systems (20 hrs)</b></p> <ol style="list-style-type: none"> <li>7. Practice on Windows interface and navigating windows. (3 Hrs)</li> </ol>	<p><b>Introduction to Computers (3 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Safe working practices</li> <li>• Scope of the COPA trade.</li> <li>• Safety rules and safety signs.</li> <li>• Types and working of fire extinguishers. Introduction to Computer components</li> </ul> <p><b>Introduction to computer system (4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Concepts of Hardware and Software.</li> <li>• Function of mother board components and various processors.</li> <li>• Various Input/ Output devices in use and their</li> </ul>

	<p>8. Practice on managing files and folders using removable drives. (4 Hrs)</p> <p>9. Customize the desktop (2 hrs)</p> <p>10. Settings and manage user accounts. (1 Hr)</p> <p>11. View system properties and control panel details. (3 Hrs)</p> <p>12. Work with keyboard shortcut commands. (4 Hrs)</p> <p>13. Print and scan document using different commands. (3 Hrs)</p> <p><b>Computer basics and Software Installation (20 Hrs)</b></p> <p>14. View the BIOS settings and their modifications. (3 Hrs)</p> <p>15. Install Windows operating system. (4 Hrs)</p> <p>16. Format hard disk and create partition. (3 Hrs)</p> <p>17. Identify and rectify common hardware and software issues during OS installation. (3 Hrs)</p> <p>18. Install necessary application software for Windows i.e. Office Package, PDF Reader, Media Player etc. (2 Hrs)</p> <p>19. Configure Bluetooth and Wi-Fi settings. (1 Hr)</p> <p>20. Install Drivers for printer, scanner, webcam and DVD etc. (2 Hrs)</p> <p>21. Burn data, video and audio files on CD/DVD using application software. (2 Hrs)</p> <p><b>DOS Command Line Interface (9Hrs)</b></p> <p>22. Use basic DOS commands for directory listing. (5 Hrs)</p> <p>23. Manage files and folders using DOS commands. (4 Hrs)</p> <p><b>Install Ubuntu Linux operating system and execute basic Linux commands (27 Hrs)</b></p> <p>24. Installation of Ubuntu Linux operating system (6 Hrs)</p> <p>25. Install necessary application software for Linux i.e. Office Package, PDF Reader, Media Player etc. (4 Hrs)</p> <p>26. Use Basic Linux commands for directory listing, file and folder management, password etc. (6 Hrs)</p> <p>27. Use the Linux graphical user interface for file and folder management, exploring the system etc. (6 Hrs)</p> <p>28. Customize desktop settings and manage user accounts in Linux. (3 Hrs)</p>	<p>features</p> <p>Introduction Windows Operating System (9 Hrs)</p> <ul style="list-style-type: none"> <li>• Introduction to operating System</li> <li>• Main features of Windows OS</li> <li>• Concept of various shortcut commands.</li> </ul> <p><b>Introduction to the booting process (6 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to various types of memories and their features.</li> <li>• Basic Hardware and software issues and their solutions.</li> <li>• Usage of Application software and Antivirus.</li> </ul> <p><b>Introduction to DOS Command Line Interface &amp; Linux Operating Systems (10 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to basic DOS Internal and External Commands.</li> <li>• Introduction to Open Source Software</li> <li>• Introduction to Linux Operating System features, structure, files and processes</li> <li>• Basic Linux commands.</li> </ul>
--	--	--

		29. View system properties and manage system setting in Linux. (2 Hrs)	
Professional Skill - 47 Hrs.; Professional Knowledge - 14 Hrs	Create, format, and edit document using word processing application software. (Mapped NOS: SSC/N3022)	<p><b>Using Word Processing Software (47 hrs)</b></p> <p><b>Manage documents (11 Hrs.)</b></p> <p>30. Navigate within documents (2 Hrs)</p> <ul style="list-style-type: none"> <li>• Search for text</li> <li>• Link to locations within documents</li> <li>• Move to specific locations and objects in documents</li> <li>• Show and hide formatting symbols and hidden text</li> </ul> <p><b>31. Format documents (4.6 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Set up document pages</li> <li>• Apply style sets</li> <li>• Insert and modify headers and footers</li> <li>• Configure page background elements</li> </ul> <p><b>32. Save and share documents (2 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Save documents in alternative file formats</li> <li>• Modify basic document properties</li> <li>• Modify print settings</li> <li>• Share documents electronically</li> </ul> <p><b>33. Inspect documents for issues (2.4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Locate and remove hidden properties and personal information</li> <li>• Locate and correct accessibility issues</li> <li>• Locate and correct compatibility issues</li> </ul> <p><b>Format documents (8 Hrs.)</b></p> <p>34. Insert text and paragraphs (2 Hrs)</p> <ul style="list-style-type: none"> <li>• Find and replace text</li> <li>• Insert symbols and special characters</li> </ul> <p><b>35. Format text and paragraphs (3 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Apply text effects</li> <li>• Apply formatting by using Format Painter</li> <li>• Set line and paragraph spacing and indentation</li> <li>• Apply built-in styles to text</li> <li>• Clear formatting</li> </ul> <p><b>36. Create and configure document sections (3 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Format text in multiple columns</li> <li>• Insert page, section, and column breaks</li> <li>• Change page setup options for a section</li> </ul>	<p><b>Using Word Processing Software (14 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to the various applications in MS office.</li> <li>• Introduction to Word features, Office button, toolbars.</li> <li>• Creating, saving and formatting and printing documents using Word.</li> <li>• Working with objects, macro, mail merge, templates and other tools in Word.</li> </ul>

**Manage tables and lists (9.5 Hrs)**

**37.Create tables (3 Hrs)**

- Convert text to tables
- Convert tables to text
- Create tables by specifying rows and columns

**38.Modify tables (3 Hrs)**

- Sort table data
- Configure cell margins and spacing
- Merge and split cells
- Resize tables, rows, and columns
- Split tables
- Configure a repeating row header

**39.Create and modify lists (3.5 Hrs)**

- Format paragraphs as numbered and bulleted lists
- Change bullet characters and number formats
- Define custom bullet characters and number formats
- Increase and decrease list levels
- Restart and continue list numbering
- Set starting number values Create and manage references (3 Hrs.)

**40.Create and manage reference elements (1.4 Hrs)**

- Insert footnotes and endnotes
- Modify footnote and endnote properties
- Create and modify bibliography citation sources
- Insert citations for bibliographies

**41.Create and manage reference tables (1.6 Hrs)**

- Insert tables of contents
- Customize tables of contents
- Insert bibliographies

**Manage graphic elements (8.5 Hrs.)**

**42.Insert illustrations and text boxes (3 Hrs)**

- Insert shapes
- Insert pictures
- Insert 3D models
- Insert Smart Art graphics
- Insert screenshots and screen clippings
- Insert text boxes

**43.Format illustrations and text boxes (3 Hrs)**

- Apply artistic effects
- Apply picture effects and picture styles
- Remove picture backgrounds
- Format graphic elements
- Format SmartArt graphics
- Format 3D models

		<p><b>44. Add text to graphic elements (1 Hr)</b></p> <ul style="list-style-type: none"> <li>• Add and modify text in text boxes</li> <li>• Add and modify text in shapes</li> <li>• Add and modify SmartArt graphic content</li> </ul> <p><b>45. Modify graphic elements (1.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Position objects</li> <li>• Wrap text around objects</li> <li>• Add alternative text to objects for accessibility</li> </ul> <p><b>Manage document collaboration (3.5 Hrs.)</b></p> <p><b>46. Add and manage comments (1 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Add comments</li> <li>• Review and reply to comments</li> <li>• Resolve comments</li> <li>• Delete comments</li> </ul> <p><b>47. Manage change tracking (2.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Track changes</li> <li>• Review tracked changes</li> <li>• Accept and reject tracked changes</li> <li>• Lock and unlock change tracking</li> </ul> <p><b>Manage Mailings (3.5 Hrs)</b></p> <p><b>48. Perform mail merge (3.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Create envelopes</li> <li>• Create labels</li> <li>• Create a new mailing list</li> <li>• Perform mail merge using an existing list</li> </ul>	
<p>Professional Skill - 72 Hrs.;</p> <p>Professional Knowledge - 18 Hrs</p>	<p>Create, format, edit and develop a workbook by using spreadsheet application software. (Mapped NOS: SSC/N3022)</p>	<p><b>Spread Sheet Application (72 Hrs)</b></p> <p><b>Manage Worksheets and Workbooks (12 Hrs.)</b></p> <p><b>49. Open files in MS Excel (1.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Open MS Excel</li> <li>• Create a new Excel file</li> <li>• Create a new Excel file from a template</li> <li>• Open an existing Excel file</li> </ul> <p><b>50. Import data (1.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Import data from txt files</li> <li>• Import data from csv files</li> </ul> <p><b>51. Navigate within workbooks (2 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Search data</li> <li>• Navigate to named cells, ranges or workbook elements</li> <li>• Insert and remove hyperlinks</li> </ul> <p><b>52. Format worksheets and workbooks (2 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Modify page setup</li> <li>• Adjust row height and column width</li> <li>• Customize headers and footers</li> </ul> <p><b>53. Customize options and views (3 Hrs)</b></p>	<p><b>Spread Sheet Application (18 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to Excel features and Data Types.</li> <li>• Cell referencing and linking Sheets.</li> <li>• Introduction to various functions in all categories of Excel.</li> <li>• Concepts of sorting, filtering and validating data.</li> <li>• Analyzing data using charts, data tables, pivot tables, goal seek and scenarios</li> </ul>



- Display and modify content in different views
- Freeze rows and columns
- Change window views
- Modify basic workbook properties
- Display formulas
- 54. Configure content for collaboration (2 Hrs)**
- Set a print area
- Save workbooks in alternative file formats
- Configure print settings Manage data cells and ranges (12 Hrs.)
- 55. Manipulate data (3 Hrs)**
- Paste data by using special paste options
- Fill cells by using Auto Fill
- Insert and delete multiple columns or rows
- Insert and delete cells
- 56. Format cells and ranges (5 Hrs)**
- Merge and Unmerge cells
- Modify cell alignment, orientation and indentation
- Format cells using Format Painter
- Wrap text within cells
- Apply number formats
- Apply cell formats from the Format cells dialog box
- Apply cell styles
- Clear cell formatting
- 57. Define and reference named ranges (4 Hrs)**
- Define a named range
- Name a table
- Summarize data visually Insert spark lines
- Apply built in conditional formatting
- Remove conditional formatting
- Manage tables and table data (12 Hrs.)**
- 58. Create and format tables (4 hrs)**
- Create excel tables from cell ranges
- Apply table styles
- Convert tables to cell ranges
- 59. Manage tables and table data (5 Hrs)**
- Add or remove table rows and columns
- Configure table style options
- Insert and configure total rows
- 60. Filter and sort table data (3 Hrs)**
- Filter records
- Sort data by multiple columns Perform operations using formulas and functions (12 Hrs.)
- 61. Insert references (3 Hrs)**

		<ul style="list-style-type: none"> <li>• Insert relative, absolute and mixed references</li> <li>• Reference named ranges and named tables in formulas</li> </ul> <p><b>62. Calculate and transform data (5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Perform calculations using AVERAGE(), MIN(), MAX() and SUM()</li> <li>• Count cells by using COUNT(), COUNTIF() and COUNTBLANK()</li> <li>• Perform conditional operations by using the IF() function</li> </ul> <p><b>63. Format and modify text(4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Format text using RIGHT(), LEFT() and MID() functions</li> <li>• Format text using UPPER(), LOWER() and LEN() functions</li> <li>• Format text using CONCAT() and TEXTJOIN() functions</li> </ul> <p><b>Manage Charts (12 Hrs.)</b></p> <p><b>64. Create Charts (3 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Create charts</li> <li>• Create chart sheets</li> </ul> <p><b>65. Modify charts (4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Add data series to charts</li> <li>• Switch between rows and columns in source data</li> <li>• Add and modify chart elements</li> <li>• Add trend lines to chart</li> </ul> <p><b>66. Format charts (5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Apply chart layouts</li> <li>• Apply chart styles</li> <li>• Add alternative text to charts for accessibility</li> </ul> <p><b>Manage Pivot Tables (12 Hrs.)</b></p> <p><b>67. Create Pivot Tables(12 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Create Pivot tables from cell ranges</li> <li>• Manipulate fields (columns) to get desired analysis</li> <li>• Use Filters for pivot tables</li> <li>• Represent data as Count, Sum, Average &amp; % of row / column</li> <li>• Group data in Columns &amp; rows for aggregate reports</li> </ul>	
Professional Skill - 53 Hrs; Professional Knowledge - 13 Hrs	Create and customize slides for presentation. (Mapped NOS: SSC/N3022)	<p><b>Power point Presentations (9.5 Hrs.)</b></p> <p><b>68. Open files in MS PowerPoint (1 Hr)</b></p> <ul style="list-style-type: none"> <li>• Open MS PowerPoint</li> <li>• Create a new PowerPoint file</li> <li>• Create a new PowerPoint file from a template</li> <li>• Open an existing PowerPoint file</li> </ul> <p><b>69. Format PowerPoint Presentations (1.5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Add slides</li> <li>• Add titles and text</li> <li>• Select slide layouts</li> </ul>	<p><b>Power point Presentations (13 Hrs.)</b></p> <ul style="list-style-type: none"> <li>• Image editing, Presentations</li> <li>• Introduction to Open Office.</li> <li>• Introduction to the properties and editing of images.</li> <li>• Introduction to different formats of images and their uses.</li> <li>• Introduction to Power Point and its advantages.</li> <li>• Creating Slide Shows.</li> </ul> <p>Fine tuning the presentation and good presentation technique</p>

- Add PowerPoint templates
- Duplicate slides
- 70. Modify slide masters, handout masters, and note masters (2 Hrs)**
- Change the slide master theme or background
- Modify slide master content
- Modify slide layouts
- 71. Change presentation options and views (1.5 Hrs)**
- Change slide size
- Display presentations in different views
- 72. Save and share PowerPoint Presentations (1.5 Hrs)**
- Save presentations in alternative file formats
- Configure different Print settings
- Share presentations electronically
- 73. Configure and present slide shows (1 Hr)**
- Hide unwanted slides while presenting
- Configure slide show options
- Present slide shows by using Presenter View
- 74. Prepare presentations for collaboration (1 Hrs)**
- Protect presentations by using passwords
- Export presentations to other formats
- Format presentations (6.5 Hrs.)**
- 75. Insert text and paragraphs (2 Hrs)**
- Find and replace text
- Insert symbols and special characters
- 76. Format text and paragraphs (2 Hrs)**
- Apply text effects
- Apply formatting by using Format Painter
- Set line and paragraph spacing and indentation
- Apply built-in styles to text
- 77. Create and configure sections (2.5 Hrs)**
- Format text in multiple columns
- Text and image presentation styles
- Clear formatting
- Manage tables and bulleted text (8 Hrs)**
- 78. Create tables (3 Hrs)**
- Insert tables in PowerPoint

- Apply built-in table styles
  - Create tables by specifying rows and columns
- 79. Modify tables (3 Hrs)**
- Insert and delete table rows and columns
  - Configure cell margins and spacing
  - Merge and split cells
  - Resize tables, rows, and columns
- 80. Create and modify bulleted text (2 Hrs)**
- Format paragraphs as numbered and bulleted lists
  - Change bullet characters and number formats
  - Increase and decrease list indents
  - Set starting number values
  - Restart and continue list numbering on different slides
- Create and manage reference elements (hyperlinks) (1.25 Hrs)**
- Create hyperlinks within presentations
  - Create hyperlinks in presentations for files and other sources
- Manage graphic elements (11.5 Hrs)**
- 81. Insert illustrations and text boxes (3.5 Hrs)**
- Insert shapes
  - Insert pictures
  - Insert SmartArt graphics
  - Insert screenshots and screen clippings
- 82. Format illustrations and text boxes (4 Hrs)**
- Apply artistic effects
  - Apply picture effects and picture styles
  - Remove picture backgrounds
  - Crop images
  - Format graphic elements
  - Format SmartArt graphics
- 83. Add and modify text in graphic elements (4 Hrs)**
- Add and modify text in text boxes
  - Add and modify text in shapes
  - Add and modify SmartArt graphic text
  - Create, insert and modify charts
- Manage Audio & Video elements (6.5 Hrs)**
- 84. Add Audio elements (2 Hrs)**
- Import audio files in presentations
  - Configure audio playback options
- 85. Add Video elements (4.5 Hrs)**
- Import video files in presentations

		<ul style="list-style-type: none"> <li>• Resize video to fit slide</li> <li>• Configure video playback options</li> </ul> <p><b>Manage transitions and animations (9 Hrs)</b></p> <p><b>86.Add slide transitions (5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Add same slide transition for all slides</li> <li>• Set transition effect duration</li> <li>• Configure transition start and finish options</li> <li>• Customise select slide transitions</li> </ul> <p><b>87.Add animations (4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Animate text and graphic elements</li> <li>• Order shapes, images, and text boxes</li> <li>• Group shapes, images, and text boxes</li> <li>• Configure animation effects</li> <li>• Configure animation paths</li> <li>• Reorder animations on a slide</li> </ul> <p><b>Manage collaboration (0.75 Hrs)</b></p> <p>88.Add and manage comments (0.75 Hrs)</p> <ul style="list-style-type: none"> <li>• Add comments</li> <li>• Review and reply to comments</li> </ul>	
Professional Skill - 53 Hrs.; Professional Knowledge - 13 Hrs	Create and manage database file by using MySQL. (NOS: SSC/N9401)	<p><b>Demonstrate on (15 Hrs.)</b></p> <p>89.Installation of MySQL. (1 Hr)</p> <p>90.Troubleshooting basic installation issues. (1 Hr)</p> <p>91.Creation and use of database. (3 Hr)</p> <p>92.Designing of tables. (3 Hr)</p> <p>93.Applying data integrity rules. (2 Hr)</p> <p>94.Using the DDL, DCL and DML statements. (2 Hrs)</p> <p>95. Enforcing constraints, primary key and foreign key. (2 Hrs)</p> <p>96. Adding indices to Tables. (1 Hr)</p> <p><b>Demonstrate on (15 Hrs)</b></p> <p>97.Simple select queries. (5 Hrs)</p> <p>98.Insert and delete queries Update queries. (10 Hrs)</p> <p><b>Demonstrate on (23 Hrs)</b></p> <p>99. Using the Number, Date and Character functions. Joins and Functions (11.5 Hrs)</p> <p>100. Joins, Group by, Having, Sub query. (11.5 Hrs)</p>	<p><b>Database Concepts (4 Hrs.)</b></p> <ul style="list-style-type: none"> <li>• Concept of DBMS, RDBMS.</li> <li>• Data Models, Concept of DBA, Database Users.</li> <li>• Database Schema.</li> <li>• Designing Database using Normalization Rules.</li> <li>• Various data types Data integrity, DDL DML and DCL statements.</li> <li>• Enforcing Primary key and foreign key.</li> <li>• Adding Indices.</li> </ul> <p><b>Queries (4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Concepts of Transactions</li> <li>• ACID Property of Transaction Constraints.</li> </ul> <p><b>Joins and Functions (5 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Joining of tables</li> <li>• Sub Queries “ Functions used in query like sum, average, max, min, count etc.</li> </ul>
Professional Skill - 68 Hrs; Professional Knowledge - 16 Hrs	Install, setup/configure, troubleshoot, and secure computer network including Internet. (Mapped NOS: SSC/N3022)	<p><b>Computer Network (68 Hrs.) Set-up &amp; configure a Computer Network (48 Hrs.)</b></p> <p>101. View Network connections. (2 Hrs)</p> <p>102. Connect a computer to a network and share Devices i.e. Printers, files, folders and drives. (4 Hrs)</p> <p>103. Work with various Network devices, connectors and cables.</p>	<p><b>Communicating in a Connected World (12 Hrs.)</b></p> <ul style="list-style-type: none"> <li>• Local Networks,</li> <li>• Communicating on a Local Network, Principles of Communications,</li> <li>• How do Ethernet Networks Work?,</li> <li>• How are Networks Built?,</li> <li>• Routing Across Networks</li> </ul>

		<p>Create straight and cross cable and punch a UTP cable in the patch socket and test the connectivity. (4 Hrs)</p> <p>104. Practice IP Addressing and Subnet masking for IPV4/ IPV6 and pinging to test networks. (4 Hrs)</p> <p>105. Configure Hub and Switch. (4 Hrs)</p> <p>106. Set up and configure wired and wireless LAN in a Computer Lab within at least three computers. (6 Hrs)</p> <p>107. Use patch panel &amp; I/O Box for wired LAN and installing &amp; configuring Internet connection in a single PC and in a LAN. (6 Hrs)</p> <p>108. Set up a proxy server/ DHCP Server with firewall. (8 Hrs)</p> <p>109. Set up video conferencing using open-source software. (4 Hrs)</p> <p>110. Use various tools (by open source /free) for network troubleshooting, maintenance and security for both Wired and Wireless (6 Hrs)</p> <p>111. Set up Internet access &amp; communication (10 Hrs)</p> <ul style="list-style-type: none"> <li>• Set-up internet connectivity</li> <li>• Set-up digital communication</li> </ul> <p>112. Use the Internet (10 Hrs)</p> <ul style="list-style-type: none"> <li>• Browse the Internet</li> <li>• Use e-mail</li> <li>• Use Social Media</li> <li>• Use the phone for online activities</li> </ul>	<ul style="list-style-type: none"> <li>• Explain how end- user devices and local networks interact with the global Internet.</li> <li>• Communicating in a Connected World Explain the concept of network communication.</li> <li>• Local Networks Explain the roles of devices in a network.</li> <li>• What Does a Home Network Look Like?</li> <li>• How Does Wi-Fi Work?</li> <li>• Introduction to LAN Devices, Internetworking Devices,</li> </ul> <p><b>Internet Concepts (4 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to www, Concept of Internet, Web Browsers, internet servers and search engines.</li> <li>• Concepts of Domain naming Systems and E mail communication.</li> <li>• Introduction to video chatting tools and Social Networking concepts.</li> </ul>
<p>Professional Skill - 67 Hrs;</p> <p>Professional Knowledge - 17 Hrs</p>	<p>Develop web pages using HTML and CSS. (Mapped NOS: SSC/N0503, SSC/N0501)</p>	<p><b>Create simple static web pages using HTML tags (67 Hrs.)</b></p> <p>113. Practice HTML (46 Hrs)</p> <ul style="list-style-type: none"> <li>• Practice with basic HTML elements (e.g. head, title, body), tag and attributes.</li> <li>• Design simple web page with text, paragraph and line break using HTML tags</li> <li>• Format text, change background colour and insert pictures in web page</li> <li>• Design simple web page with tables and lists.</li> <li>• Use marquees, hyperlinks and mail to link in designing web pages</li> <li>• Create frames, add style and design layout.</li> <li>• Display a web page within a web page using iframes.</li> <li>• Insert text, check and combo box in web page.</li> <li>• Design web page using password field, submit button</li> <li>• Reset button and radio button etc.</li> </ul>	<p><b>Web Design Concepts (17 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Concepts of Static and Dynamic Web pages.</li> <li>• Introduction to HTML and various tags in HTML.</li> <li>• Concepts of different controls used in Web Pages.</li> <li>• Concepts of CSS and applying CSS to HTML.</li> <li>• Introduction to open source CMS viz, Joomla, Word press etc. and Web authoring tools viz. Kompozer, WordPress, Front Page etc.</li> </ul>

		<ul style="list-style-type: none"> <li>• Design a web page adding flash file, audio and video files.</li> <li>• Design web page with forms and form controls using HTML tags</li> </ul> <p>114. Create simple static web pages using CSS (21 Hrs )</p> <ul style="list-style-type: none"> <li>• CSS syntax, Adding colors, fonts, backgrounds, images borders, text alignment, text transformation, Lists etc.</li> <li>• 3 types of CSS</li> <li>• Adding a Navigation Bars(vertical/horizontal bars)</li> <li>• CSS drop downs &amp; Forms</li> <li>• CSS counters and website layout, Multiple backgrounds &amp; Putting the stylesheet in a separate file</li> <li>• CSS Animations &amp; CSS Buttons</li> </ul>	
Professional Skill - 173 Hrs; Professional Knowledge - 35 Hrs	Develop web pages using JavaScript. (Mapped NOS: SSC/N0503, SSC/N0501)	<p><b>JavaScript (173 Hrs)</b> <b>Embed JavaScript in HTML Pages (127 Hrs)</b></p> <p>115. Practicing the JavaScript in creating dynamic HTML pages. (53 Hrs)</p> <p>116. Embed JavaScript in HTML to Display Information in Web pages. (31 Hrs)</p> <p>117. Use error handling techniques in JavaScript. (11 Hrs)</p> <p>118. Use objects and classes in JavaScript. (31 Hrs)</p> <p>119. Describe Animation and Multimedia using JavaScript. (1 Hr)</p> <p>Create a dynamic website using an open-source tool (40 Hrs)</p> <p>120. Develop dynamic HTML pages using JavaScript.</p> <p>Deploy a simple web project (6 Hrs)</p> <p>121. Deploy web project using IIS.</p>	<p><b>Introduction to JavaScript (35 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to Programming and Scripting Languages.</li> <li>• Introduction to JavaScript and its application for the web.</li> <li>• Introduction to Web Servers and their features.</li> <li>• JavaScript Basics - Data types, Variables, Constants and Conversion between data types.</li> <li>• Arithmetic, Comparison, Logical Operators in JavaScript. Operator precedence.</li> <li>• Program Control Statements and loops in JavaScript.</li> <li>• Arrays in JavaScript - concepts, types and usage.</li> <li>• The String data type in JavaScript. Introduction to String, Math and Date.</li> <li>• Introduction to Functions in JavaScript.</li> <li>• Built in JavaScript functions overview.</li> <li>• Concepts of Pop Up boxes in JavaScript.</li> <li>• Introduction to the Document Object Model.</li> <li>• Concepts of using Animation and multimedia files in Java Script.</li> </ul>
Professional Skill - 73 Hrs Professional Knowledge - 17 Hrs	Create workbooks with advanced formulas, macros, charts, pivot tables and demonstrate ability to use Power tools. (NOS: SSC/N9402)	<p><b>Data Visualization or analysis using Excel - (73 Hrs)</b> <b>Create advanced formulas and macros (24 Hrs)</b></p> <p>122. Create and modify simple macros (6 Hrs)</p> <p>123. Perform form controls and create simple data entry form with macros. (6 Hrs)</p>	<p><b>Advanced Excel Concepts - Theory- (17 Hrs)</b></p> <ul style="list-style-type: none"> <li>• MS excel revision (row, columns, basic formatting, insert menu, Print setup, etc. ) and Look up introduction and functions</li> <li>• Types of references and cell naming</li> <li>• Excel Linkage Custom Format and Excel Protection</li> </ul>

		<p>124. Look up data by using functions. (6 Hrs)</p> <p>125. Use advanced date functions.(6 Hrs)</p> <p><b>Manage advanced charts and tables (21 hrs)</b></p> <p>126. Create and modify advanced charts. (10 Hrs)</p> <p>127. Create and modify PivotTables. (11 Hrs)</p> <p><b>Use Power Query and Power BI (24 Hrs)</b></p> <p>128. Create a Power Query, Power Query Function. Invoking the Power Query function and combining queries. Organize the workbook queries (12 Hrs)</p> <p>129. Use Power BI for simple data visualizations. (12 Hrs)</p>	<ul style="list-style-type: none"> <li>• Tips and tricks</li> <li>• Pivot table and Pivot chart</li> <li>• Conditional formatting</li> <li>• Advanced Graphs</li> <li>• Power Queries</li> </ul>
Professional Skill - 25hrs; Professional Knowledge - 10 Hrs.	Browse, select, and transact using E-commerce websites(NOS: SSC/N9403)	<p><b>Make a dashboard in Excel (4 Hrs)</b></p> <p><b>Browse e-Commerce sites to identify products &amp; services (6.5 Hrs)</b></p> <p>130. Demonstrate e-Commerce sites. (1.5 Hrs)</p> <p>131. List features of e-commerce sites. (2 Hrs)</p> <p>132. Use e-commerce sites to source an item. (3 Hrs)</p> <p>Shop online (4.5 Hrs)</p> <p>133. Undertake transactions on an e-commerce site. (4.5 Hrs)</p> <p><b>Manage e-commerce operations (14 Hrs)</b></p> <p>134. Add products to an ecommerce website. (4 Hrs)</p> <p>135. Practice order processing. (3 Hrs)</p> <p>136. Practice payment processing. (4.5 Hrs)</p> <p>137. Identify common security issues. (3.5 Hrs )</p>	<p><b>e-Commerce (10 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Introduction to E Commerce and advantages.</li> <li>• Building business on the net.</li> <li>• Payment and Order Processing, Authorization, Chargeback and other payment methods.</li> <li>• Security issues and payment gateways.</li> </ul>
Professional Skill - 20 Hrs Professional Knowledge - 10 Hrs.	Secure information from Internet by using cyber security concept. (NOS: SSC/N9404)	<p><b>138. Protect information, computers and networks from viruses, spyware and other malicious code (19 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Explain Cyber security (2 Hrs)</li> <li>• Secure computers &amp; the network (5.5 Hrs)</li> <li>• Reduce cyber security threats (2 Hrs)</li> <li>• Secure a Wi-Fi Network (4 Hrs)</li> <li>• Use Anti-Virus software (3 Hrs)</li> <li>• Perform back-ups of files, data &amp; information (2.5 Hrs)</li> </ul> <p><b>139. Explain compliance with IT Act (1 Hr)</b></p> <ul style="list-style-type: none"> <li>• Identify steps for information privacy. (0.5 Hrs)</li> <li>• Identify common cybercrimes and penalties applicable. (0.5 Hrs)</li> </ul>	<p><b>Cyber Security (10 Hrs)</b></p> <ul style="list-style-type: none"> <li>• Overview of Information Security, S S L , H T T P S , Security threats, information Security vulnerability and Risk management.</li> <li>• Introduction to Directory Services, Access Control, Security, Privacy protection, Audit and Security.</li> <li>• Introduction to IT Act and penalties for cybercrimes.</li> </ul>



Professional Skill -25 Hrs; Professional Knowledge 15 Hrs.	Explain Cloud concepts & services and Describe Application Development Life Cycle. (NOS: SSC/N9405)	<b>Cloud Computing (15 Hrs) Working with Cloud Services (12 Hrs)</b> 140. Practice with IaaS using free cloud services. (4 Hrs) 141. Practice with PaaS using free cloud services. (4 Hrs) 142. Practice with SaaS using free cloud services. (4 Hrs) <b>Web hosting in Cloud (3 Hrs)</b> 143. Host a website in a free cloud. (3 Hrs) <b>Develop an application and perform the Application Development Life Cycle (10 Hrs)</b> 144. Identify Phases of the Application Development Life Cycle. (5 Hrs) 145. Describe Roles in each of the phases of Application Development Life Cycle. (5 Hrs)	<b>Introduction to Cloud Computing (12 Hrs)</b> <ul style="list-style-type: none"> <li>• Benefits of cloud services, different categories.</li> <li>• Resources available in cloud.</li> </ul> <b>Explain the Application Development Life Cycle (3 Hrs)</b> <ul style="list-style-type: none"> <li>• Identify Phases of the Application Development Life Cycle.</li> <li>• Describe Roles in each of phases of the Application Development Life Cycle.</li> </ul>
---	---	---	--

To be selected either “Programming in Python” or “Programming in Java” as Elective Module for COPA.

COPA - Elective Module - I		Programming in Python	
Professional Skill - 70 Hrs; Professional Knowledge - 30 Hrs.	Write programs using Python language. ( NOS: SSC/N9406)	<b>Programming language (Python) Use Python from command line (7 Hrs)</b> <ol style="list-style-type: none"> <li>1. Install, set up the environment &amp; run Python. (3 Hrs)</li> <li>2. Use Command Line and IDE to create and execute a python program. (4 Hrs)</li> </ol> <b>Perform Operations using Data Types and Operators (15 Hrs)</b> <ol style="list-style-type: none"> <li>3. Write and test a python program to demonstrate print statement, comments, different types of variables. (5 Hrs)</li> <li>4. Write and test a python program to perform data and data type operations, string operations, date, input and output, output formatting and operators. (5 Hrs)</li> <li>5. Determine the sequence of execution based on operator precedence. (5 Hrs)</li> </ol> <b>Control Flow with Decisions and Loops (20hrs)</b> <ol style="list-style-type: none"> <li>6. Construct and analyze code segments that use branching statements. (10 Hrs)</li> <li>7. Construct and analyze code segments that perform iteration. (10 Hrs)</li> </ol> <b>Document and Structure Code (18 Hrs)</b> <ol style="list-style-type: none"> <li>8. Document code segments using comments and documentation strings. (3 Hrs)</li> <li>9. Construct and analyze code segments that include List comprehensions, tuple, set and</li> </ol>	<b>Programming language (Python) (30 Hrs)</b> <ul style="list-style-type: none"> <li>• Introduction to Python History</li> <li>• Features, Setting up path Basic Syntax, Comments, Variable</li> <li>• Different Data Types</li> <li>• Casting, string, Boolean</li> <li>• Python Operators</li> <li>• Conditional Statements</li> <li>• Looping</li> <li>• Control Statements, String Manipulation, Lists, Tuple, sets</li> <li>• Dictionaries</li> <li>• Arrays</li> <li>• Iterators, modules, dates, math,</li> <li>• Modules, Input and Output.</li> </ul>

		<p>Dictionary comprehensions. (16 Hrs)</p> <p><b>Perform Operations Using Modules and Tools (10 Hrs)</b></p> <p>10. Perform basic operations using built-in modules. (5 Hrs)</p> <p>11. Solve complex computing problems by using built-in modules. (5 Hrs)</p>	
<b>COPA - Elective Module - II</b>		<b>Programming in JAVA</b>	
Professional Skill - 70 Hrs; Professional Knowledge - 30 Hrs.	Writing programs using JAVA. (SSC/N9407)	<p><b>Object Oriented Programming and JAVA Language (15 Hrs)</b></p> <p>1. Installing JAVA.</p> <p>2. Setting the Class path.</p> <p>3. Writing and Executing a simple JAVA Program to display "Hello".</p> <p><b>Demonstrate writing JAVA programs:</b></p> <p>4. Use of various data types in JAVA.</p> <p>5. Use of various operators in JAVA.</p> <p>6. Create and use of Local, Instance and Class variables.</p> <p>7. Read text from the keyboard using scanner class read text from the keyboard using console class.</p>	<p><b>Explain the following:</b></p> <ul style="list-style-type: none"> <li>Object Oriented Programming with Core Java</li> <li>Java Programming features</li> <li>JVM, Byte codes and Class path</li> <li>Java Program Development</li> <li>Compilation and Execution of JAVA programs</li> <li>Basic JAVA language elements - keywords, comments, data types and variables.</li> <li>JAVA Arithmetic, Assignment, Relational, Logical, Increment / Decrement operators and expressions.</li> <li>JAVA String Operators</li> <li>JAVA Input and Output streams, System in, System out.</li> <li>Input using Scanner class and Console class methods. (10Hrs.)</li> </ul>
		<p><b>JAVA Program Flow Control (20 Hrs)</b></p> <p><b>Demonstrate writing JAVA programs:</b></p> <p>8. Use of the if and if ... else statements.</p> <p>9. Use of the Switch statement.</p> <p>10. Use of the Do ... While and while - do loops.</p> <p>11. Use of the For Loop.</p> <p>12. Use of the Break and Continue Keywords.</p> <p>13. Use of the JAVA Numbers Class methods.</p> <p>14. Use of the JAVA Character Class methods.</p> <p>15. Use of the JAVA String Class methods.</p> <p>16. Create and use of arrays.</p> <p><b>JAVA Classes, Overloading and Inheritance (20 Hrs)</b></p> <p><b>Demonstrate writing JAVA programs:</b></p> <p>17. Create and use of simple classes, objects and methods in JAVA.</p> <p>18. Pass data and Objects to Methods.</p> <p>19. Return data and Objects from Methods.</p> <p>20. use of constructors in JAVA.</p> <p>21. Create and use of Overloaded methods in JAVA.</p> <p>22. Override methods in JAVA.</p> <p>23. Create and use of Super class, Sub class in JAVA.</p>	<p><b>Explain the following:</b></p> <ul style="list-style-type: none"> <li>Decision making and flow control using if...then, if then else, nested if, switch case and the conditional ternary operators in JAVA.</li> <li>Loop control flow using while - do, do - while loops, for loop, using the break, continue statements.</li> <li>Terminating the JAVA program. JAVA Number, Character and String Classes. Arrays in JAVA. (6Hrs.)</li> </ul> <p><b>Explain the following:</b></p> <ul style="list-style-type: none"> <li>JAVA Objects, Classes and Methods.</li> <li>Passing data and objects as parameters to methods.</li> <li>Method Overloading.</li> <li>Constructors and Overloaded constructors.</li> <li>Inheritance in JAVA.</li> <li>Method Overriding in JAVA. (8Hrs.)</li> </ul>

		<p><b>Abstract Classes and Interfaces in JAVA (15 Hrs) Demonstrate writing JAVA programs:</b></p> <p>24. Create and use virtual methods.  25. Create abstract classes and methods.  26. Create interfaces in JAVA.  27. Override methods in JAVA.  28. Create and implement an interface.  29. Extend interfaces in JAVA.  30. Create and use a package in JAVA.</p>	<p><b>Explain the following:</b></p> <ul style="list-style-type: none"> <li>• Concept of Virtual methods.</li> <li>• Concept of Abstract classes and methods</li> <li>• Features of Abstract Classes</li> <li>• JAVA Interfaces and their advantages</li> <li>• Method Overriding in JAVA</li> <li>• Polymorphism in JAVA</li> <li>• Creating, implementing and extending interfaces</li> <li>• Creating and using Packages in JAVA. (6Hrs.)</li> </ul>
		<p>Troubleshooting Java issues Download and Install Java, Check and Verify Java Configurations, Test Java, Remove Old Versions of Java, Find Java version, Always redirected to the java.com download page</p>	

**Industrial Visit/Project work / Broad Area:**

- a) Create and host a web site of at least 6 web pages using JavaScript & CSS containing interactive objects, functions etc.
- b) Create a project with Excel on Payroll Systems.
- c) Create a database with MySQL on Library management system.
- d) Create project in Python/Java programming language.

NOT TO BE REPRODUCED



## विद्युत सुरक्षा (Electrical safety)

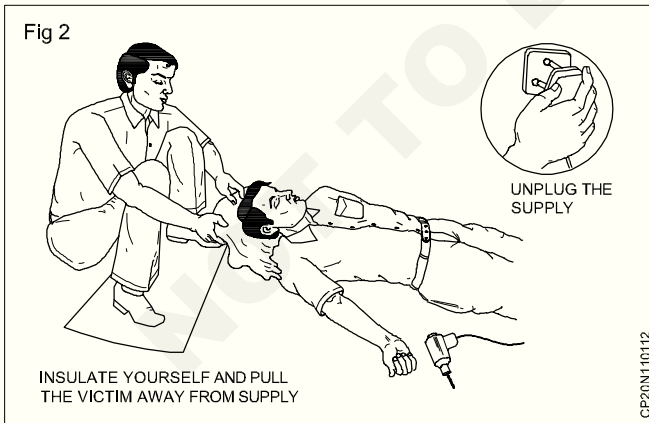
उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

• थेट वायरच्या संपर्कात असलेल्या व्यक्तीला कसे वाचवायचे ते स्पष्ट करा.

विद्युत शॉकची तीव्रता शरीरातून जाणार्या विद्युत् प्रवाहाच्या पातळीवर आणि संपर्काच्या कालावधीवर अवलंबून असते. उशीर करू नका, त्वरित कार्य करा. विद्युत प्रवाह खंडित झाला आहे याची खात्री करा.

अपघातग्रस्त व्यक्ती अद्याप पुरवठ्याच्या संपर्कात असल्यास - एकतर वीज बंद करून, प्लग काढून टाकून किंवा केबल ओपन करून संपर्क खंडित करा. नसल्यास, कोरडे लाकूड, रबर किंवा प्लॉस्टिक यांसारख्या इन्सुलेट कंटेंट वर उभे रहा किंवा स्वतःला इन्सुलेशन करण्यासाठी हाताशी जे काही आहे त्याचा वापर करा आणि व्यक्तीला ढकलून किंवा खेचून संपर्क तोडून टाका. (आकृती १ आणि २)

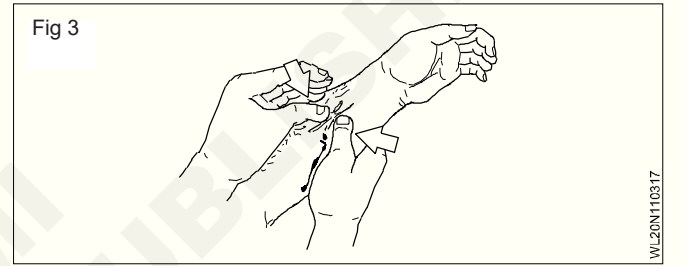
जर तुम्ही उष्णतारोधक वापरत नसाल तर, सर्किट डेड होईपर्यंत किंवा व्यक्ती इन्फॉर्मेट्स पासून दूर जाईपर्यंत तुमच्या उघड्या हातांनी पीडिताला स्पर्श करू नका.



पीडित व्यक्ती उंचीवर असल्यास, त्याला पडण्यापासून रोखण्यासाठी किंवा कमीतकमी तो सुरक्षित पडण्यासाठी उपाय योजले पाहिजेत.

पीडितेवर इलेक्ट्रिक जळणे कदाचित मोठे एरिया व्यापू शकत नाही परंतु खोलवर असू शकते. स्वच्छ, निर्जंतुक ड्रेसिंगने एरिया झाकणे आणि शॉकसाठी सबचार करणे एवढेच तुम्ही करू शकता. शक्य तितक्या लवकर तज्ञांची मदत घ्या.

जर अपघातग्रस्त व्यक्ती बेशुद्ध असेल परंतु श्वास घेत असेल तर, मान, छाती आणि कंबरेवरील कपडे सैल करा आणि अपघातग्रस्त व्यक्तीला पुनर्प्राप्तीच्या स्थितीत ठेवा. (पिक्चर ३)



श्वासोच्छवास आणि नाडीचे प्रमाण सतत तपासा.

अपघातग्रस्त व्यक्तीला उबदार आणि आरामदायी ठेवा. (पिक्चर ४) मदतीसाठी पाठवा.

बेशुद्ध माणसाला तोंडाने काहीही देऊ नका.

बेशुद्ध व्यक्तीकडे लक्ष न देता सोडू नका.

अपघातग्रस्त व्यक्ती श्वास घेत नसल्यास - त्वरित कार्य करा - वाया घालवू नका



## सुरक्षा सराव - अग्निशामक (Safety practice - fire extinguishers)

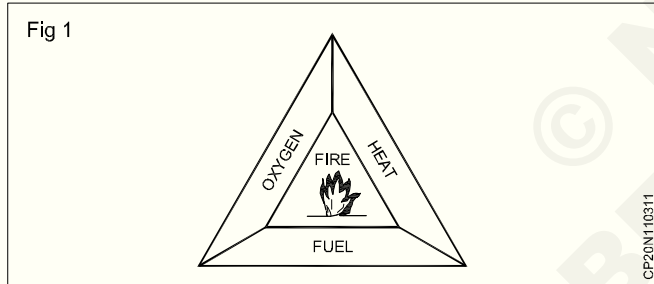
उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- थेट वायरच्या संपर्कात असलेल्या व्यक्तीला कसे वाचवायचे ते समजावून सांगा
- विविध टाइप च्या अग्निशामक यंत्रांमध्ये फरक करा
- आगीच्या वर्गाच्या आधारे वापरण्यासाठी योग्य टाइप चे अग्निशामक यंत्र निश्चित करा
- आग लागल्यास अवलंबल्या जाणाऱ्या सामान्य प्रक्रियेचे डिस्क्रिप्शन करा.

आग म्हणजे ज्वलनशील पदार्थांचे जळणे. नको त्या ठिकाणी आणि नको त्या प्रसंगी आणि अनकंट्रोलबल प्रमाणात आग लागल्यास मालमत्तेचे आणि साहित्याचे नुकसान किंवा नाश होऊ शकतो. यामुळे लोकांना दुखापत होऊ शकते आणि कधीकधी जीवितहानी देखील होऊ शकते. त्यामुळे आग रोखण्यासाठी सर्वतोपरी प्रयत्न करणे आवश्यक आहे. जेव्हा आगीचा उद्रेक आढळतो, तेव्हा ते तात्काळ सुधारात्मक कारवाईद्वारे कंट्रोल आणि विझवले जाणे आवश्यक आहे.

आग रोखणे शक्य आहे का? होय, आग लागण्यास कारणीभूत असलेल्या तीन एलिमेंटपैकी कोणासही दूर करून आग रोखली जाऊ शकते.

आग सतत जळत राहण्यासाठी खालील तीन एलिमेंट एकत्र असणे आवश्यक आहे. (आकृती क्रं 1)



**इंधन:** ऑक्सिजन आणि पुरेसे तापमान असल्यास कोणताही पदार्थ, द्रव, घन किंवा वायू जळतो.

**उष्णता:** प्रत्येक इंधन एका स्पेसिफिक तापमानात जळण्यास सुरवात होईल. ते बदलते आणि इंधनावर अवलंबून असते. गरम झाल्यावर घन आणि द्रव वाष्प सोडतात आणि ही वाफ पेटते. काही द्रव गरम करावे लागत नाहीत कारण ते सामान्य खोलीच्या तापमानाला 150 सेल्सिअस म्हटल्यास वाफ सोडतात, उदा. पेट्रोल

**ऑक्सिजन:** सामान्यतः आग पेटवत ठेवण्यासाठी हवेत पुरेशा प्रमाणात असते.

**आग विझवणे:** यापैकी कोणतेही एलिमेंट एकत्र केल्याने किंवा काढून टाकल्याने आग विझते. हे साध्य करण्याचे तीन बेसिक पाथ आहेत.

- **इंधनाची** आग ची तीव्रता कमी राहिल्याने हा एलिमेंट निघून जातो.
- **स्मोदरिंग** - म्हणजे. ऑक्सिजनच्या पुरवठ्यापासून आग विलग करून त्यावर फोम, वाळू इ.

- **कूलिंग** - तापमान कमी करण्यासाठी पाण्याचा वापर करा. यापैकी कोणतेही एक एलिमेंट काढून टाकल्यास आग विझते

**आग रोखणे:** बहुसंख्य आग लहान उद्रेकांपासून सुरू होतात जी सुरक्षित धारण होईपर्यंत कोणाच्या लक्षात येत नाहीत. बऱ्याच आगींना अधिक सावधगिरीने आणि काही सोप्या सामान्य ज्ञान नियमांचे पालन करून प्रतिबंधित केले जाऊ शकते.

ज्वलनशील कचरा (तेलाने भिजवलेला कापूस कचरा, भंगार लाकूड, कागद इ.) चुकीच्या कोपऱ्यात जमा करणे आगीचा धोका आहे. असा कचरा कलेक्शन पॉईंट वरून काढला पाहिजे.

विद्युत उपकरणांना आग लागण्याचे कारण म्हणजे गैरवापर किंवा दुर्लक्ष. सैल कनेक्शन, चुकीचे रेट केलेले फ्यूज, ओव्हरलोड सर्किट्समुळे जास्त गरम होते ज्यामुळे आग लागू शकते. केबल्समधील कंडक्टरमधील इन्सुलेशनच्या नुकसानामुळे आग लागते.

कपडे आणि आग लागतील अशी कोणतीही वस्तू हीटरपासून दूर ठेवावी. कामकाजाच्या दिवसाच्या शेवटी हीटर बंद असल्याची खात्री करा.

अत्यंत ज्वलनशील द्रव आणि पेट्रोलियम मिश्रण (पातळ, चिकट द्रावण, सॉल्व्हेंट्स, केरोसीन, स्पिरिट, एलपीजी गॅस इ.) ज्वलनशील पदार्थांच्या साठवण क्षेत्रात साठवले पाहिजे.

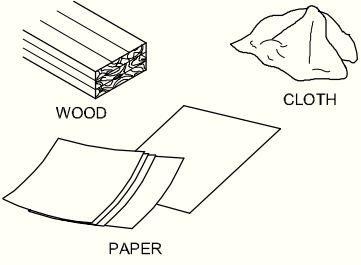
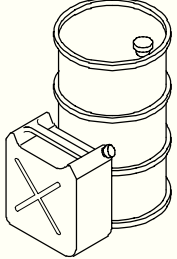
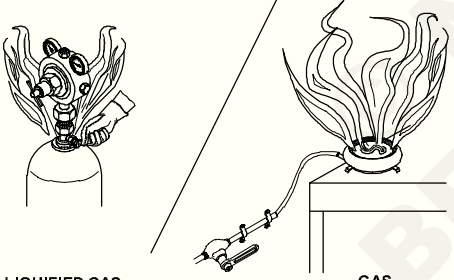

ब्लोल्म्प आणि टॉर्च वापरात नसताना ते चालू ठेवू नयेत.

**आग विझवणे:** इंधनाच्या स्वरूपानुसार आगीचे चार प्रकारात वर्गीकरण केले जाते.

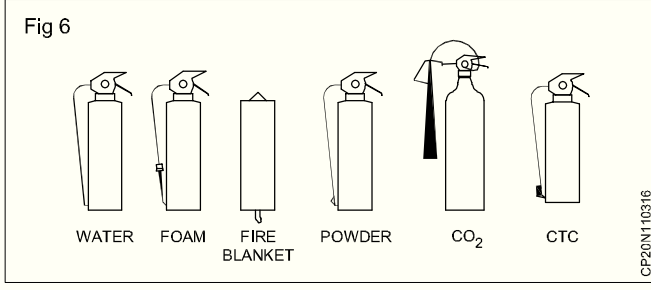
वेगवेगळ्या टाइप च्या आगींना (आकृती 2, आकृती 3 आकृती 4 आणि आकृती 5) वेगवेगळ्या मार्गांनी आणि वेगवेगळ्या विझविणाऱ्या एजंट्ससह सामोरे जावे लागते.

आग विझवण्यासाठी वापरली जाणारी कंटेंट किंवा पदार्थ म्हणजे आग विझवणारा एजंट आणि सामान्यतः (परंतु नेहमीच नाही) अग्निशामक यंत्रामध्ये आगीमध्ये फवारणी करण्यासाठी सोडण्याची यंत्रणा असते.

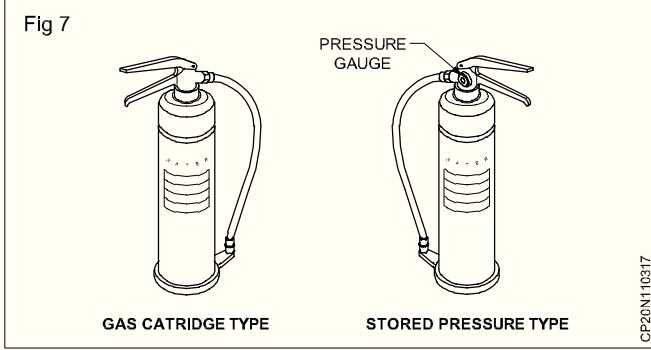
स्पेसिफिक टाइप ची आग विझवण्यासाठी योग्य टाइप चे एजंट जाणून घेणे महत्त्वाचे आहे; चुकीच्या एजंटचा वापर केल्याने गोष्टी आणखी वाईट होऊ शकतात. 'विद्युत आग' असे कोणतेही वर्गीकरण नाही, कारण ही केवळ वीज असते अशा कंटेंट मध्ये आग असते.

फ्युएल	एक्सटेनिग्विशर
<p>Fig 2</p> <p>CLASS 'A' FIRE</p>  <p>WOOD CLOTH PAPER</p> <p>CP20NT10312</p>	<p>सर्वात प्रभावी म्हणजे. पाण्याने थंड करणे. आगीच्या पायथ्याशी पाण्याचे जेट्स फवारावे आणि नंतर हळूहळू वरच्या दिशेने जावे.</p>
<p>Fig 3</p> <p>CLASS 'B' FIRE</p>  <p>FLAMMABLE LIQUIDS AND LIQUIFIABLE SOLIDS</p> <p>CP20NT10313</p>	<p>smothered पाहिजे. बर्निंग लिक्विडची संपूर्ण पृष्ठपार्त झाकणे हे उद्दीष्ट आहे. यामुळे आगीला ऑक्सिजनचा पुरवठा खंडित होण्याचा रिझल्ट होतो.</p> <p>जळत्या द्रवपदार्थावर कधीही पाणी वापरू नये.</p> <p>या टाइप च्या आगीवर फोम, कोरडी पावडर किंवा CO<sub>2</sub> वापरले जाऊ शकते.</p>
<p>Fig 4</p> <p>CLASS 'C' FIRE</p>  <p>LIQUIFIED GAS GAS</p> <p>CP20NT10314</p>	<p>द्रवीभूत वायूंचा सामना करताना अत्यंत सावधगिरी बाळगणे आवश्यक आहे. संपूर्ण परिसरात स्फोट होऊन अचानक आगीचा भडका उडण्याचा धोका आहे. सिलिंडरमधून भरलेल्या इक्विपमेंट्स ला आग लागल्यास - गॅसचा पुरवठा बंद करा. सर्वात सुरक्षित पाथ म्हणजे अलार्म वाजवणे आणि प्रशिक्षित कर्मचार्यांनी आग विझवणे.</p> <p>या टाइप च्या आगीवर कोरड्या पावडरचा वापर केला जातो.</p>
<p>Fig 5</p> <p>CLASS 'D' FIRE</p>  <p>METALS</p> <p>CP20NT10315</p>	<p>स्पेशल पावडर आता विकसित केली गेली आहेत जी या टाइप च्या आगीवर कंट्रोल ठेवण्यास आणि/किंवा विझविण्यास सक्षम आहेत.</p> <p>अग्निशामक एजंट्सची स्टॅण्डर्ड रेन्ज अपुरी किंवा धोक्याची असते जेव्हा धातूच्या आगींना सामोरे जावे लागते.</p> <p>विद्युत उपकरणांना आग.</p> <p>हॅलोन, कार्बन डाय ऑक्साईड, ड्राय पावडर आणि वाष्पीकरण द्रव (CTC) विझवण्याचे साधन विद्युत उपकरणांमध्ये आग हाताळण्यासाठी वापरले जाऊ शकतात. कोणत्याही परिस्थितीत विद्युत उपकरणांवर फोम किंवा द्रव (उदा. पाणी) विझविण्याचे साधन वापरले जाऊ नये.</p>

वेगवेगळ्या टाइप च्या आगींना तोंड देण्यासाठी विविध टाइप चे अग्निशामक 'एजंट' सबलब्ध आहेत. (पिक्चर 6)



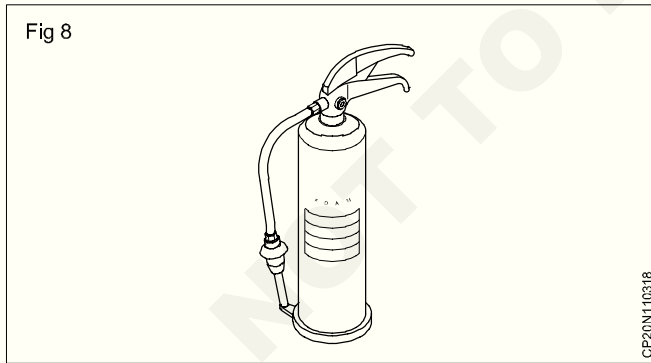
पाण्याने भरलेले विझविण्याचे साधन: ऑपरेशनच्या दोन मेथड्स आहेत (पिक्चर 7)



- गॅस कार्टेज टाइप
- संचयित दबाव टाइप

ऑपरेशनच्या दोन्ही पद्धतींसह आवश्यकतेनुसार डिस्चार्जमध्ये व्यत्यय आणला जाऊ शकतो, कंटेनर चे प्रोटेक्शन करणे आणि पाण्याचे अनावश्यक नुकसान टाळणे.

**फोम एक्टिंग्विशर (पिक्चर 8):** हे संचयित दाब किंवा गॅस कार्टेज टाइप चे असू शकतात. वापरण्यापूर्वी नेहमी एक्टिंग्विशरवरील ऑपरेटिंग सूचना तपासा.



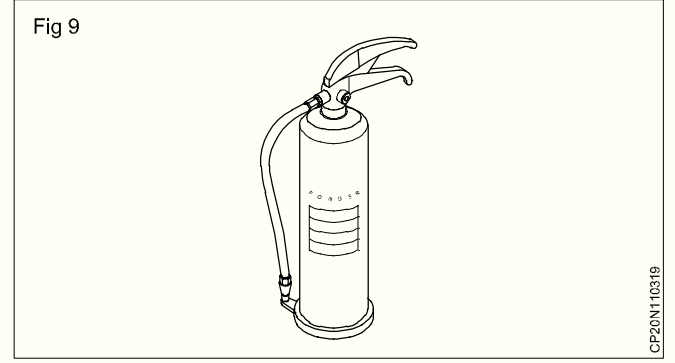
साठी सर्वात योग्य

- ज्वलनशील द्रव आग
- रनींग द्रव आग.

जेथे विद्युत उपकरणे गुंतलेली आहेत अशा आगींवर वापरू नये.

**कोरडी(झाय) पावडर एक्टिंग्विशर (पिक्चर 9):** झाय पावडरने बसवलेले एक्टिंग्विशर्स गॅस कार्टेज किंवा संचयित दाब टाइप चे असू शकतात.

देखावा आणि ऑपरेशनची मेथड्स पाण्याने भरलेल्या प्रमाणेच आहे. मुख्य वेगळे वैशिष्ट्य म्हणजे काट्याच्या आकाराचे नोजल. ड वर्गातील आगीचा सामना करण्यासाठी पावडर विकसित करण्यात आली आहे.



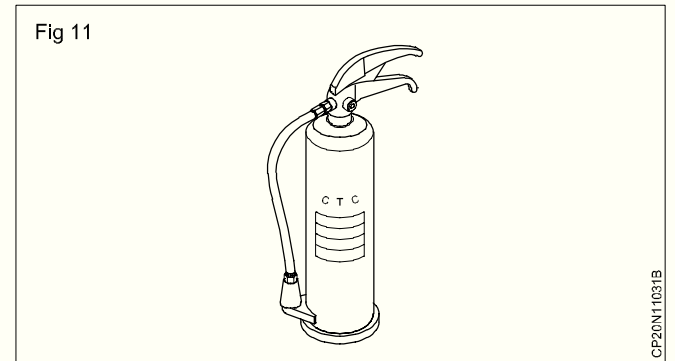
**कार्बन डायऑक्साइड (CO<sub>2</sub>):** स्पेसिफिक आकाराच्या डिस्चार्ज हॉर्नद्वारे हा टाइप सहजपणे ओळखला जातो. (पिक्चर 10)



क्लास ब आगीसाठी योग्य. करप्टेड होणे टाळले जाणे आवश्यक आहे तेथे सर्वात योग्य. खुल्या हवेत सामान्यतः प्रभावी नाही.

वापरण्यापूर्वी कंटेनरवरील ऑपरेटिंग सूचना नेहमी तपासा. ऑपरेशनच्या विविध गॅझेट्ससह सबलब्ध जसे की - प्लंजर, लीव्हर, ट्रिगर इ.

**Halon एक्टिंग्विशर(पिक्चर 11):** हे अग्निशामक कार्बन टेट्राक्लोराईड आणि ब्रोमोक्लोरोडिफ्लोरो मिथेन (BCF) ने भरलेले असू शकतात. ते एकतर गॅस कार्टेज किंवा संचयित दाब टाइप असू शकतात.



ते द्रव ओतणाऱ्या लहान आग विझवण्यात अधिक प्रभावी आहेत. ही विझविणारी यंत्रे स्पेशल तः विद्युत उपकरणांवर वापरण्यासाठी योग्य आणि सुरक्षित आहेत कारण रसायने विद्युतदृष्ट्या गैर-वाहक असतात.



या विझवणार्या यंत्रांनी सोडलेले धूर धोकादायक असतात, स्पेशल मर्यादित जागेत.

#### आग लागल्यास सामान्य प्रक्रिया:

- अलार्म वाजवा.
- सर्व यंत्रकॅटेंट आणि वीज (गॅस आणि वीज) बंद करा.
- दारे आणि खिडक्या बंद करा, परंतु त्यांना कुलूप किंवा बोल्ट करू नका. हे आगीला दिलेला ऑक्सिजन मर्यादित करेल आणि त्याचा प्रसार रोखेल.

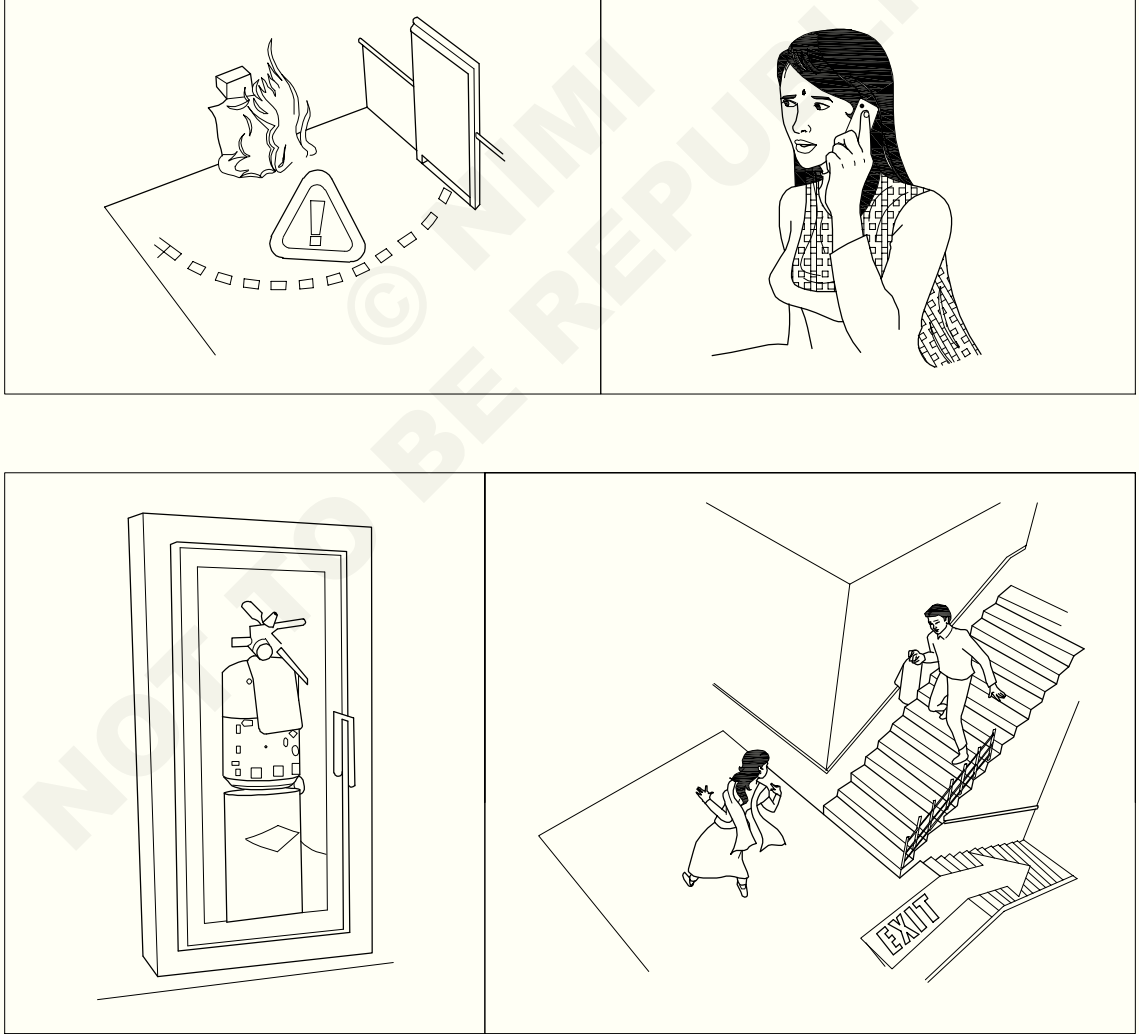
- तुम्ही सुरक्षितपणे करू शकत असल्यास आगीचा सामना करण्याचा प्रयत्न करा. अडकण्याचा धोका पत्करू नका.
- आगीशी लढण्यात सहभागी नसलेल्या कोणीही आपत्कालीन निर्गमन पाथ वापरून शांतपणे बाहेर पडावे आणि नियुक्त केलेल्या असेंब्ली पॉईंटवर जावे. हे करण्यात अयशस्वी झाल्याचा अर्थ असा होऊ शकतो की काही व्यक्ती बेहिशेबी आहे आणि इतरांना स्वतःला धोका पत्करून त्याचा सर्च घेण्याच्या त्रासात स्वतःला इन्सर्ट वे लागेल.

## अग्निशामक साधनांचा सराव करा (Practice on fire extinguishers)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- आगीच्या प्रकारानुसार अग्निशामक यंत्रांच्या निवडीबद्दल सांगा
- अग्निशामक यंत्रांच्या कार्यमेथड्स सांगा
- आग कशी विझवायची ते समजावून सांगा.

Fig 1



CP20N110321

प्रक्रिया (पिक्चर 1)

- आगीचे निरीक्षण करताना आग, आग, आग असे ओरडून आसपासच्या लोकांना सतर्क करा.
- अग्निशमन सेवेला कळवा किंवा ताबडतोब इनफार्मेशन देण्याची व्यवस्था करा.
- उघडी आणीबाणी अस्तित्वात आहे आणि त्यांना दूर जाण्यास सांगा.
- विद्युत वीज पुरवठा "बंद" करा.

लोकांना आगीच्या जवळ जाऊ देऊ नका

- विश्लेषण करा आणि आगीचा टाइप ओळखा. चार्ट 1 पहा.

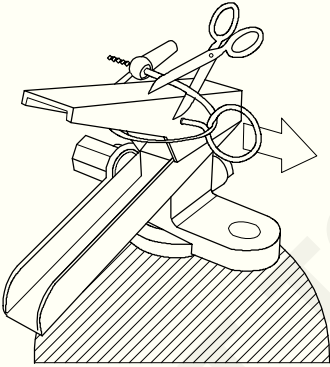
चार्ट-1

क्लास 'अ'	लाकूड, कागद, कापड, घन पदार्थ
क्लास 'बी'	तेल आधारित आग (ग्रीन, पेट्रोल, तेल) द्रवीकरणीय
वायू क्लास 'सी'	वायू आणि द्रवीकरणीय वायू
क्लास 'डी'	धातू आणि विद्युत उपकरणे

आग 'ब' आहे असे समजा; टाइप (ज्वलनशील द्रवरूप घन पदार्थ)

- CO<sub>2</sub> (कार्बन डाय ऑक्साईड) अग्निशामक यंत्र निवडा.
- शोधा आणि उचला, CO<sub>2</sub> अग्निशामक यंत्र. त्याच्या एक्सस्टेप डेटसाठी क्लिक करा.
- सील तोडा (पिक्चर 2)

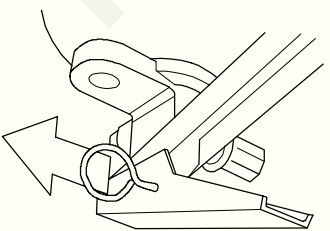
Fig 2



CP20N110322

- हँडलमधून सेफ्टी पिन खेचा (अग्निशामक यंत्राच्या शीर्षस्थानी असलेला पिन) (पिक्चर 3)

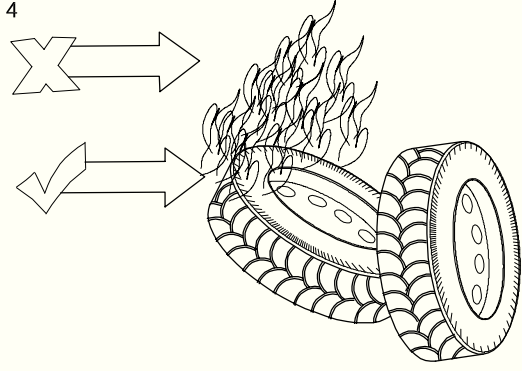
Fig 3



CP20N110323

- अग्निशामक नोजल किंवा रबरी नळीला आगीच्या तळाशी टागेट करा (हे इंधनाच्या आगीचा सोर्स काढून टाकेल) (पिक्चर 4)

Fig 4

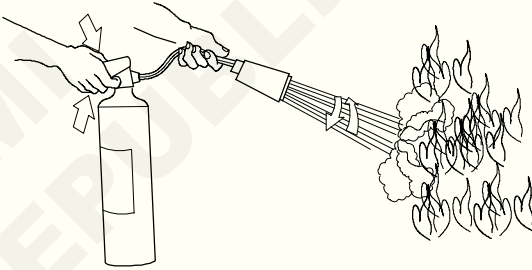


CP20N110324

स्वतःला कमी ठेवा

- एजंट डिस्चार्ज करण्यासाठी हँडल लीव्हर हळू हळू दाबा (पिक्चर 5)
- आग विझत नाही तोपर्यंत इंधनाच्या आगीवर अंदाजे 15 सेमी बाजूने स्वीप करा (पिक्चर 5)

Fig 5



CP20N110325

अग्निशामक यंत्रे दूरवरून वापरण्यासाठी तयार केली जातात.

खबरदारी

- आग विझवताना, आग भडकू शकते
- घबराट होऊ नका कारण ते त्वरित बंद होईल.
- तुम्ही अग्निशामक यंत्र वापरल्यानंतरही आग चांगला प्रतिसाद देत नसेल तर स्वतःहून दूर जा फायर पॉइंटपासून दूर.
- जिथे विषारी धूर निघत असेल तिथे आग विझवण्याचा प्रयत्न करू नका, व्यावसायिकांसाठी सोडा.
- लक्षात ठेवा की संपत्तीपेक्षा तुमचे जीवन महत्त्वाचे आहे. त्यामुळे स्वतःला किंवा इतरांना धोका देऊ नका.

आग विझवण्याचे साधे ऑपरेशन लक्षात ठेवण्यासाठी. लक्षात ठेवा P.A.S.S. हे तुम्हाला अग्निशामक यंत्र वापरण्यास मदत करेल.

P पुलासाठी P

A for ध्येय

S Squeeze साठी

S स्वीपसाठी

## संगणकाचा परिचय (Introduction to computers)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- संगणक परिभाषित आणि वर्गीकृत करा
- संगणकाचे फायदे आणि मर्यादांची यादी करा
- संगणकाच्या एप्लिकेशन ची यादी करा
- संगणकातील व्होल्टेज आणि करंट्सचे डिस्क्रिप्शन करा.

**संगणक - व्याख्या:** संगणक हे एक इलेक्ट्रॉनिक मशीन आहे, जे स्वतःच्या मेमरीमध्ये साठवलेल्या सूचनांच्या नियंत्रणाखाली कार्य करते जे डेटा (इनपुट) स्वीकारू शकते, स्पेसिफाइड नियमांनुसार डेटा हाताळू शकते (प्रक्रिया), रिझल्ट (आउटपुट) तयार करू शकते आणि भविष्यातील वापरासाठी रिझल्ट संचयित करू शकते. .

तांत्रिकदृष्ट्या, संगणक एक प्रोग्राम करण्यायोग्य मशीन आहे. याचा अर्थ ते प्रोग्राम केलेल्या सूचनांची लिस्ट कार्यान्वित करू शकते आणि दिलेल्या नवीन सूचनांना प्रतिसाद देऊ शकते.

### संगणकाचा इतिहास

#### चार्ल्स बॅबेजचे मशीन

आजच्या संगणकाच्या कार्याची तत्त्वे 1833 च्या सुमारास इंग्रजी गणितज्ञ चार्ल्स बॅबेज यांनी "विश्लेषणात्मक इंजिन" नावाच्या मशीनचा सर्च लावला. एक मशीन जे मर्यादित तंत्रांचा वापर करून फंक्शन्सच्या टेबलची गणना आणि प्रिंटेड करू शकते.

विश्लेषणात्मक इंजिनचे चार पार्ट होते. एक मिल, जो विपार्ट होता ज्याने गणना केली, मूलतः CPU; स्टोअर, इनफार्मेशन रेकॉर्ड ठेवली होती, मूलतः मेमरी; रीडर, जे पंच्ड कार्ड, मूलतः कीबोर्ड आणि प्रिंटर वापरून डेटा प्रविष्ट करण्यास अनुमती देईल.

म्हणून, चार्ल्स बॅबेज यांना पिक्चर 1 प्रमाणे "संगणकाचे जनक" मानले जाते.

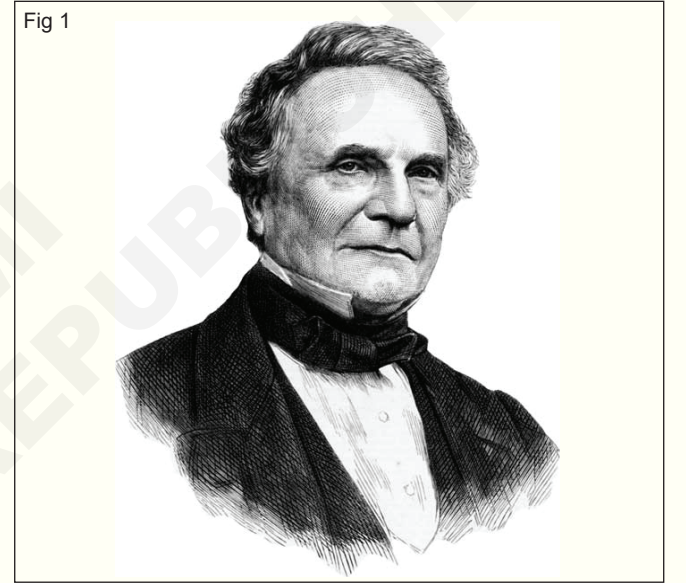
संगणकाच्या पिढ्यांचे वैशिष्ट्य एक प्रमुख तांत्रिक विकास आहे ज्याने बेसिक पणे संगणक चालवण्याच्या मेथड्स त बदल केला, परिणामी वाढत्या प्रमाणात लहान, स्वस्त, अधिक शक्तिशाली आणि अधिक कार्यक्षम आणि विश्वासार्ह उपकरणे. संगणकाच्या विविध पिढ्या खाली लिस्ट बद्ध केल्या आहेत:

**पहिली पिढी (1946-1954):** मध्ये 1946 इलेक्ट्रॉनिक व्हॉल्व्ह (व्हॅक्यूम ट्यूब) वापरणारे डिजिटल संगणक प्रथम पिढीचे संगणक म्हणून ओळखले जातात. इलेक्ट्रॉनिक व्हॉल्व्ह म्हणजेच व्हॅक्यूम ट्यूब वापरणारा पहिला 'संगणक'. व्हॅक्यूम ट्यूबच्या उच्च किमतीमुळे मुख्य मेमरीसाठी त्यांचा वापर रोखला गेला. त्यांनी ध्वनी लहरींचा प्रसार करण्याच्या स्वरूपात इनफार्मेशन स्टोर केली.

व्हॅक्यूम ट्यूब खूप वीज वापरते. हे संगणक आकाराने मोठे होते आणि त्यावर प्रोग्रॅम लिहिणे अवघड होते. या पिढीतील काही संगणक असे:

**मार्क I:** IBM ऑटोमॅटिक सिक्वेन्स कंट्रोल कॅल्क्युलेटर (ASCC), हार्वर्ड विद्यापीठाने मार्क I. म्हटले, एक होते इलेक्ट्रो-मेकॅनिकल संगणक. ची दीर्घ सर्विस यशस्वीरित्या पार पाडणारी मार्क I. ही पहिली मशीन आहे अर्थमेटिक आणि लॉजिकल ऑपरेशन. मार्क I. आहे पहिल्या पिढीचा संगणक.

Fig 1



**ENIAC:** जॉन एकर्ट आणि जॉन माउची यांनी 1946 मध्ये पेनसिल्व्हेनिया विद्यापीठ, यूएसए येथे तयार केलेला हा पहिला इलेक्ट्रॉनिक संगणक होता. त्याला इलेक्ट्रॉनिक न्यूमेरिकल इंटिग्रेटर आणि कॅल्क्युलेटर (ENIAC) असे नाव देण्यात आले. ENIAC 30-50 फूट लांब, 30 टन वजनाचे होते, त्यात 18,000 व्हॅक्यूम ट्यूब, 70,000 प्रतिरोधक, 10,000 कॅपेसिटर आणि 150,000 वॉट विजेची आवश्यकता होती. आज संगणक ENIAC पेक्षा कितीतरी पटीने शक्तिशाली आहे, तरीही आकार खूपच लहान आहे.

**EDVAC:** याचा अर्थ Electronic Discrete Variable Automatic Computer असा आहे आणि तो 1950 मध्ये विकसित करण्यात आला. संगणकामध्ये डेटा आणि सूचना साठवण्याची संकल्पना येथे मांडण्यात आली. संगणकाला डेटा आणि सूचना या दोन्हीमध्ये जलद एक्सेस असल्याने यामुळे अधिक जलद ऑपरेशन होऊ शकले. सूचना स्टोर करण्याचा दुसरा फायदा म्हणजे संगणक इंटर्नल लॉजिकल निर्णय घेऊ शकतो. EDVAC हा बायनरी सिरीयल संगणक होता ज्यामध्ये ऑटोमॅटिक बेरीज, वजाबाकी, गुणाकार, प्रोग्राम केलेले भागाकार आणि अल्ट्रासोनिक सिरीयल मेमरीसह ऑटोमॅटिक तपासणी होते.

**EDSAC:** याचा अर्थ आहे इलेक्ट्रॉनिक विलंब स्टोरेज ऑटोमॅटिक संगणक आणि द्वारे विकसित केले गेले एम.व्ही. विल्क्स 1949 मध्ये केंब्रिज विद्यापीठात. EDSAC पहिले आहे स्टोर-प्रोग्राम संगणक. EDSAC ने तीन मिलीसेकंद रेन्ज मध्ये गणना केली. त्याने मानवी हस्तक्षेपाशिवाय अर्थमेटिक आणि लॉजिकल ऑपरेशन्स केले. यशाची गुरुकिल्ली स्टोर सूचनांमध्ये होती ज्यावर ती केवळ त्याच्या ऑपरेशनसाठी अवलंबून होती.

### या यंत्राने संगणक युगाची सुरुवात केली.

**UNIVAC-1:** याचा अर्थ युनिव्हर्सल ऑटोमॅटिक संगणक आहे आणि 1951 मध्ये युनायटेड स्टेट्‌सने विकसित केलेला हा पहिला व्यावसायिक संगणक होता. या मशीनची लांबी 25 फूट बाय 50 फूट होती, त्यात 5,600 ट्यूब, 18,000 क्रिस्टल डायोड आणि 300 रिले होते. याने सिरीयल सर्किटरी, 2.25 मेगाहर्ट्झ बिट रेटचा वापर केला आणि इंटरनल स्टोरेज क्षमता 1,000 शब्द किंवा 12,000 अक्षरे होती.

UNIVAC चा वापर करण्यात आला सामान्य उद्देश संगणन मोठ्या प्रमाणात इनपुट आणि आउटपुटसह. UNIVAC हा चुंबकीय टेप युनिटसह सुसज्ज असलेला पहिला संगणक होता आणि बफर मेमरी वापरणारा पहिला संगणक होता.

### पहिल्या पिढीच्या संगणकाच्या मर्यादा

पहिल्या पिढीतील संगणकांचे प्रमुख दोष खालीलप्रमाणे आहेत.

- त्यांनी त्यांचे मुख्य इलेक्ट्रॉनिक एलिमेंट म्हणून वाल्व किंवा व्हॅक्यूम ट्यूबचा वापर केला.
- ते आकाराने मोठे होते, प्रक्रियेत मंद होते आणि साठवण क्षमता कमी होती.
- त्यांनी भरपूर वीज वापरली आणि भरपूर उष्णता निर्माण केली.
- त्यांची संगणकीय क्षमता मर्यादित होती.
- ते इतके अचूक आणि विश्वासार्ह नव्हते.
- त्यांनी प्रोग्रामिंगसाठी मशीन लेव्हलची भाषा वापरली.
- ते खूप महाग होते.

**दुसरी पिढी (1955-1964):** दुसऱ्या पिढीचा संगणक वापरला ट्रान्झिस्टर CPU एलिमेंटसाठी आणि मुख्य मेमरी आणि चुंबकीय डिस्कसाठी फेराइट कोर दुय्यम स्मृती साठी. यांसारख्या उच्च-स्तरीय भाषांचा वापर केला FORTRAN (1956), ALGOL (1960) आणि COBOL (1960 - 1961)). इनपुट आउटपुट (I/O) प्रोसेसर I/O ऑपरेशन्स कंट्रोल करण्यासाठी समाविष्ट केले होते.

उत्पादन खर्चही खूप कमी होता. त्यामुळे संगणकाचा आकार खूपच कमी झाला.

दुसऱ्या पिढीमध्ये सेंट्रल प्रोसेसिंग युनिट (CPU), मेमरी, प्रोग्रामिंग भाषा आणि इनपुट आणि आउटपुट युनिट्सची संकल्पना विकसित झाली. दुसऱ्या पिढीतील काही संगणक म्हणजे IBM 1620, IBM 1401, CDC 3600.

### वैशिष्ट्ये

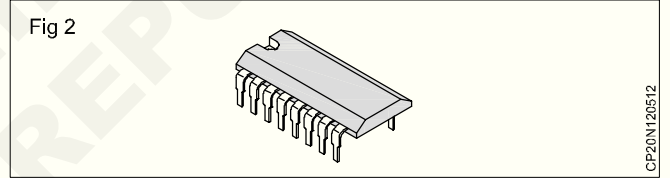
- व्हॅक्यूम ट्यूबऐवजी ट्रान्झिस्टर वापरण्यात आले.
- प्रक्रियेचा वेग पहिल्या पिढीतील संगणकांपेक्षा वेगवान आहे (मायक्रो सेकंद)
- आकाराने लहान (51 चौरस फूट)
- इनपुट आणि आउटपुट साधने जलद होती.

**तिसरी पिढी (1964-1977):** 300 ट्रान्झिस्टर क्षमता होणारी एक लहान चिप विकास करून. ही इंटीग्रेटेड सर्किट्स (IC) म्हणून प्रसिद्ध आहेत त्यालाच चिप्स म्हणतात.

सिंगल IC मध्ये सिलिकॉनच्या एका पातळ तुकड्यावर अनेक ट्रान्झिस्टर, रजिस्टर्स आणि कॅपेसिटर बनवलेले असतात. त्यामुळे संगणकाचा आकार आणखी कमी झाला हे अगदी उघड आहे. या काळात विकसित झालेले काही संगणक होते IBM-360, ICL-1900, IBM 370, आणि VAX-750. उच्च स्तरीय भाषा जसे की बेसिक (नवशिक्यांसाठी सर्व उद्देश प्रतीकात्मक सूचना कोड) या काळात विकसित झाले.

या पिढीचे संगणक आकाराने लहान, कमी किमतीचे, मोठी मेमरी आणि प्रक्रियेचा वेग खूप जास्त आहे. लवकरच ICs ने बदलले LSI (लार्ज स्केल इंटीग्रेशन), ज्यामध्ये सुमारे 100 एलिमेंट होते.

सुमारे 100 एलिमेंट असलेल्या IC ला LSI म्हणतात (पिक्चर 2).



### वैशिष्ट्ये

- त्यांनी ट्रान्झिस्टरच्या जागी इंटीग्रेटेड सर्किट (IC) चिप्स वापरल्या. • सेमीकंडक्टर मेमरी उपकरणे वापरली गेली.
- आकार मोठ्या प्रमाणात कमी झाला होता, प्रक्रियेची गती जास्त होती आणि ते अधिक अचूक आणि विश्वासार्ह होते.
- लार्ज स्केल इंटीग्रेशन (LSI) आणि व्हेरी लार्ज स्केल इंटीग्रेशन (VLSI) देखील विकसित केले गेले.
- या पिढीमध्ये मिनी कॉम्प्युटरची ओळख झाली.
- त्यांनी प्रोग्रामिंगसाठी उच्च स्तरीय भाषा वापरली.

**चौथी पिढी (1978 - सध्या):** सुमारे 100 एलिमेंट असलेल्या IC ला LSI (लार्ज स्केल इंटीग्रेशन) म्हणतात आणि एक, ज्यामध्ये असे 1000 पेक्षा जास्त एलिमेंट असतात, त्याला VLSI (व्हेरी लार्ज स्केल इंटीग्रेशन) म्हणतात.

हे मायक्रोप्रोसेसर नावाच्या सिंगल सिलिकॉन चिपवर तयार केलेल्या मोठ्या प्रमाणात इंटीग्रेटेड सर्किट्स (LSIC) वापरते. मायक्रोप्रोसेसरच्या विकासामुळे संगणकाचे सेंट्रल प्रोसेसिंग युनिट (सीपीयू) सिंगल चिपवर ठेवणे शक्य झाले आहे. या संगणकांना मायक्रो संगणक म्हणतात.

नंतर खूप मोठ्या प्रमाणात इंटीग्रेटेड सर्किट्स (VLSIC) ने LSIC ची जागा घेतली. अशा प्रकारे पूर्वीच्या काळी खूप मोठी खोली व्यापणारा संगणक आता टेबलवर ठेवता येतो. तुम्ही तुमच्या शाळेत पाहत असलेला वैयक्तिक संगणक (PC) हा फोर्थ जनरेशन कॉम्प्युटर मेन मेमरी आहे ज्यामध्ये 4 M बिट आकारापर्यंत जलद सेमीकंडक्टर चिप्स वापरल्या जातात. चौथ्या पिढीतील काही संगणक म्हणजे IBM PC, AppleMacintosh, इ.

हार्ड डिस्क दुय्यम मेमरी म्हणून वापरली गेली. कीबोर्ड, डॉट मॅट्रिक्स प्रिंटर इत्यादी विकसित केले. ऑपरेटिंग सिस्टम (OS)-जसे की MS-DOS, UNIX, Apple चे Macintosh सबलब्ध होते. ऑब्जेक्ट ओरिएंटेड भाषा, C++ इत्यादी विकसित केले गेले.

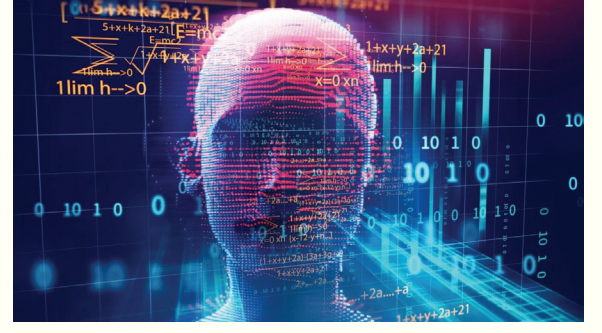
### वैशिष्ट्ये

- त्यांनी त्यांचे मुख्य स्विचिंग एलिमेंट म्हणून मायक्रोप्रोसेसर (VLSI) वापरले.
- त्यांना मायक्रो संगणक किंवा वैयक्तिक संगणक असेही म्हणतात.
- त्यांचा आकार डेस्कटॉपपासून लॅपटॉप किंवा पामटॉपपर्यंत बदलतो.
- त्यांच्याकडे प्रक्रिया करण्याची गती खूप जास्त आहे; ते 100% अचूक, विश्वासार्ह, मेहनती आणि बहुमुखी आहेत.
- त्यांची साठवण क्षमता खूप मोठी आहे.

**पाचवी पिढी (करंट आणि भविष्य):** ५ व्या पिढीचे संगणक ULSI (अल्ट्रा-लार्ज स्केल इंटीग्रेशन) चिप्स वापरतात. ULSI चिप्समध्ये लाखो ट्रांझिस्टर एकाच IC मध्ये ठेवलेले असतात.

या कालावधीत 64 बिट मायक्रोप्रोसेसर विकसित करण्यात आले आहेत.. मेमरी चिप्स आणि 1 GB पर्यंत फ्लॅश मेमरी, 600 GB पर्यंत हार्ड डिस्क आणि 50 GB पर्यंत ऑप्टिकल डिस्क विकसित करण्यात आली आहेत. (पिक्चर 3)

Fig 3



आर्टिफिशियल इंटेलिजेंसवर आधारित पाचव्या पिढीतील संगणकीय उपकरणे अजूनही विकासात आहेत, जरी काही ॲप्लिकेशन्स आहेत, जसे की व्हॉइस रेकग्निशन, जे आज वापरले जात आहेत.

आर्टिफिशियल इंटेलिजन्स ही संगणक विज्ञानाची शाखा आहे जी संगणकांना मानवासारखे वागणूक देण्याशी संबंधित आहे. मॅसॅच्युसेट्स इन्स्टिट्यूट ऑफ टेक्नॉलॉजीमध्ये जॉन मॅककार्थी यांनी 1956 मध्ये हा शब्द तयार केला होता. कृत्रिम बुद्धिमत्तेमध्ये हे समाविष्ट आहे:

- **खेळ खेळणे:** बुद्धिबळ आणि चेकर्ससारखे गेम खेळण्यासाठी प्रोग्रामिंग संगणक
- **तज्ञ सिस्टिम :** वास्तविक जीवनातील परिस्थितींमध्ये निर्णय घेण्यासाठी प्रोग्रामिंग संगणक (उदाहरणार्थ, काही तज्ञ सिस्टिम डॉक्टरांना लक्षणांवर आधारित रोगांचे निदान करण्यात मदत करतात)
- **नैसर्गिक भाषा:** नैसर्गिक मानवी भाषा समजून घेण्यासाठी संगणक प्रोग्रामिंग
- **न्यूरल नेटवर्क:** प्राण्यांच्या मेंदूमध्ये होणाऱ्या शारीरिक संबंधांचे पुनरुत्पादन करण्याचा प्रयत्न करून बुद्धिमत्तेचे अनुकरण करणाऱ्या सिस्टिम
- **रोबोटिक्स:** इतर संवेदी उत्तेजनांना पाहण्यासाठी आणि ऐकण्यासाठी आणि प्रतिक्रिया देण्यासाठी प्रोग्रामिंग संगणक

पिढी	इलेक्ट्रॉनिक घटक	फायदे	तोटे
पहिला	व्हॅक्यूम ट्यूब आणि संगणकीय कार्य	गणना करण्यात मदत केली	1 मोठा आकार 2 खूप महाग 3 मंद गती 4 कमी अचूकता 5 कमी स्टोरेज 6 उच्च शक्ती आवश्यकता 7 उच्च उष्णता निर्मिती 8 उच्च अपयश दर 9 मशीन भाषा वापरली 10 कोणतीही ऑपरेटिंग सिस्टम नाही

दुसरा	ट्रान्झिस्टर	1 लहान आकार 2 कमी खर्च 3 चांगला वेग 4 कमी शक्ती वापर आणि कमीउष्णता पिढी स्टेटमेंट सभा भाषा 5 चांगली साठवण क्षमता 6 उत्तम अचूकता आणि अधिक विश्वासार्हता	1 हवेची गरज आहे कंडिशनिंग 2 स्थिर देखभाल 3 चालत नाही सिस्टिम 4 नंतरचा टप्पा संगणक वापरले स्टेटमेंट सभा भाषा
तिसर्या	इंटीग्रेटेड सर्किट (IC) लहान आणि मध्यम प्रमाणात	1 सर्वात चांगले । आणि ॥ च्या तुलनेत पैलू 2 वापरलेली ऑपरेटिंग सिस्टिम आणि उच्च स्तरीय भाषा	1 स्टार्टिंग समस्या उत्पादकांसह 2 अंतर्दृष्टी नाही मध्ये प्राप्त इंटरनल काम
चौथा	VLSI किंवा मायक्रोप्रोसेसर	1 कमी खर्च 2 उत्कृष्ट वेग आणि विश्वासार्हता 3 माणसाच्या जवळचे संगणक	1 कमी शक्तिशाली मुख्य पेक्षा फ्रेम संगणक
पाचवे (ज्ञान आ व श्य क इनफार्मेशन प्रक्रिया)	सिस्टिम) ULSI किंवा बायो-चिप्स	1 खूप स्वस्त 2 सुपर स्पीड 3 खूप उच्च साठवण क्षमता 4 उच्च अत्याधुनिक OS 5 बुद्धिमत्ता आहे आणि निर्णय घेण्याची क्षमता	1 नवीन निम्न पातळी भाषा आवश्यक

#### संगणकांचे वर्गीकरण

खालील निकषांनुसार संगणकांचे वर्गीकरण केले जाते:

- ऑपरेशनचे तत्त्व
- संगणकीय शक्ती, मेमरी क्षमता आणि खर्च
- तांत्रिक विकास
- ऑपरेशनचे तत्त्व
  - a एक अॅनालॉग संगणक
  - b डिजिटल संगणक
  - c हायब्रीड संगणक

#### अॅनालॉग संगणक

हा एक संगणक आहे जो सतत बदलणारे फिजिकल प्रमाण जसे की करंट, तापमान, दाब इ. मोजतो आणि गणनेसाठी डेटा म्हणून वापरला जाऊ शकतो अशा प्रमाणात रूपांतरित करतो. हे संगणक सतत वेगवेगळ्या प्रमाणात हाताळतात म्हणून ते फक्त अंदाजे रिझल्ट देतील. त्याचे आउटपुट सहसा मीटर किंवा स्केलवर प्रदर्शित केले जाते. अॅनालॉग कॉम्प्युटरमध्ये कमी मेमरी आणि कमी फंक्शन्स असतात. हे अभियांत्रिकी आणि वैज्ञानिक एप्लिकेशन साठी वापरले जातात.

- थर्मामीटर
- स्पीडोमीटर
- अॅनालॉग घड्याळ

#### डिजिटल संगणक

डिजिटल संगणक डिजिटल डेटासह कार्य करतो. डिजिटल संगणक बायनरी क्रमांक सिस्टिम वापरतो. बायनरी नंबर सिस्टिम मध्ये फक्त दोन अंक '0' आणि '1' असतात. डिजिटल संगणक डिजिटल सिग्नलमध्ये डेटाचे प्रतिनिधित्व करतो. A '0' OFF चे प्रतिनिधित्व करतो आणि '1' ON चे प्रतिनिधित्व करतो. डिजिटल संगणक डेटावर अर्थमेटिक आणि लॉजिकल ऑपरेशन्स करतो. हे डिजिटल स्वरूपात आउटपुट देते.

डिजिटल संगणक खूप वेगवान आहेत. हे संगणक रिझल्ट स्टोर करू शकतात. त्यांच्याकडे मोठी मेमरी आहे (म्हणजे डेटा साठवण्याची क्षमता). आज ऑफिस आणि घरांमध्ये वापरले जाणारे बहुतेक संगणक डिजिटल संगणक आहेत.

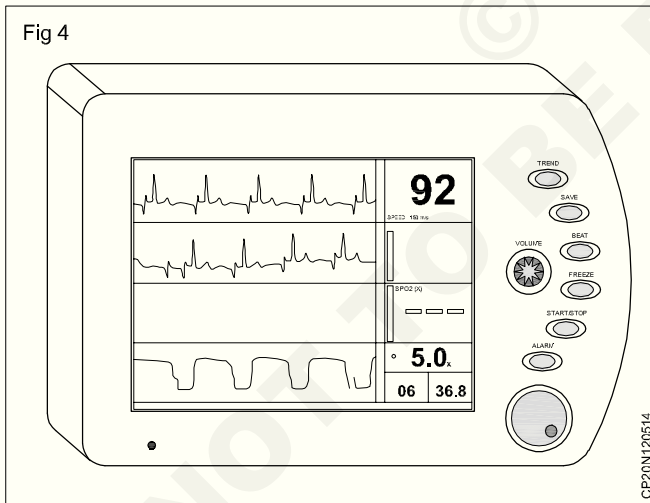
डिजिटल संगणक पुढील दोन गटांमध्ये विपार्ट ले गेले आहेत:

- स्पेशल उद्देशाचे संगणक
- सामान्य उद्देश संगणक

अॅनालॉग संगणक	डिजिटल संगणक
1 अॅनालॉग संगणक सतत व्हॅल्यु वर कार्य करतात.	1 डिजिटल संगणक स्वतंत्र व्हॅल्यु वर कार्य करतात.
2 अॅनालॉग कॉम्प्युटरमध्ये कमी मेमरी असते.	2 डिजिटल कॉम्प्युटरमध्ये खूप मोठी मेमरी असते
3 अॅनालॉग संगणकांचा वेग कमी असतो.	3 डिजिटल कॉम्प्युटरचा वेग वेगवान आहे.
4 एनालॉग संगणक कमी विश्वासार्ह आहेत.	4 डिजिटल संगणक अधिक विश्वासार्ह आहेत.
5 अॅनालॉग संगणक. अभियांत्रिकी विज्ञान आणि वैद्यकीय क्षेत्रात वापरले जाणारे	5 जीवनाच्या सर्व क्षेत्रात डिजिटल संगणक वापरला जातो. विज्ञान आणि वैद्यकीय एरिया .
6 अॅनालॉग संगणकांचा वापर वेग आणि तापमान यांसारख्या अॅनालॉग प्रमाणांची कॅल्क्युलेशन / मोजण्यासाठी केला जातो.	6 गणितीय आणि लॉजिकल ऑपरेशन्सची गणना करण्यासाठी डिजिटल संगणकांचा वापर केला जातो. हे बेरीज, वजाबाकी, भागाकार, गुणाकार आणि इतर गणिती आणि सांख्यिकीय क्रिया सोडवू शकते
7 एनालॉग संगणक कमी अचूक रिझल्ट देतात.	7 डिजिटल संगणक 100% अचूक रिझल्ट प्रदान करतात.
8 सामान्यतः अॅनालॉग संगणक स्पेसिफिक उद्देशाने असतात	8 डिजिटल संगणक हे सामान्य हेतू आहेत.
9 सामान्यतः अॅनालॉग संगणक स्पेसिफिक उद्देशाने असतात	9 डिजिटल संगणक हे सामान्य हेतू आहेत.
10 अॅनालॉग संगणकांची उदाहरणे आहेत: थर्मामीटर, अॅनालॉग घड्याळ, स्पीडोमीटर इ.	10 डिजिटल संगणकांची उदाहरणे आहेत: वैयक्तिक संगणक, लॅपटॉप, स्मार्ट फोन इ.

### स्पेशल उद्देश संगणक

हा एक संगणक आहे जो स्पेसिफिक टाइप च्या समस्यांचे निराकरण करण्यासाठी डिझाइन केलेला आहे. जहाजे आणि विमानांमध्ये वापरले जाणारे संगणक इ.



### सामान्य उद्देश संगणक

हा एक संगणक आहे जो विविध टाइप च्या समस्यांचे निराकरण करण्यासाठी डिझाइन केलेला आहे, सामान्य उद्देश संगणक विविध प्रोग्राम संचयित करू शकतो आणि त्यावर प्रक्रिया करू शकतो.

एनालॉग आणि डिजिटल संगणकांमधील फरक टेबल 2 मध्ये लिस्ट बद्ध आहेत हायब्रीड संगणक(पिव्चर 4)

हायब्रीड कॉम्प्युटर हे अॅनालॉग आणि डिजिटल दोन्ही कॉम्प्युटरचे संयोजन आहे. हायब्रीड संगणक अॅनालॉग आणि डिजिटल दोन्ही डेटा

हाताळू शकतो. हायब्रीड कॉम्प्युटरमध्ये अॅनालॉग आणि डिजिटल दोन्ही कॉम्प्युटरची सर्वोत्कृष्ट वैशिष्ट्ये एकत्र केली जातात. हे अॅनालॉग आणि डिजिटल दोन्ही स्वरूपात डेटा स्वीकारू शकते.

### एप्लिकेशन

हॉस्पिटलमध्ये हायब्रीड कॉम्प्युटर उपकरणे वापरली जातात जी रुग्णाच्या हृदयाचे कार्य, तापमान आणि रक्तदाब इत्यादी मोजू शकतात. ही गणना नंबर मध्ये रूपांतरित केली जाऊ शकते आणि डिजिटल स्वरूपात दर्शविली जाऊ शकते. उदाहरणार्थ, द व्हिटल साइन्स मॉनिटरिंग युनिट याला थोडक्यात (VSM) देखील म्हणतात. यात ब्लड प्रेशर मॉनिटर, ईसीजी मॉनिटर, रेस्पिरेटरी मॉनिटर आहे आणि ऍनेस्थेसियाचे निरीक्षण करण्यासाठी देखील वापरले जाते.

- संकरित संगणक स्पेसिफिप आणि क्षेपणास्त्र सिस्टिम मध्ये देखील वापरले जातात.
- हायब्रीड कॉम्प्युटर मशीन्सचा वापर सामान्यतः वैज्ञानिक एप्लिकेशन मध्ये केला जातो
- हायब्रीड संगणक औद्योगिक प्रक्रिया कंट्रोल करण्यासाठी वापरले जातात.

कॉम्प्युटिंग पॉवर, मेमरी क्षमता आणि किमतीच्या आधारे संगणकांचे वर्गीकरण केले जाते.

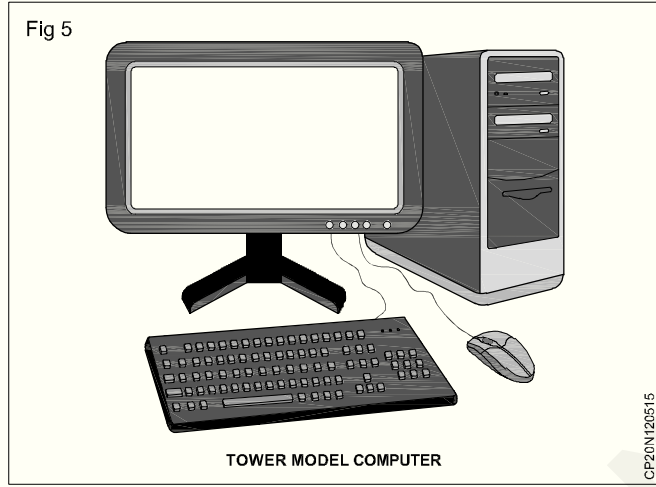
- मायक्रो कॉम्प्युटर किंवा पर्सनल कॉम्प्युटर.
- मिनी संगणक.
- मेनफ्रेम संगणक.
- सुपर कॉम्प्युटर.

## मायक्रो संगणक

मायक्रो कॉम्प्युटरला वैयक्तिक संगणक असेही म्हणतात. हे 1970 मध्ये सादर केले गेले. पीसी आणि ऍपल मॅकिंटॉश ही वैयक्तिक संगणकांची उदाहरणे आहेत. वैयक्तिक संगणकांचे प्रमुख टाइप म्हणजे डेस्कटॉप संगणक आणि पोर्टेबल संगणक.

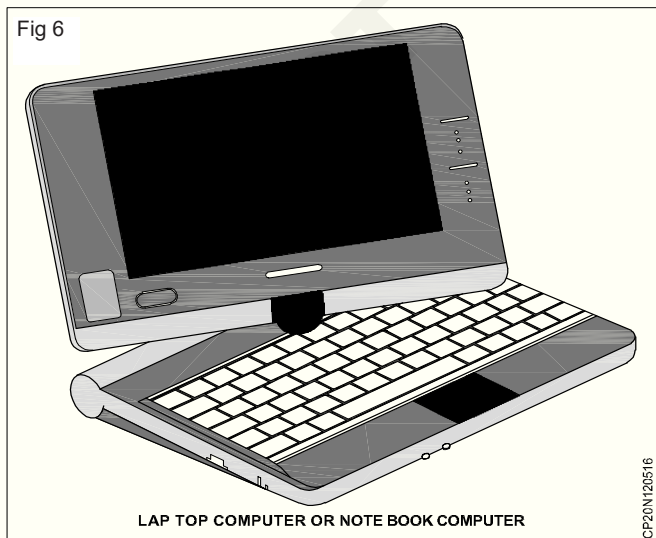
## डेस्कटॉप संगणक

हे संगणक टेबल किंवा डेस्कटॉपवर सहजपणे बसू शकतात, म्हणून हे नाव. हे संगणक दोन मॉडेल्स किंवा केसिंगमध्ये येतात. डेस्कटॉप मॉडेलमध्ये, सिस्टम युनिट डेस्कटॉप किंवा टेबलवर ठेवली जाते. सिस्टम युनिटवर मॉनिटर ठेवला आहे. टॉवर मॉडेलमध्ये, मॉनिटर आणि सिस्टम युनिट दोन्ही टेबलवर (पिक्चर 5) प्रमाणे ठेवलेले आहेत.



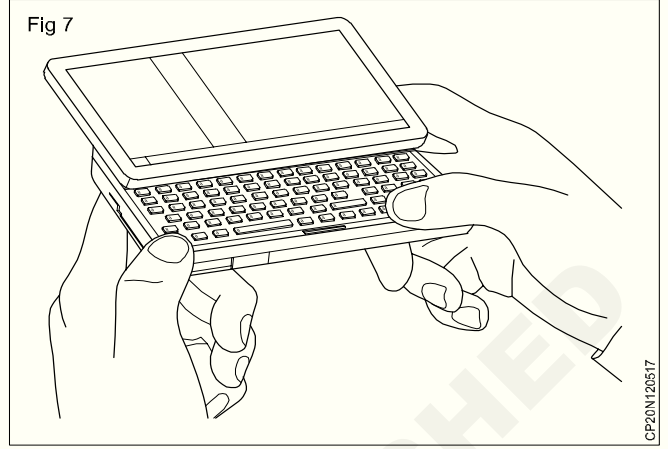
**पोर्टेबल संगणक:** पोर्टेबल एक वैयक्तिक संगणक आहे जो एका ठिकाणाहून दुसऱ्या ठिकाणी सहज वाहून नेला जाऊ शकतो. नोटबुक संगणक आणि हँडहेल्ड संगणक (स्मार्ट फोन) ही पोर्टेबल संगणकांची उदाहरणे आहेत. नोटबुक संगणकाला लॅपटॉप संगणक असेही म्हणतात. लॅपटॉप कॉम्प्युटर आकाराने खूप लहान असतात आणि ते मांडीवर सहज ठेवता येतात.

**लॅप टॉप संगणक किंवा नोटबुक संगणक:** लॅपटॉप संगणक किंवा नोटबुक संगणक आकृती 6 आणि आकृती 7 मध्ये दर्शविल्याप्रमाणे असेल. त्याला टॉवर मॉडेल संगणक असेही म्हणतात.



## पामटॉप संगणक/डिजिटल डायरी/नोटबुक/पीडीए

हातातील संगणक (स्मार्ट फोनसारखा) देखील पोर्टेबल आहे. हाताने पकडलेला संगणक पामटॉप संगणक म्हणून ओळखला जातो. पामटॉपमध्ये कीबोर्ड नसतो परंतु स्क्रीन इनपुट आणि आउटपुट डिव्हाइस दोन्ही म्हणून काम करते. हे यूजर च्या हातात सहज बसते.



## मायक्रो कॉम्प्युटरचा वापर

पीसी हा ऑफिसमध्ये वापरला जाणारा संगणकाचा सर्वात सामान्य टाइप आहे. आता अनेक घरांमध्ये याचा सर्वास वापर केला जातो. हे व्यवसाय आणि अभियांत्रिकी अनुप्रयोगासाठी देखील वापरले जातात.

## मिनी संगणक

1960 च्या दशकात मिनी कॉम्प्युटरची सुरुवात झाली. लघुसंगणक हा वैयक्तिक संगणकापेक्षा मोठा आणि अधिक शक्तिशाली संगणक आहे. ते प्रति सेकंद पाच दशलक्ष सूचना कार्यान्वित करू शकते. यात साधारणपणे दोन किंवा अधिक प्रोसेसर असतात.

मिनी कॉम्प्युटर 4000 कनेक्टेड वापरकर्त्यांना एकाच वेळी सर्विस देऊ शकतो. हे सामान्यतः वापरकर्त्यांद्वारे वैयक्तिक संगणक किंवा टर्मिनलद्वारे एक्सेस केले जाते. मॉनिटर आणि कीबोर्ड असलेल्या इन्टिग्रेटेड ला टर्मिनल म्हणतात. त्याला डंब टर्मिनल असेही म्हणतात. यात प्रक्रिया करण्याची शक्ती नाही आणि तो स्वतंत्र संगणक म्हणून काम करू शकत नाही. VAX-8800, AS400 हे काही लघुसंगणक मॉडेल्स आहेत

## मिनी कॉम्प्युटरचा वापर

इनफार्मेशन चे सेंट्रलाइज्ड स्टोअर प्रदान करण्यासाठी लहान आणि मध्यम आकाराच्या कंपन्यांद्वारे मिनी संगणकांचा वापर केला जातो.

## मेनफ्रेम संगणक

मेनफ्रेम संगणक 1975 मध्ये सादर केले गेले. मेनफ्रेम संगणक हा आकाराने खूप मोठा संगणक आहे. हे प्रोसेसर आहे. हे एकाच वेळी अनेक वापरकर्त्यांसाठी मल्टिपल फंक्शन करण्यासाठी डिझाइन केलेले आहे. मेनफ्रेम संगणक एकाच वेळी 50,000 वापरकर्त्यांना सर्विस देऊ शकतात.

वापरकर्ते टर्मिनल किंवा वैयक्तिक संगणकाद्वारे मेनफ्रेम संगणकावर एक्सेस करतात. एक सामान्य मेनफ्रेम संगणक प्रति सेकंद 16 दशलक्ष सूचना कार्यान्वित करू शकतो. काही मुख्य संगणक मॉडेल्स आहेत



- NEC 610
- डिसेंबर १०

### मेनफ्रेम संगणकाचा वापर

मेनफ्रेम संगणक प्रामुख्याने कॉर्पोरेट आणि सरकारी संस्थांद्वारे गंभीर एप्लिकेशन साठी, मोठ्या प्रमाणात डेटा प्रक्रिया जसे की संवेदना, उद्योग आणि ग्राहक आकडेवारी आणि व्यवहार प्रक्रियेसाठी वापरले जातात.

### सुपर कॉम्प्युटर

1980 च्या दशकात सुपर कॉम्प्युटरची ओळख झाली. सुपर कॉम्प्युटर हा सर्वात वेगवान संगणक आहे. सुपर कॉम्प्युटर आकाराने सर्वात मोठा आहे आणि इतर कोणत्याही संगणकापेक्षा किमतीत सर्वात महाग आहे.

हा सर्वात अत्याधुनिक, जटिल आणि एडवॉन्स संगणक आहे. त्याची साठवण क्षमता खूप मोठी आहे. ते एका सेकंदात लाखो सूचनांवर प्रक्रिया करू शकते. सुपर कॉम्प्युटर हा एका काळातील सर्वात वेगवान आणि शक्तिशाली संगणक आहे. सुपर कॉम्प्युटर खूप महाग आहेत. सुपरकॉम्प्युटर्सचा वापर अत्यंत गणना-केंद्रित कामांसाठी केला जातो. सुपर कॉम्प्युटरचा वापर स्पेशल ऍप्लिकेशन्ससाठी देखील केला जातो ज्यांना प्रचंड प्रमाणात गणिती गणना आवश्यक असते.

### सुपर कॉम्प्युटरची ऍप्लिकेशन्स

- हवामान अंदाज,
- हॉलिवूड पिक्चर पटांप्रमाणे ॲनिमेटेड ग्राफिक्स,
- द्रव डायनेमिक गणना
- अणुऊर्जा रिसर्च न
- अंतराळ विज्ञान
- शस्त्रे आणि क्षेपणास्त्र डिझाइन
- पेट्रोलियम अन्वेषण, आणि इ.

आज, सुपर कॉम्प्युटर क्रे, IBM आणि Hewlett-Packard सारख्या पारंपारिक कंपन्यांद्वारे तयार केले जातात. ऑक्टोबर 2010 पासून, Tianhe-1A सुपर कॉम्प्युटर जगातील सर्वात वेगवान आहे; ते चीन मध्ये स्थित आहे.

सुपर कॉम्प्युटर आणि मेनफ्रेममधील मुख्य फरक असा आहे की सुपर कॉम्प्युटर शक्य तितक्या जलद एक प्रोग्राम कार्यान्वित करण्यासाठी आपली सर्व शक्ती चॅनेल करतो, तर मेनफ्रेम एकाच वेळी अनेक प्रोग्राम कार्यान्वित करण्यासाठी त्याची शक्ती वापरते. आधुनिक सुपर कॉम्प्युटरमध्ये हजारो मायक्रोप्रोसेसर असतात. सुपर कॉम्प्युटर ऑनलाइन प्रक्रियेसाठी सबग्रहासारख्या हाय-स्पीड सुविधा वापरतो.

सुपर कॉम्प्युटर मॉडेल्सची बेरीज CRAY-XP, ETA10, Param आणि Deep Blue आहेत.

### संगणकाचे फायदे

- संगणकामध्ये उच्च विश्वासार्हतेसह खूप उच्च प्रक्रिया गती असते.

- मेमरीमध्ये मोठ्या प्रमाणात इनफार्मेशन स्टोर केली जाऊ शकते, कोणताही स्पेसिफिक डेटा/प्रोग्राम त्वरित पुनर्प्राप्त केला जाऊ शकतो.
- एखाद्या गुंतागुंतीच्या समस्येचे निराकरण अतिशय वेगाने शक्य आहे.
- मोठ्या प्रमाणातील डेटावर प्रक्रिया केल्याने लिपिकीय कामाची बरीच बचत होते ज्यामुळे प्रक्रिया खर्च कमी होतो.
- भट्टी, विषारी वातावरण, निर्वात, मानवरहित सबग्रह इ. अशा वातावरणात संगणक कार्यक्षमतेने कार्य करतात जेथे मानवाची सबस्थिती शक्य नसते.

### संगणकाची मर्यादा

- उच्च स्टार्टिंग खर्च.
- इनपुट इनफार्मेशन विधानांच्या स्वरूपात तयार करावी लागते ज्याला प्रोग्राम म्हणतात ज्यासाठी मोठ्या प्रमाणात ज्ञान आवश्यक असते.
- जेव्हा मोठ्या प्रमाणात डेटासाठी कारकुनी डेटा प्रोसेसिंग असेल आणि पुनरावृत्ती होत असेल तेव्हाच संगणकाचा वापर किफायतशीर असेल
- हे फक्त एक मशीन आहे ते स्वतःच चुका सुधारू शकत नाही.

### संगणकाची फंक्शन

सर्व संगणक खालील बेसिक युनिट्सपासून बनलेले आहेत आकृती मध्ये दाखवले आहे 8). ते खालीलप्रमाणे आहेत:-

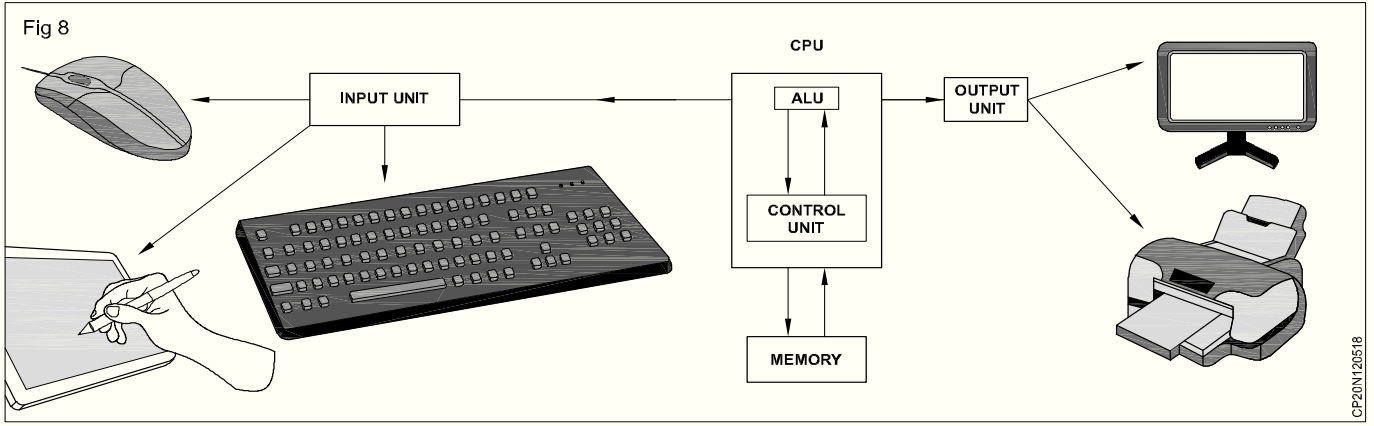
- 1 इनपुट युनिट
- 2 सेंट्रल प्रोसेसिंग युनिट (CPU)
  - a र्थमेटिक लॉजिक युनिट (ALU)
  - b कंट्रोल युनिट (CU)
- 3 मेमरी
- 4 आउटपुट युनिट

### इनपुट युनिट

कोणत्याही समस्येचे निराकरण करण्यासाठी संगणकांना डेटा आणि सूचना प्राप्त करणे आवश्यक आहे. म्हणून आपल्याला संगणकामध्ये डेटा आणि सूचना इनपुट करणे आवश्यक आहे. इनपुट युनिटमध्ये एक किंवा अधिक इनपुट उपकरणे असतात. कीबोर्ड हे सर्वात जास्त वापरले जाणारे इनपुट सबकरण आहे. काही इनपुट डिव्हाइसेस टेबल 1 मध्ये लिस्ट बद्ध आहेत.

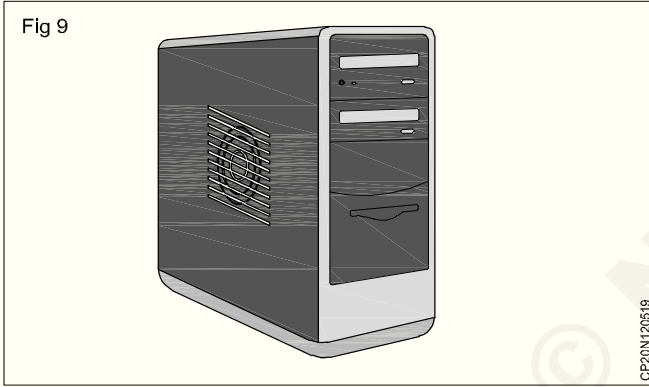
इनपुट उपकरणे खालील फंक्शन करतात.

- बाहेरील जगाचा डेटा आणि सूचना स्वीकारा.
- संगणकाला समजेल अशा फॉर्ममध्ये रूपांतरित करा.
- पुढील प्रक्रियेसाठी रूपांतरित डेटा संगणक सिस्टिम ला पुरवठा करा.



### सेंट्रल प्रोसेसिंग युनिट (CPU) (पिक्चर 9)

सेंट्रल प्रोसेसिंग युनिट (CPU) हा आकृती -8 प्रमाणे संगणकाचा इलेक्ट्रॉनिक मेंदू आहे. वैयक्तिक संगणकातील CPU सहसा एकच चिप असते. हे एकतर यूजर कडून किंवा सॉफ्टवेअरकडून आलेल्या सूचनांचे आयोजन आणि पालन करते. प्रोसेसर अनेक एलिमेंटनी बनलेला असतो. CPU खालील फंक्शन करते:



- हे सर्व आकडेमोड करते.
- हे सर्व निर्णय घेते.
- हे संगणकाच्या सर्व युनिट्सवर कंट्रोल ठेवते.

CPU चे दोन स्पेसिफिक एलिमेंट खालीलप्रमाणे आहेत:

अर्थमेटिक लॉजिक युनिट (ALU), जे अर्थमेटिक आणि लॉजिकल ऑपरेशन्स करते.

कंट्रोल युनिट (CU), जे मेमरीमधून सूचना काढते आणि डीकोड करते आणि ते कार्यान्वित करते, आवश्यकतेनुसार ALU वर कॉल करते.

### मेमरी

मेमरी म्हणजे संगणक किंवा इतर डिजिटल इलेक्ट्रॉनिक इक्विपमेंट्स मध्ये वापरण्यासाठी तात्पुरत्या किंवा कायमस्वरूपी प्रोग्राम किंवा डेटा संचयित करण्यासाठी वापरल्या जाणाऱ्या फिजिकल इक्विपमेंट्स चा रेफरन्स देते.

संगणकात मेमरी दोन टाइप ची असते.

- प्रायमरी मेमरी
- सेकंडरी मेमरी

### आउटपुट युनिट

आउटपुट युनिट प्रोसेसिंग युनिटकडून इनफार्मेशन घेते आणि मानवामध्ये रिझल्ट प्रदान करते वाचनीय (रिडेबल) फॉर्म.

### आउटपुट उपकरणे

काही आउटपुट उपकरणे आहेत

- मॉनिटर
- प्रिंटर
- प्लॉटर
- स्पीकर

### संगणक एप्लिकेशन

**विज्ञान:** शास्त्रज्ञ थिअरी विकसित करण्यासाठी आणि डेटाचे विश्लेषण आणि टेस्टिंग करण्यासाठी संगणक वापरत आहेत. संगणकाची उच्च गती आणि अचूकता विविध वैज्ञानिक विश्लेषणे करण्यास अनुमती देते. भूकंपाचा इमारतींवर कसा रिझल्ट होतो किंवा प्रदूषणाचा हवामानावर कसा रिझल्ट होतो याचा तपशीलवार अभ्यास करण्यासाठी त्यांचा वापर केला जाऊ शकतो. सबग्रह

आधारित एप्लिकेशन संगणकाच्या वापराशिवाय शक्य नव्हते. शिवाय, संगणकाशिवाय सौरमाला आणि विश्वाची इनफार्मेशन मिळणे शक्य होणार नाही.

**शिक्षण:** संगणकानेही शिक्षणाच्या संपूर्ण प्रक्रियेत क्रांती घडवून आणली आहे. सध्या, क्लास खोल्या, ग्रंथालये आणि संग्रहालये शिक्षण अधिक मनोरंजक बनवण्यासाठी संगणकाचा सक्षमपणे वापर करत आहेत. रेकॉर्डेड टेलिव्हिजन शोच्या विपरीत, संगणक-अनुदानित शिक्षण (CAE) आणि संगणक-आधारित प्रशिक्षण (CBT) पॅकेजमुळे शिकणे अधिक परस्परसंवादी बनत आहे.

**औषध आणि आरोग्य सर्विस :** वैद्यकीय क्षेत्रात संगणकाचा वापर वाढला आहे. आता, डॉक्टर आजाराचे निदान करण्यापासून ते गुंतागुंतीच्या शस्त्रक्रियेदरम्यान रुग्णाच्या स्थितीचे निरीक्षण करण्यासाठी संगणक वापरत आहेत. ऑटोमेटेड इमेजिंग तंत्राचा वापर करून, डॉक्टर एखाद्या व्यक्तीच्या शरीरात पाहण्यास सक्षम आहेत आणि प्रत्येक अवयवाचा तपशीलवार अभ्यास करू शकतात (उदा. CT स्कॅन किंवा MRI स्कॅन), जे

काही वर्षांपूर्वी शक्य नव्हते. स्पेशल -उद्देशीय संगणकांची अनेक उदाहरणे आहेत जी मानवी शरीरात कार्य करू शकतात जसे की कॉक्लियर इम्प्लांट, एक स्पेशल टाइप चे श्रवण यंत्र ज्यामुळे कर्णबधिरांना ऐकणे शक्य होते.

**अभियांत्रिकी/आर्किटेक्चर/उत्पादन:** वास्तुविशारद आणि अभियंते डिझाईनिंग आणि ड्रॉइंगमध्ये संगणकाचा मोठ्या प्रमाणावर वापर करत आहेत. संगणक अशा वस्तू तयार करू शकतो ज्या तिन्ही आयामांमधून पाहता येतील. व्हर्च्युअल रिअॅलिटी सारख्या तंत्रांचा वापर करून, वास्तुविशारद अशा घरांचा सर्च घेऊ शकतात जे डिझाइन केले गेले आहेत परंतु बांधलेले नाहीत. उत्पादक कारखाने धोकादायक कामे करण्यासाठी संगणकीकृत रोबोटिक शस्त्रे वापरत आहेत. याशिवाय, कॉम्प्युटर-एडेड मॅन्युफॅक्चरिंग (सीएमए) उत्पादनाची रचना करण्यासाठी, भागांची ऑर्डर देण्यासाठी आणि उत्पादनाचे नियोजन करण्यासाठी वापरले जाऊ शकते. अशा प्रकारे, संगणक संपूर्ण उत्पादन प्रक्रियेत समन्वय साधण्यास मदत करतात.

**मनोरंजन:** मनोरंजन उद्योगात संगणकाचा अधिक वापर होत आहे. ते प्रतिमा आणि आवाज कंट्रोल करण्यासाठी वापरले जातात. प्रेक्षकांना मंत्रमुग्ध करणारे स्पेशल इफेक्ट्स संगणकाशिवाय शक्य झाले नसते. याशिवाय, संगणकीकृत ॲनिमेशन आणि रंगीत ग्राफिक्स यांनी पिक्चर पट उद्योगाचे आधुनिकीकरण केले आहे.

**कम्युनिकेशन:** ई-मेल किंवा इलेक्ट्रॉनिक मेल हे कम्युनिकेशन माध्यमांपैकी एक आहे ज्यामध्ये संगणक वापरला जातो. संगणक आणि टेलिफोन लाईनच्या साहाय्याने ई-मेलद्वारे, संदेश आणि रिपोर्ट एका व्यक्तीकडून एक किंवा अधिक व्यक्तींना पाठवले जातात. या सेवेचा फायदा असा आहे की संदेश ट्रान्सफर करताना वेळेची बचत होते, कागदाचा अपव्यय टळतो इत्यादी. शिवाय, संदेश प्राप्त करणारी व्यक्ती जेव्हाही मोकळी असेल तेव्हा ते संदेश वाचू शकते आणि ते सेव्ह करू शकते, त्याला उत्तर देऊ शकते, ते फॉरवर्ड करू शकते किंवा संगणकावरून हटवू शकते.

**बिजनेस एप्लिकेशन:** संगणकाचा हा एक महत्त्वाचा यूजेस आहे. सुरुवातीला, संगणकाचा वापर बॅच प्रोसेसिंग जॉबसाठी केला जात असे, जेथे एखाद्याला संगणकाकडून त्वरित प्रतिसादाची आवश्यकता नसते. सध्या, संगणक प्रामुख्याने रिअल-टाइम एप्लिकेशन्ससाठी वापरले जातात (जसे की विक्री काउंटरवर) ज्यांना संगणकाकडून त्वरित प्रतिसाद आवश्यक असतो. बिजनेस अंदाज, पगाराची बिले आणि वैयक्तिक रेकॉर्ड तयार करण्यासाठी, बँकिंग ऑपरेशन्स आणि डेटा स्टोरेजमध्ये, विविध टाइप च्या जीवन विमा व्यवसायात आणि मॅनेजमेंट ला मदत म्हणून संगणकांचा वापर केला जातो अशा विविध समस्या आहेत. व्यवसाय देखील संगणकांचे नेटवर्किंग वापरत आहेत, जेथे डेटा आणि इनफार्मेशन सामायिक करण्यासाठी अनेक संगणक एकत्र जोडलेले आहेत. ई-मेल आणि इंटरनेटच्या वापरामुळे व्यवसाय करण्याच्या मेथड्स बदलल्या आहेत.

**प्रकाशन:** संगणकांनी डेस्कटॉप प्रकाशन (DTP) म्हणून ओळखले जाणारे एरिया तयार केले आहे. डीटीपीमध्ये, संगणक आणि लेझर प्रिंटरच्या साहाय्याने प्रकाशनाचे काम स्वतःच करता येते. संगणक आणि डीटीपी सॉफ्टवेअर वापरून कंटेंट टेबल आणि इंडेक्स तयार करणे यासारख्या दीर्घ मॅन्युअल तासांची आवश्यकता असलेली अनेक कामे ऑटोमॅटिक पणे करता येतात.

**बँकिंग:** बँकिंग आणि वित्त क्षेत्रात संगणकाचा मोठ्या प्रमाणावर वापर केला जातो. पैसे जमा करण्यासाठी आणि काढण्यासाठी लोक एटीएम (ऑटोमेटेड टेलर मशीन) सर्विस 24 तास वापरू शकतात. जेव्हा बँकेच्या विविध शाखा संगणक नेटवर्कद्वारे जोडल्या जातात, तेव्हा आंतरशाखीय व्यवहार जसे की चेक आणि ड्राफ्ट कोणत्याही विलंबाशिवाय करता येतात.

### रेल्वे आरक्षण सिस्टिम

या सिस्टिम चा वापर करून, युजर खालील ऑपरेशन्स ऑनलाइनद्वारे करू शकतो. (वेब साईट: [www.irctc.co.in](http://www.irctc.co.in))

- ट्रेन आणि तिची वेळ शोधा
- जागा आणि बर्थ उपलब्धता तपासा
- तिकिटे बुक करणे आणि रद्द करणे
- PNR ची स्थिती (प्रवाशाचे नाव रेकॉर्ड)

### दूरध्वनी / विद्युत मंडळ बिलिंग:

या सिस्टिम चा वापर करून वापरकर्ते खालील ऑपरेशन्स ऑनलाइन करू शकतात. (वेबसाईट: [portal.bsnl.in](http://portal.bsnl.in) - BSNL)

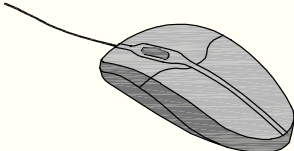
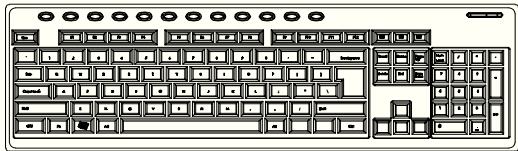
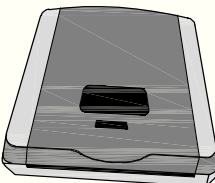
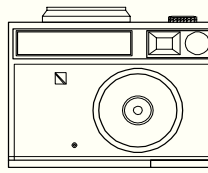

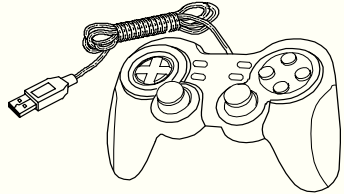
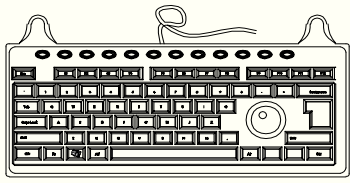
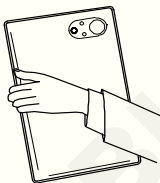



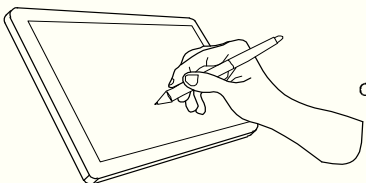
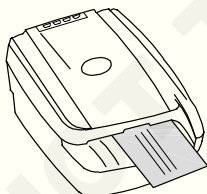
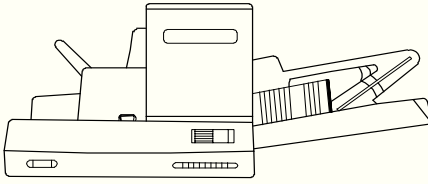
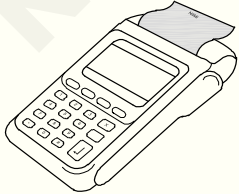
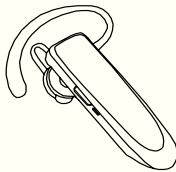
- दूरध्वनी/विद्युत मंडळाचा क्रमांक नोंदवा
- चेक करा आणि बिलाची रक्कम भरा
- तक्रारी नोंदवा

### ई-गव्हर्नन्स

ई-गव्हर्नन्स म्हणजे तंत्रज्ञानावर आधारित एडमिनिस्ट्रेशन . ईजीगव्हर्नन्स म्हणजे इनफार्मेशन आणि दळणवळण तंत्रज्ञानाचा (आयसीटी) यूजेस सरकारी सर्विस , इनफार्मेशन कम्युनिकेशन व्यवहारांची देवाणघेवाण, विविध स्टँड-अलोन सिस्टिम आणि सरकारमधील सर्विस चे एकत्रीकरण.

नागरिक (G2C), सरकार-ते-व्यवसाय (G2B, सरकार-ते-सरकार (G2G) तसेच बँक ऑफिस प्रक्रिया आणि संपूर्ण सरकारी फ्रेम वर्कमध्ये परस्परसंवाद.

सामान्य माणसाला (शहरी आणि ग्रामीण दोन्ही भागात) त्रासओपन , पारदर्शक आणि कार्यक्षम सर्विस देण्याच्या दृष्टीकोनातून ई-गव्हर्नन्स सर्व क्षेत्रांचा समावेश करते.

<p>Fig 1</p> 	<p>MOUSE</p>  <p>KEYBOARD</p>
 <p>SCANNER</p>	 <p>CAMERA</p>
 <p>WEB CAMERA</p>	 <p>JOY STICK</p>
 <p>TRACK BALL</p>	 <p>TOUCH PAD</p>
 <p>LIGHT PEN</p>	 <p>BAR CODE READER</p>
 <p>MICROPHONE</p>	 <p>GRAPHIC TABLET</p>
 <p>MAGNETIC INK CHARACTER READER</p>	 <p>OPTICAL MARK READER</p>
 <p>MAGNETIC CARD READER</p>	 <p>BLUETOOTH</p>

CP20N1205T1

## CPU आर्किटेक्चर आणि मदरबोर्डचा परिचय (Introduction to CPU architecture and motherboard)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- हार्डवेअर काय आहे ते सांगा आणि इंटरनल आणि एक्सटर्नल हार्डवेअरची टेस्टिंग करा
- लिस्ट बद्ध हार्डवेअरची थोडक्यात इनफार्मेशन द्या
- पार्टिशन आणि त्यांचे टाइप काय आहेत ते सांगा
- बूटिंग आणि त्याची प्रक्रिया स्पष्ट करा.

### संगणक हार्डवेअर

संगणकाच्या फिजिकल एककांना संगणकाचे हार्डवेअर असे म्हणतात.

### इंटरनल हार्डवेअर उदाहरणे

- Blu-Ray, CD-ROM, आणि DVD
- सीपीयू
- हार्ड ड्राइव्ह
- मदरबोर्ड
- रॅम
- ध्वनी कार्ड
- व्हिडिओ कार्ड
- SMPS

### एक्सटर्नल हार्डवेअर उदाहरणे

- फ्लॉट-पॅनल, मॉनिटर, आणि LCD
- कीबोर्ड
- माउस
- प्रिंटर
- स्कॅनर

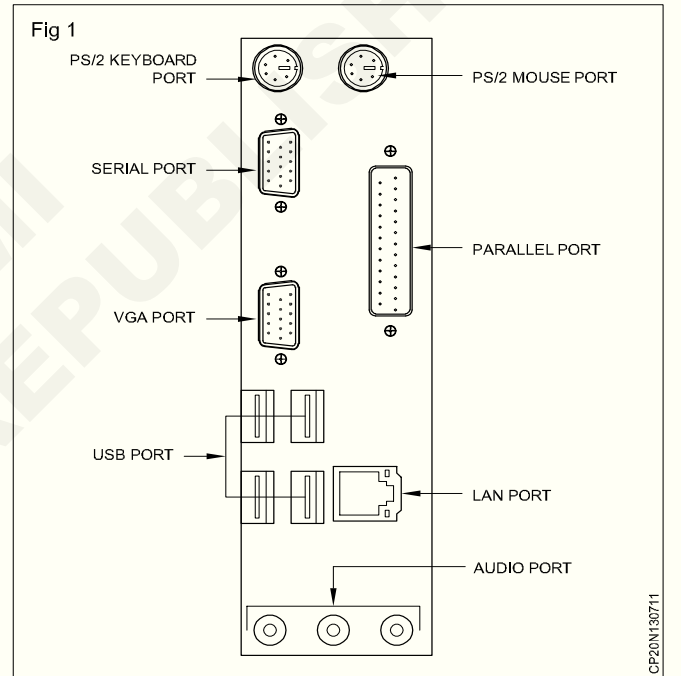
### CPU आणि ALU

सेंट्रल प्रोसेसिंग युनिट (CPU) हे संगणकाचे हृदय आहे. हे हार्डवेअर आहे, जे सिस्टमचे बेसिक अंकगणितीय, लॉजिकल आणि इनपुट/आउटपुट ऑपरेशन्स करून संगणक प्रोग्रामच्या सूचनांचे पालन करते.

### CPU पोर्ट्स आणि कनेक्टर

पोर्ट हे कॉम्प्युटर कॅबिनेटच्या मागील बाजूस असलेले कनेक्टर असते जेथे तुम्ही प्रिंटर, कीबोर्ड, स्कॅनर, मॉडेम इ. सारखे एक्सटर्नल सबकरण प्लग इन करता. यामुळे सूचना आणि डेटा संगणक आणि डिव्हाइस दरम्यान प्रवाहित होऊ शकतो. संगणक पोर्ट्सना सामान्यतः इनपुट/आउटपुट पोर्ट (I/O पोर्ट) असेही संबोधले जाते. हे पोर्ट एकतर अनुक्रमिक किंवा पॅरलल असू शकतात. आकृती 1 वैयक्तिक संगणकावर सामान्यपणे सबलब्ध असलेले पोर्ट दाखवते.

बहुतेक कनेक्टर वेगळे केले जातात, केबलला फक्त योग्य दिशेने प्लग इन करण्याची परवानगी देते. कीबोर्ड आणि माउस "PS2" (वैयक्तिक सिस्टिम 2) कनेक्टर वापरतात. PS2 कनेक्टर रंग-कोडेड आहेत. जांभळा कनेक्टर कीबोर्डसाठी आहे. हिरवा कनेक्टर माउससाठी आहे.



- **PS/2 पोर्ट:** स्टॅण्डर्ड कीबोर्ड आणि माउस बहुतेक वेळा PS/2 पोर्टद्वारे संगणकाशी कनेक्ट होतात. प्लग इन करण्यासाठी कीबोर्ड किंवा माउस केबल, प्रथम केबल कनेक्टरशी जुळवा. नंतर केबलला कनेक्टरमध्ये ढकलून द्या. कनेक्टरला जबरदस्ती करू नका याची खात्री करा कारण तुम्ही पिन वाकवणार आहात
- **सीरियल आणि पॅरलल पोर्ट:** सीरियल पोर्ट आणि पॅरलल पोर्ट प्रिंटर आणि इतर एक्सटर्नल उपकरणांना कनेक्शनची परवानगी देतात. सीरियल पोर्टद्वारे बाइट हस्तांतरित करण्यासाठी, आठ बिट रंगेत आहेत आणि बिट-बिट पाठवले जातात. तथापि, पॅरलल पोर्टमध्ये, सर्व आठ बिट एकाच वेळी हस्तांतरित केले जातात
- पॅरलल पोर्ट, सीरियल पोर्ट आणि व्हिडिओ पोर्ट सर्व "D" टाइप चे कनेक्टर वापरतात (अनुक्रमे DB-25M, DB-9M आणि DB-15F). त्यांना त्यांच्या आकारामुळे डी कनेक्टर म्हणतात, जे केबलला फक्त एकाच मार्गाने प्लग करण्याची परवानगी देतात.

## यूएसबी (युनिव्हर्सल सीरियल बस) पोर्ट

डिजिटल कॅमेरे, स्कॅनर आणि प्रिंटर यांसारखी उपकरणे अनेकदा USB पोर्टद्वारे मदरबोर्डशी जोडली जातात. USB कनेक्टरचा स्पेसिफिक आयताकृती आकार त्याला सहज ओळखता येतो.

**USB मध्ये अनेक वैशिष्ट्ये आहेत ज्यामुळे ते PC वर स्पेशल तः** लोकप्रिय होते. प्रथम, यूएसबी डिव्हाइस गरम बदलण्यायोग्य आहेत. तुमची सिस्टीम रीस्टार्ट न करता तुम्ही ते घालू किंवा काढू शकता

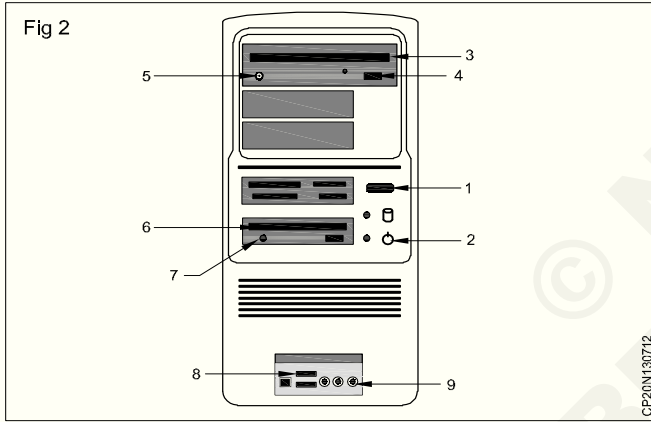
**LAN (लोकल एरिया नेटवर्क) पोर्टः** LAN पोर्टचा वापर पीसीला लोकल नेटवर्कशी किंवा हाय स्पीड इंटरनेट सर्विस शी जोडण्यासाठी केला जातो.

**VGA (व्हिडिओ ग्राफिक्स अरे) पोर्टः** VGA पोर्ट इंटीग्रेटेड व्हिडिओमध्ये एक्सेस प्रदान करतो.

**ऑडिओ पोर्टः** हे इंटीग्रेटेड ऑडिओमध्ये एक्सेस प्रदान करते.

ऑडिओ जॅक मागील पॅनेलवरील सर्वात गोथळात टाकणारे कनेक्टर आहेत. जरी जॅक कधीकधी कलरकोड केलेले असले तरी, क्वचितच त्यात प्लग इन करणारी उपकरणे.

## CPU फ्रंट पॅनेल (पिक्चर 2)



त्यात खालील पार्ट असू शकतात.

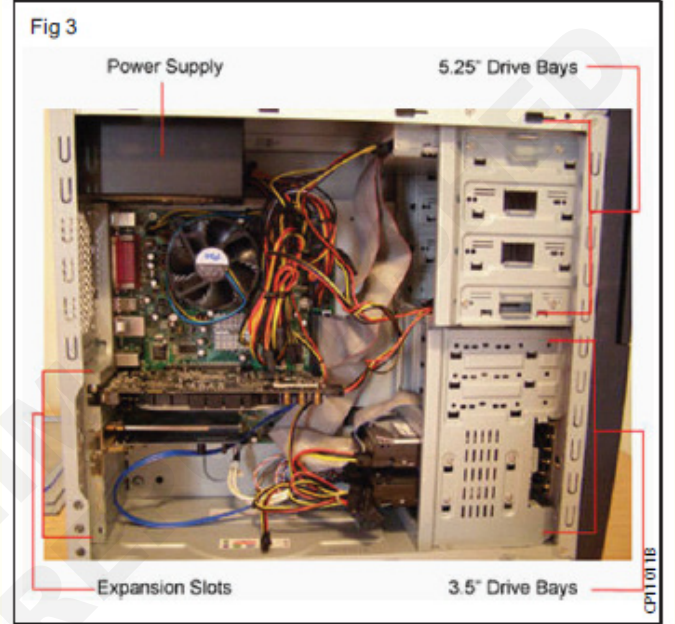
- पॉवर चालू/बंद स्विच
- पॉवर इंडिकेटर
- CD/DVD ड्राइव्ह
- CD/DVD ड्राइव्ह उघडा/बंद करा बटण
- CD/DVD ड्राइव्ह इंडिकेटर
- फ्लॉपी डिस्क ड्राइव्ह
- फ्लॉपी डिस्क ड्राइव्ह इंडिकेटर
- USB पोर्ट
- ऑडिओ आणि माइक कनेक्टर

निर्मात्याच्या आधारावर CPU कॅबिनेटची रचना बदलू शकते. सिस्टम युनिट आणि त्याचे एलिमेंट

सिस्टीम युनिट हे अनेक सबयुक्त एलिमेंटनी भरलेले बॉक्ससारखे युनिट आहे, प्रत्येक स्वतंत्र कार्य करते. हे एलिमेंट संगणकाचे मुख्य कार्य पूर्ण करण्यासाठी एकत्र काम करतात, उदा. इनपुट स्वीकारा आणि प्रक्रिया करा आणि आउटपुट वितरित करा. हा विपार्ट एकामागून एक या एलिमेंटचा तपशीलवार डिस्क्रिप्शन करेल. आकृती 3 सिस्टम युनिटचे विविध एलिमेंट दर्शविते.

## वीज पुरवठा

पॉवर पुरवण्यासाठी पॉवर सप्लाय पीसीमधील जवळपास प्रत्येक डिव्हाइसला जोडतो. हे केसच्या मागील बाजूस स्थित आहे. सिस्टम युनिट पॉवर प्रोटेक्शन यंत्राद्वारे AC मेनमधून पॉवर काढते.



ही शक्ती थेट इंटरनल एलिमेंटना पुरवली जात नाही. त्याऐवजी, इंटरनल वीज पुरवठा म्हटल्या जाणाऱ्या एलिमेंटपैकी एक, एसी इनपुटला 5 आणि 12 व्होल्टच्या डीसी आउटपुटमध्ये रूपांतरित करतो. साधारणपणे, इंटरनल वीज पुरवठा म्हणून संबोधले जाते स्विच मोड पॉवर सप्लाय (SMPS).

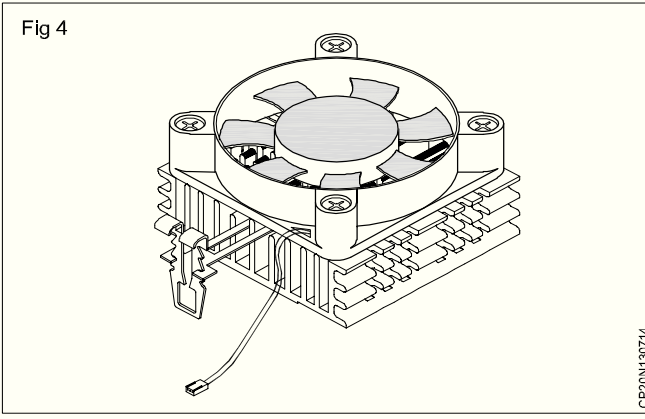
फ्लॉपी ड्राइव्ह, हार्ड डिस्क ड्राइव्ह, मदरबोर्ड आणि एक्सटर्नल सबकरण जसे की कीबोर्ड सारख्या इतर इंटरनल एलिमेंटना आवश्यक व्होल्टेज पुरवण्यासाठी SMPS केबल कनेक्टर प्रदान करते. सिस्टीम युनिटचा चालू/बंद स्विच प्रत्यक्षात SMPS चा एक पार्ट आहे.

## पंखा

SMPS मध्ये एक छोटा पंखा असतो, ज्याला एक्झॉस्ट फॅन म्हणतात, SMPS ला जोडलेला असतो (पिक्चर 4). जोपर्यंत संगणक चालू असतो तोपर्यंत हा पंखा फिरतो. त्याचे कार्य SMPS युनिट थंड करणे आहे.

**ड्राइव्ह बेः** 5.25" आणि 3.5" ड्राइव्ह बेजमध्ये संगणकामध्ये अनेक टाइप चे स्टोरेज उपकरणे असू शकतात.

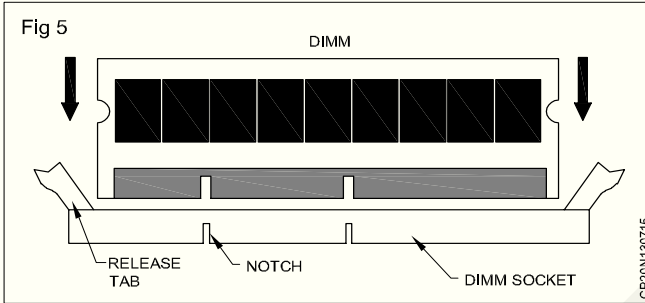
**एक्सपान्शन स्लॉटः** एक्सपान्शन स्लॉट हा एक स्लॉट आहे जो संगणकाच्या मदर बोर्डच्या आत असतो जो अतिरिक्त परिधीयांना त्याच्याशी जोडण्याची परवानगी देतो.



मेमरी स्लॉट

रँडम ऍक्सेस मेमरी समाविष्ट करण्यासाठी मेमरी स्लॉट वापरला जातो

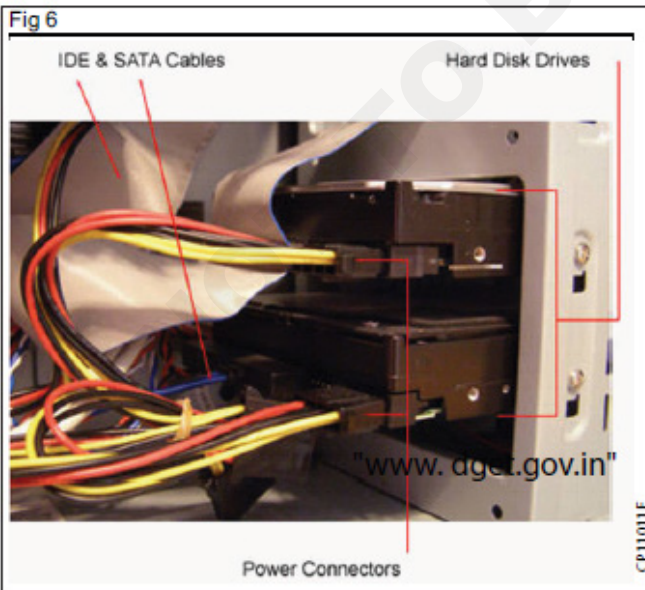
मेमरी (RAM) आकृती 5 मध्ये दर्शविली आहे.



स्टोरेज ड्रायव्हर्स

स्टोरेज ड्रायव्हर्स जसे की हार्ड ड्राइव्ह्स, ऑप्टिकल ड्राइव्ह आणि फ्लॉपी ड्राइव्ह हे सर्व केबल्सद्वारे मदरबोर्डशी जोडले जातात आणि संगणकाच्या आत माउंट केले जातात.

IDE आणि SATA केबल्स: आकृती 6 मदरबोर्डशी वेगवेगळ्या प्रकारे जोडलेल्या दोन हार्ड डिस्क ड्राइव्ह दाखवते. एक जुने IDE केबल कनेक्शन वापरतो तर दुसरा SATA (सिरियल अँडव्हान्सड टेक्नॉलॉजी अटँचमेंट)



**पॉवर पोर्ट:** ड्राइव्हवरील पॉवर पोर्टमध्ये प्लग केलेल्या केबल्सद्वारे ड्राइव्हवर पॉवर वितरित केली जाते.

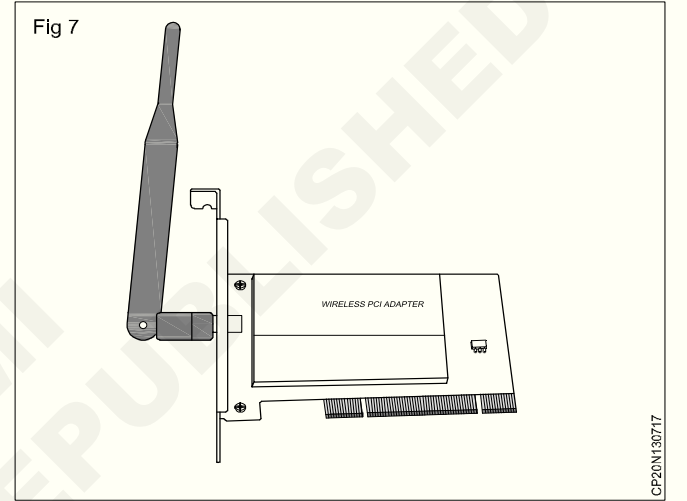
## पेरिफेरल कार्ड स्लॉट

पेरिफेरल कार्ड्स हे मदर बोर्डवर सबलब्ध असलेले अतिरिक्त एक्सपान्शन स्लॉट आहेत ज्यावर पेरिफेरल कार्ड टाकले जाऊ शकतात.

खालील पेरिफेरल कार्ड आहेत

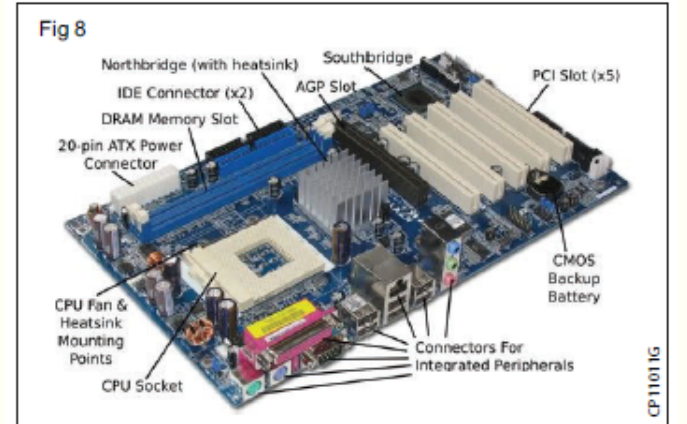
- साऊंड कार्ड
- व्हिडिओ कार्ड
- फॅशन
- वायरलेस नेटवर्क

आकृती 7 PCI (पेरिफेरल एलिमेंट इंटरकनेक्ट) कनेक्टरसह डिझाइन केलेले पेरिफेरल कार्ड दाखवते.



मदरबोर्ड

मदरबोर्ड हे प्रिंटेड सर्किट आहे जे संगणकाचा पाया आहे आणि CPU, RAM आणि इतर सर्व संगणक हार्डवेअर एलिमेंटना पिक्चर 8 प्रमाणे एकमेकांसोबत कार्य करण्यास अनुमती देते.



मदरबोर्ड हा संपूर्ण सिस्टिम चा प्राथमरी एलिमेंट आहे. मदर बोर्ड हा एक मोठा बोर्ड असतो ज्यामध्ये अनेक लहान इलेक्ट्रॉनिक सर्किट असतात आणि इतर एलिमेंट व्हा मान असतात. सर्व पेरिफेरल उपकरणे मदरबोर्डशी जोडलेली आहेत. मदरबोर्डचे एलिमेंट आहेत:

- कीबोर्ड / माउस पोर्ट
- पॅरलल आणि सिरीयल पोर्ट
- प्रोसेसर सॉकेट
- AGP स्लॉट
- PCI स्लॉट
- एक स्लॉट
- CMOS बॅटरी
- डेटा कार्ड कनेक्टर
- मेमरी स्लॉट
- फ्लॉपी पोर्ट
- फॅन हेडर
- मुख्य पॉवर कनेक्टर

**फ्लॉपी पोर्ट:** फ्लॉपी ड्राइव्ह 34-पिन रिबन केबलद्वारे संगणकाशी कनेक्ट होते, जे यामधून मदरबोर्डशी कनेक्ट होते. फ्लॉपी कंट्रोलर हा फ्लॉपी ड्राइव्ह कंट्रोल करण्यासाठी वापरला जातो.

**रॅम स्लॉट:** रॅडम-एक्सेस मेमरी (RAM) सध्या CPU द्वारे वापरले जाणारे प्रोग्राम आणि डेटा स्टोर करते.

रॅमबाइट्स नावाच्या युनिटमध्ये मोजले जाते. RAM अनेक प्रकारे पॅकेज केली गेली आहे

- SIMM-सिंगल इनलाइन मेमरी मॉड्यूल -32 किंवा 72 पिन
- DIMM- ड्युअल इनलाइन मेमरी मॉड्यूल -168 पिन.

DIMM मॉड्यूलच्या PC च्या बहुतेक वापरात

**ROM BIOS चिप:** याचा अर्थ रीड ओन्ली मेमरी बेसिक इनपुट-आउटपुट सिस्टम.

बिल्ट-इन सॉफ्टवेअर जे डिस्कवरून प्रोग्राम एक्सेस न करता संगणक काय करू शकतो हे ठरवते. पीसी वर, BIOS कीबोर्ड, डिस्प्ले स्क्रीन, डिस्क ड्राइव्ह, सीरियल कम्युनिकेशन्स आणि अनेक विविध फंक्शन्स कंट्रोल करण्यासाठी आवश्यक असलेले सर्व कोड समाविष्टीत आहे.

BIOS सामान्यतः a मध्ये ठेवलेले असते रॉम संगणकासोबत येणारी चिप (याला अनेकदा रॉम BIOS). हे सुनिश्चित करते की BIOS नेहमी सबलब्ध असेल आणि डिस्कच्या अपयशामुळे नुकसान होणार नाही.

हे संगणकाला स्वतः बूट करणे देखील शक्य करते. कारण RAM पेक्षा वेगवान आहे रॉम, तथापि, अनेक संगणक उत्पादक सिस्टम डिझाइन करतात जेणेकरून BIOS वरून कॉपी केली जाईल रॉम करण्यासाठी रॅम प्रत्येक वेळी संगणक बूट होतो. याला सावली म्हणून ओळखले जाते.

बर्याच आधुनिक पीसीमध्ये फ्लॅश BIOS आहे, याचा अर्थ असा की BIOS फ्लॅश मेमरी चिपवर रेकॉर्ड केले गेले आहे, जे आवश्यक असल्यास अपडेट केले जाऊ शकते. पीसी BIOS बर्यापैकी प्रमाणित आहे, म्हणून सर्व पीसी या लेयर वर समान आहेत (जर भिन्न आहेत BIOS आवृत्त्या). अतिरिक्त या फंक्शन्स सहसा सॉफ्टवेअर मॉड्यूलद्वारे जोडले जातात.

याचा अर्थ तुम्ही DOS च्या नवीन आवृत्तीमध्ये बदल न करता अपग्रेड करू शकता BIOS. पीसी BIOS जे प्लग-अँड-प्ले (PnP) हाताळू शकतात अशा उपकरणांना PnPBIOS म्हणून ओळखले जाते, हे BIOS नेहमी रॉम एवजी फ्लॅश मेमरीसह लागू केले जातात.

## CMOS बॅटरी

CMOS(कॉम्प्लिमेंटरी मेटल ऑक्ससाईड सेमीकंडक्टर) हा शब्द सामान्यतः संगणकाच्या मदरबोर्डवरील थोड्या प्रमाणात मेमरीचे डिस्क्रिप्शन करण्यासाठी वापरला जातो BIOS सेटिंग्ज

बहुतेक CMOS बॅटरी मदरबोर्डच्या आयुष्यभर टिकतील (बहुतेक प्रकरणांमध्ये 10 वर्षांपर्यंत) परंतु काहीवेळा बदलण्याची आवश्यकता असेल. चुकीची किंवा संध सिस्टीमची date आणि वेळ आणि BIOS सेटिंग्ज नष्ट होणे ही CMOS बॅटरी मृत किंवा संपुष्टात येण्याची प्रमुख सिम्बॉल्स आहेत.

**ISA स्लॉट:** ( इंडस्ट्री स्टँडर्ड आर्किटेक्चर) हे एक्सपान्शन बसचे स्टॅण्डर्ड आर्किटेक्चर आहे. ISA सुसंगत कार्ड कनेक्ट करण्यासाठी मदरबोर्डमध्ये काही स्लॉट असू शकतात.

**PCI स्लॉट:** इंटेलने सादर केले पेरिफेरल कॉम्पोनन्ट इंटरकनेक्ट बस प्रोटोकॉल. पीसीआय बसचा वापर I/O उपकरणांना संगणकाच्या मुख्य तर्काशी जोडण्यासाठी केला जातो. PCI बसने ISA बसची जागा घेतली आहे. PC मदरबोर्डमध्ये एक PCI स्लॉट असतो परंतु सामान्यतः एकापेक्षा जास्त.

PCI बस आर्किटेक्चर हे प्रोसेसर-स्वतंत्र बस स्पेसिफिकेशन आहे जे पेरिफेरलला CPU न वापरता थेट सिस्टम मेमरीमध्ये एक्सेस करण्यास अनुमती देते.

**AGP स्लॉट:** द आकसलॅटेड ग्राफिक्स पोर्ट (AGP) हे संगणकाच्या मदरबोर्डवर व्हिडिओ कार्ड जोडण्यासाठी हायस्पीड पॉइंट-टू-पॉइंट चॅनेल आहे.

## वीज पुरवठा प्लग इन

पॉवर सप्लाय, त्याच्या नावाप्रमाणे, पीसी (पर्सनल कॉम्प्युटर) ऑपरेट करण्यासाठी आवश्यक विद्युत उर्जा प्रदान करते. वीज पुरवठा स्टॅण्डर्ड 110-V AC पॉवर घेतो आणि +/-12-Volt, +/-5-Volt आणि 3.3-Volt DC पॉवरमध्ये रूपांतरित होतो. पॉवर सप्लाय कनेक्टरमध्ये 20-पिन असतात आणि कनेक्टर फक्त एकाच दिशेने जाऊ शकतो.

## हार्ड डिस्क आणि पार्टिशन

पार्टिशन ही हार्ड डिस्कला अनेक भागांमध्ये स्लिट करण्याची प्रक्रिया आहे आणि OS इंस्टाल करण्यासाठी कोणत्याही एका भागाचा किंवा पार्टिशन चा वापर करते किंवा मल्टिपल OS इंस्टाल करण्यासाठी दोन किंवा अधिक पार्टिशन वापरतात.

परंतु त्याचे नेहमी एक पार्टिशन असू शकते आणि एकच OS इंस्टाल करण्यासाठी संपूर्ण हार्ड डिस्क जागा वापरता येते, परंतु मोठ्या हार्ड डिस्कच्या वापरकर्त्यासाठी हे डेटा मॅनेजमेन्ट दुःस्वप्न होईल.



आता, मास्टर बूट रेकॉर्ड (MBR) च्या संरचनेमुळे, फक्त चार पार्टीशन आहेत, आणि या चार पार्टीशन ना प्रायमरी पार्टीशन म्हणतात..

एक्सटेंडेड पार्टीशन हे स्वतःच वापरण्यायोग्य पार्टीशन नाही, परंतु ते "कंटेनर" सारखे आहे आणि ते लॉजिकल ड्राइव्ह ठेवण्यासाठी वापरले जाते! म्हणजेच हे एक्सटेंडेड पार्टीशन मल्टिपल लॉजिकल विभाजनांमध्ये विपार्ट ले जाऊ शकते.

पार्टीशन मध्ये बूट करण्यासाठी, ते बूट करण्यायोग्य पार्टीशन किंवा ऍक्टिव्ह पार्टीशन म्हणून नियुक्त केले पाहिजे. ऍक्टिव्ह पार्टीशन हे असे पार्टीशन आहे जे बूट करण्यायोग्य म्हणून ध्वजांकित केलेले आहे किंवा ज्यामध्ये OS आहे, हे सामान्यतः प्रायमरी पार्टीशन आहे.

पार्टीशन चे टाइप :

- मास्टर
- पार्टीशन
- एक्सटेंडेड आणि
- लॉजिकल एक्सटेंडेड

**मास्टर बूट रेकॉर्ड (MBR):** MBR हे एक लहान 512 बाइट पार्टीशन आहे जे हार्ड डिस्कच्या पहिल्या फिजिकल सेक्टरमध्ये आहे. पोजिशन CHS 0,0,1 म्हणजे 0 असे दर्शविले आहे सिलेंडर, 0 वा हेड आणि 1 ला सेक्टर.

MBR मध्ये बूटस्ट्रॅप प्रोग्राम म्हणून ओळखला जाणारा एक छोटा प्रोग्राम आहे जो कोणत्याही OS मध्ये बूट करण्यासाठी जबाबदार असतो. MBR मध्ये पार्टीशन टेबल म्हणून ओळखले जाणारे टेबल देखील समाविष्ट आहे.

हे पार्टीशन चार्ट हार्ड डिस्कमधील सबलब्ध प्रायमरी विभाजनांची लिस्ट देणारे टेबल आहे. पार्टीशन टेबल संपूर्ण एक्सटेंडेड पार्टीशन ला एक प्रायमरी पार्टीशन मानते आणि ते टेबलमध्ये लिस्ट बद्ध करते!

त्यामुळे पार्टीशन तक्त्यामध्ये दोन संभाव्य नोंदी असू शकतात:-

- 4 प्रायमरी पार्टीशन पर्यंत.
- 3 पर्यंत प्रायमरी पार्टीशन आणि 1 एक्सटेंडेड पार्टीशन . (एकूण ४ पेक्षा जास्त नाही)

**पार्टीशन बूट सेक्टर (PBR):** हे लॉजिकल फर्स्ट सेक्टर आहे, ते सेक्टर प्रायमरी पार्टीशन च्या सुरुवातीस आहे. हे 512 बाइट एरिया देखील आहे, ज्यामध्ये OS फाइल्स सुरू करण्यासाठी किंवा चालवण्यासाठी काही प्रोग्राम आहेत. सर्व प्रायमरी विभाजनांचे स्वतःचे PBR आहेत.

**एक्सटेंडेड बूट सेक्टर (EBR):** हे लॉजिकल फर्स्ट सेक्टर आहे, ते एक्सटेंडेड पार्टीशन च्या सुरुवातीचे सेक्टर आहे. या EBR मध्ये पार्टीशन चार्ट आहे, जे एक्सटेंडेड पार्टीशन मध्ये सबलब्ध लॉजिकल विभाजनांची लिस्ट देते. म्हणजेच त्यात प्रत्येक लॉजिकल विभाजनांचे स्टार्टिंग एड्रेस समाविष्ट आहेत.

**लॉजिकल एक्सटेंडेड बूट सेक्टर (LEBR):** प्रत्येक लॉजिकल पार्टीशन च्या सुरुवातीला राहणारा हा लॉजिकल फर्स्ट सेक्टर आहे. हे प्रायमरी विभाजनांसाठी PBR सारखे आहे.

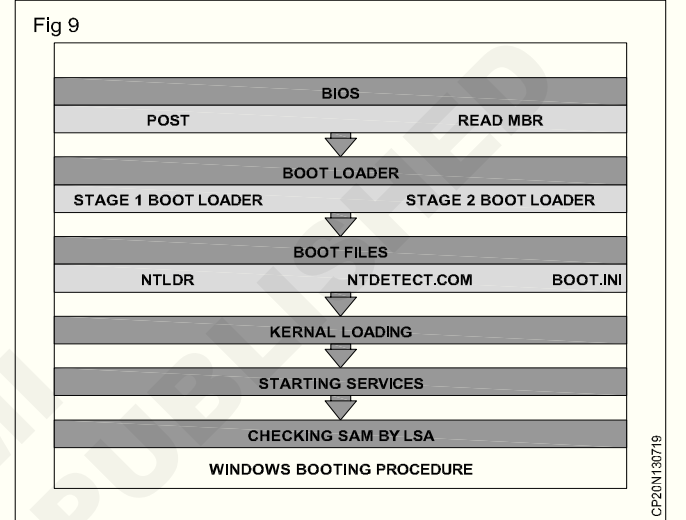
## बूट करणे

बूटिंग ही ऑपरेटिंग सिस्टीम (OS) लोड करण्याची आणि संगणकामध्ये इंस्टाल केलेली सर्व सिस्टम सॉफ्टवेअर आणि हार्डवेअर तपासण्याची प्रक्रिया आहे.

## विंडोज ऑपरेटिंग सिस्टमची बूटिंग प्रक्रिया

### BIOS ची फंक्शन

पहिली प्रक्रिया सुरू होते, जेव्हा संगणक बेसिक इनपुट आउटपुट सिस्टम (BIOS) चालू करतो तेव्हा दोन फंक्शन करतो, POST चालवणे आणि MBR (पिक्चर 9) वाचणे.



पोस्ट- POST म्हणजे पॉवर ऑनसेल्फ टेस्ट. POST संगणकाशी कनेक्ट केलेली सर्व हार्डवेअर उपकरणे जसे की RAM, हार्ड डिस्क इत्यादी तपासते आणि त्या हार्डवेअर उपकरणांसह सिस्टिम सुरळीतपणे चालू शकते याची खात्री करा. POST अयशस्वी झाल्यास सिस्टिम बीप आवाजाने थांबते.

आता BIOS तपासते बूट प्राधान्य. आम्ही सीडी ड्राइव्ह, हार्ड डिस्क किंवा फ्लॉपी ड्राइव्ह म्हणून बूट प्राधान्य सेट करू शकतो.

**MBR-** BIOS चे पुढील कर्तव्य MBR वाचणे आहे. MBR म्हणजे मास्टर बूट रेकॉर्ड आणि हा हार्ड डिस्कवरील पहिला सेक्टर आहे. MBR मध्ये पार्टीशन चार्ट आणि बूट लोडर समाविष्ट आहे.

### बूट लोडरची फंक्शन

आता BIOS ने बूट लोडरवर कंट्रोल पास केले आहे आणि बूट लोडर हा एक छोटा प्रोग्राम आहे जो संगणकाच्या मेमरीमध्ये कर्नल लोड करतो. प्रत्यक्षात बूट लोडरचे दोन टप्पे आहेत, स्टेज 1 बूट लोडर आणि स्टेज 2 बूट लोडर.

स्टेज 1 बूट लोडर हा स्टेज 2 बूट लोडरचा दुवा आहे. स्टेज 2 बूट लोडर बूट पार्टीशन मध्ये असतो आणि तो कर्नलला मेमरीमध्ये लोड करतो.

## बूट फाइल्स आणि फंक्शन्स

विंडोज ऑपरेटिंग सिस्टममध्ये तीन बूट फाइल्स आहेत आणि त्या NTLDR, NTDETECT.COM आणि Boot.ini आहेत. बूट फाइल्स हार्ड डिस्कच्या ऍक्टिव्ह पार्टीशन मध्ये आणि सामान्यतः विंडोज मशीनमधील C ड्राइव्हमध्ये आढळतात.

**NTLDR** - NTLDR चा अर्थ NT लोडर आहे आणि हा दुसरा टप्पा बूट लोडर आहे. NTLDR चा पाथ C:\Windows\i386\NTLDR आहे.

**बूट. ini** - Boot.ini मध्ये NTLDR च्या कॉन्फिगरेशन फाइल्स आहेत. जेव्हा ऑपरेटिंग सिस्टीम लोड केली जाते तेव्हा आम्ही कर्नलला कोणतेही आर्ग्युमेंट्स पास करू शकत नाही, म्हणून ते आर्ग्युमेंट्स boot.ini द्वारे पास केले जातात. तुम्ही नोटपॅड उघडून boot.ini एडिट करू शकता. Boot.ini चा पाथ C:\boot.ini आहे.

## NTDETECT.COM

ही फाइल हार्डवेअर सर्च ते आणि NTLDR कडे इनफार्मेशन देते. गोळा केलेल्या इनफार्मेशन चा वापर करून NTLDR हार्डवेअर की तयार करते आणि ही की हार्डवेअर सर्च प्यासाठी वापरली जाते.

ऑपरेटिंग सिस्टमच्या प्रत्येक रीबूटनंतर एक नवीन हार्डवेअर की तयार केली जाते आणि म्हणूनच नवीन हार्डवेअर इंस्टाल केल्यानंतर सिस्टम रीबूट करण्यास सांगते. हार्डवेअर

NTLDR द्वारे तयार केलेल्या कळ HKEY\_LOCAL\_MACHINE ' HARDWARES येथे Windows नोंदणीमध्ये आढळू शकतात.

© NIMI  
NOT TO BE REPUBLISHED

## COPA - संगणक बेसिक आणि सॉफ्टवेअर इन्स्टॉलेशन

## BIOS सेटिंग्ज आणि त्यांचे बदल पहा (View the BIOS settings and their modifications)

उद्दिष्ट: या एक्सरसाईस च्या शेवटी तुम्ही सक्षम व्हाल

• विंडोच्या आवृत्त्या आणि वैशिष्ट्यांची यादी करा.

विंडोज आवृत्त्या आणि त्याची वैशिष्ट्ये

1985 मध्ये प्रथम रिलीज झाल्यापासून मायक्रोसॉफ्ट विंडोजने नऊ प्रमुख आवृत्त्या पाहिल्या आहेत. 29 वर्षांनंतर, विंडोज खूप वेगळे दिसत आहे परंतु वेळेच्या कसोटीवर टिकून राहिलेल्या एलिमेंटशी परिचित आहे, संगणकीय शक्ती वाढते आणि - अगदी अलीकडे - कीबोर्डवरून बदल आणि टचस्क्रीनवर माउस.

विंडोजच्या इतिहासाची थोडक्यात इनफार्मेशन येथे आहे, विंडोज 1 सह बिल गेट्स यांच्या हस्ते जन्मल्यापासून ते नवीन मायक्रोसॉफ्टचे मुख्य कार्यकारी सत्यनाडेला यांच्या नेतृत्वाखालील आगमनापर्यंत.

विंडोज १

विंडोजची पहिली आवृत्ती

विंडोजसाठी हे सर्व सुरू झाले. मूळ विंडोज 1 नोव्हेंबर 1985 मध्ये रिलीज झाला आणि 16-बिटमधील ग्राफिकल यूजर इंटरफेसवर मायक्रोसॉफ्टचा पहिला खरा प्रयत्न होता.

विकासाचे नेतृत्व मायक्रोसॉफ्टचे संस्थापक बिल गेट्स यांनी केले आणि कमांड-लाइन इनपुटवर अवलंबून असलेल्या एमएस-डॉसच्या शीर्षस्थानी धावले.

हे लक्षणीय होते कारण माऊस हे सामान्य संगणक इनपुट सबकरण होण्यापूर्वी ते माऊसच्या वापरावर जास्त अवलंबून होते. वापरकर्त्यांना या विपिक्वर इनपुट सिस्टिम शी परिचित होण्यास मदत करण्यासाठी, मायक्रोसॉफ्टने एक गेम, रिव्हर्स (स्क्रीनशॉटमध्ये व्ह मान) समाविष्ट केला आहे जो कीबोर्डवर नव्हे तर माऊस नियंत्रणावर अवलंबून आहे, ज्यामुळे लोकांना माऊस फिरवण्याची आणि ऑनस्क्रीन एलिमेंटवर क्लिक करण्याची सवय लावली जाते.

विंडोज २

ओव्हरलॅप्पींग विंडोसह विंडोज 2.

विंडोज 1 रिलीज झाल्यानंतर दोन वर्षांनी, मायक्रोसॉफ्टच्या विंडोज 2 ने डिसेंबर 1987 मध्ये ते बदलले. विंडोज 2 साठी सर्वात मोठा नवकल्पना म्हणजे विंडोज एकमेकांना ओव्हरलॅप करू शकतात आणि त्यात "आयकॉनीजिंग" किंवा "आयकॉनीजिंग" ऐवजी विंडोज कमी किंवा मोठे करण्याची क्षमता देखील सादर केली गेली. झूम करत आहे".

कंट्रोल पॅनल, जिथे विविध सिस्टम सेटिंग्ज आणि कॉन्फिगरेशन पर्याय एकाच ठिकाणी एकत्रित केले गेले होते, विंडोज 2 मध्ये सादर केले गेले आणि आजपर्यंत टिकून आहे.

मायक्रोसॉफ्ट वर्ड आणि एक्सेलने देखील विंडोज 2 वर त्यांचे पहिले प्रदर्शन केले

विंडोज ३

विंडोज ३.० रंगीत.

1990 मध्ये लॉन्च झालेल्या हार्ड ड्राइव्हची आवश्यकता असलेली पहिली विंडोज. विंडोज 3 ही पहिली आवृत्ती होती ज्याला अधिक व्यापक यश मिळाले आणि Apple च्या मॅकिंटॉश आणि कमोडोर अमिगा ग्राफिकल यूजर इंटरफेसला आव्हान देणारे मानले जाते, ज्यात पीसी-सुसंगत उत्पादकांकडून संगणकावर पूर्व इंस्टाल केले जात आहे. जेनिथ डेटा सिस्टम्स.

Windows 3 ने विंडोजमध्ये MS-DOS प्रोग्राम चालवण्याची क्षमता सादर केली, ज्याने लेगसी प्रोग्राममध्ये मल्टीटास्किंग आणले आणि 256 रंगांना सपोर्ट दिले जे इंटरफेसला अधिक आधुनिक, रंगीबेरंगी फॉरमॅट आणते.

अधिक महत्त्वाचे - किमान एकूण मानवी वेळ वाया गेलेल्या - याने कार्ड-मूव्हिंग टाइमसिंक (आणि माऊस वापर इंस्ट्रक्टर) सॉलिटेअर सादर केले.

विंडोज ३.१

Minesweeper सह Windows 3.1.

Windows 1 आणि 2 दोन्हीमध्ये पॉइंट रिलीझ अपडेट होती, परंतु 1992 मध्ये रिलीज झालेला Windows 3.1 लक्षणीय आहे कारण त्याने टूटाइप फॉन्ट सादर केले ज्यामुळे Windows ला प्रथमच एक व्यवहार्य प्रकाशन मंच बनवले.

माइन्स्वीपरनेही पहिले हजेरी लावली. Windows 3.1 ला रन करण्यासाठी 1MB RAM ची आवश्यकता आहे आणि समर्थित MS-DOS प्रोग्राम्सना प्रथमच माऊसने कंट्रोल करण्याची अनुमती दिली आहे. Windows 3.1 हे CD-ROM वर वितरीत केले जाणारे पहिले Windows देखील होते, जरी एकदा हार्ड ड्राइव्हवर इंस्टाल केल्यावर ते फक्त 10 ते 15MB घेते (एक CD सामान्यतः 700MB पर्यंत संचयित करू शकते).

विंडोज ९५

विंडोज ९५: अरे हॅलो स्टार्ट मेनू.

नावाप्रमाणेच, Windows 95 ऑगस्ट 1995 मध्ये आला आणि त्यासोबत पहिले स्टार्ट बटण आणि स्टार्ट आणले.

याने "प्लग अँड प्ले" ची संकल्पना देखील सादर केली - पेरीफेरल कनेक्ट करा आणि ऑपरेटिंग सिस्टम त्यासाठी योग्य ड्रायव्हर्स सर्च ते आणि ते कार्य करते. अशी कल्पना होती; ते नेहमी सरावात काम करत नाही.

Windows 95 ने 32-बिट वातावरण, टास्क बार देखील सादर केले आणि मल्टीटास्किंगवर लक्ष केंद्रित केले. MS-DOS ने अजूनही Windows 95 साठी महत्वाची भूमिका बजावली, ज्यासाठी काही प्रोग्राम्स आणि एलिमेंट चालवणे आवश्यक होते.

इंटरनेट एक्सप्लोररने विंडोज 95 वर देखील पदार्पण केले, परंतु विंडोज 95 प्लसची आवश्यकता असल्याने ते डीफॉल्ट इंस्टाल केले गेले नाही! पॅक Windows 95 च्या नंतरच्या आवर्तनांमध्ये IE चा समावेश डीफॉल्टनुसार केला गेला, कारण त्या वेळी Netscape Navigator आणि NCSA Mosaic लोकप्रिय होते.

विंडोज ९८

Windows 98, शेवटची उत्तम DOS-आधारित Windows.

जून 1998 मध्ये रिलीज झालेले, Windows 98 ने Windows 95 वर तयार केले आणि सोबत IE 4, Outlook Express, Windows Address Book, Microsoft Chat आणि NetShow Player आणले, जे Windows 98 द्वितीय आवृत्तीमध्ये Windows Media Player 6.2 ने 1999 मध्ये बदलले.

Windows 98 ने इतर गोष्ठीबरोबरच Windows Explorer मध्ये बॅक आणि फॉरवर्ड नेव्हिगेशन बटणे आणि अॅड्रेस बार सादर केला. सर्वात मोठा बदल म्हणजे संगणक एलिमेंट आणि अॅक्सेसरीजसाठी विंडोज ड्रायव्हर मॉडेलचा परिचय - विंडोजच्या भविष्यातील सर्व आवृत्त्यांना सपोर्ट देण्यासाठी एक ड्रायव्हर.

Windows 98 मध्ये USB सपोर्ट खूप सुधारले गेले आणि USB हब आणि USB माऊस सह त्याचा व्यापक अवलंब केला.

विंडोज एमई (ME)

Windows ME.

अनेकांनी Windows मालिकेतील कमी पॉईंट्स मानले - किमान, त्यांनी Windows Vista पाहेपर्यंत - Windows Millennium Edition ही MS-DOS वर आधारित असलेली शेवटची Windows होती आणि Windows 9x लाइनमधील शेवटची.

सप्टेंबर 2000 मध्ये रिलीज झालेली, एंटरप्राइझ मार्केटला उद्देशून Windows 2000 सोबत जोडलेली ही ग्राहक-उद्दिष्ट ऑपरेटिंग सिस्टम होती. अधिक ऑटोमॅटिक सिस्टिम पुनर्प्राप्ती साधनांसह काही महत्वाच्या संकल्पना ग्राहकांना सादर केल्या.

IE 5.5, Windows Media Player 7 आणि Windows Movie Maker हे सर्व प्रथमच दिसले. विंडोज एक्सप्लोररमध्ये ऑटोकम्प्लीट देखील दिसू लागले, परंतु ऑपरेटिंग सिस्टिम बग्गी, योग्यरित्या इंस्टाल करण्यात अयशस्वी आणि सामान्यतः खराब असण्याबद्दल कुप्रसिद्ध होती.

विंडोज 2000

Windows 2000 हे ME चे एंटरप्राइझ ट्विन होते.

ME चे एंटरप्राइझ ट्विन, Windows 2000 फेब्रुवारी 2000 मध्ये रिलीज झाले आणि मायक्रोसॉफ्टच्या व्यवसायाभिमुख सिस्टिम Windows NT वर आधारित होते आणि नंतर Windows XP साठी आधार बनले.

विंडोज 2000 मध्ये मायक्रोसॉफ्टच्या ऑटोमॅटिक अपडेटने महत्वाची भूमिका बजावली आणि हायबरनेशनला सपोर्ट देणारी पहिली विंडोज बनली.

विंडोज एक्सपी

विंडोज एक्सपी आजही टिकून आहे.

विंडोज एक्सपी ऑक्टोबर 2001 मध्ये रिलीझ करण्यात आली आणि मायक्रोसॉफ्टची एंटरप्राइझ लाइन आणि ऑपरेटिंग सिस्टिमची ग्राहक लाइन एकाच छताखाली आणली गेली.

जाहिरात

हे Windows 2000 सारख्या Windows NT वर आधारित होते, परंतु Windows ME वरून ग्राहक-अनुकूल एलिमेंट आणले होते. स्टार्ट मेनू आणि टास्क बारला एक व्हिज्युअल ओवरहॉल मिळाला, ज्याने परिचित हिरवे स्टार्ट बटण, निळा टास्क बार आणि व्हिस्टा वॉलपेपर, विविध छाया आणि इतर व्हिज्युअल इफेक्टसह आणले.

ClearType, जे LCD स्क्रीनवर टेक्स्ट वाचणे सोपे करण्यासाठी डिझाइन केले गेले होते, जसे की अंगभूत सीडी बर्निंग, CD आणि इतर माध्यमांमधून ऑटोप्ले, तसेच विविध ऑटोमॅटिक अपडेट आणि रिकव्हरी टूल सादर केली गेली, जी Windows ME च्या विपरीत प्रत्यक्षात काम करत होती.

Windows XP ही सर्वात जास्त काळ चालणारी मायक्रोसॉफ्ट ऑपरेटिंग सिस्टिम होती, तीन प्रमुख अपडेट आणि सपोर्ट एप्रिल 2014 पर्यंत - त्याच्या मूळ प्रकाशन तारखेपासून 13 वर्षे पर्यंत. Windows XP बंद असतानाही अंदाजे 430m PC वर वापरले जात होते.

त्याची सर्वात मोठी समस्या सुरक्षा होती: जरी त्यात एक फायरवॉल अंगभूत आहे, परंतु ते डीफॉल्टनुसार बंद केले गेले. Windows XP ची प्रचंड लोकप्रियता हॅकर्स आणि गुन्हेगारांसाठी वरदान ठरली, ज्यांनी त्याच्या त्रुटींचा फायदा घेतला, स्पेशल तः इंटरनेट एक्सप्लोररमध्ये, निर्दयीपणे - "विश्वसनीय संगणन" सबक्रम सुरू करण्यासाठी बिल गेट्सचे नेतृत्व आणि त्यानंतरच्या सर्व्हिस पॅक अपडेट जारी करणे ज्यामुळे XP कठोर झाले. हल्ल्याच्या विरोधात जोरदार.

विंडोज व्हिस्टा

Windows Vista, Windows ME पेक्षा वादातीत वाईट.

जानेवारी 2007 मध्ये Windows Vista ने बदलण्यापूर्वी Windows XP हा अभ्यासक्रम जवळपास सहा वर्षे राहिला. Vista ने पारदर्शक एलिमेंट , सर्व आणि सुरक्षिततेवर अधिक लक्ष केंद्रित करून Windows चे फॉरमॅट अपडेट केले. "लॉन्गहॉर्न" या सांकेतिक नावाखाली त्याचा विकास अडचणीत आला होता, उत्पादनात येण्यासाठी महत्वाकांक्षी एलिमेंटचा त्याग केला गेला होता.

जाहिरात

ते बग्गी होते, यूजर वर "यूजर खाते कंट्रोल " इंटरनल अॅप परवानग्यांसाठी शेकडो विनंत्यांचे ओझे होते - विश्वासाई संगणन सबक्रमाचा रिझल्ट ज्याचा

अर्थ आता वापरकर्त्यांना विविध बदल करण्याच्या प्रोग्रामच्या प्रयत्नांना मंजूरी किंवा नामंजूर करावी लागली.

UAC ची समस्या अशी होती की यामुळे आत्मसंतुष्टता निर्माण झाली, लोक जवळजवळ कोणत्याही गोष्टीवर "होय" क्लिक करतात - सुरक्षा परत UAC पूर्व स्थितीत घेऊन. जुन्या संगणकांवर "व्हिस्टा रेडी" असे मानले जात असतानाही ते हळू चालले - एक लेबलिंग ज्याने त्यावर खटला भरला कारण Vista च्या सर्व आवृत्त्या त्या लेबलसह PC वर चालू शकत नाहीत.

मायक्रोसॉफ्टच्या डायरेक्टएक्स 10 तंत्रज्ञानाच्या Vista च्या समावेशामुळे पीसी गेमर्सना चालना मिळाली.

Windows Media Player 11 आणि IE 7, Windows Defender सोबत अंटी स्पायवेअर प्रोग्राम डेब्यू केले. Vista मध्ये स्पीच रेकग्निशन, विंडोज डीव्हीडी मेकर आणि फोटो गॅलरी, तसेच डीव्हीडीवर वितरीत होणारी पहिली विंडोज आहे. नंतर विश्वासविरोधी तपासांना प्रतिसाद म्हणून Windows Media Player शिवाय Windows Vista ची आवृत्ती तयार केली गेली.

विंडोज ७

विंडोज 7 हे सर्व काही विंडोज व्हिस्टा असायला हवे होते.

विंडोज व्हिस्टा काय असायला हवे होते म्हणून अनेकांच्या विचारात, विंडोज 7 पहिल्यांदा ऑक्टोबर 2009 मध्ये रिलीझ करण्यात आले होते. व्हिस्टाला आलेल्या सर्व समस्या आणि टीका सोडवण्याचा हेतू होता, त्याच्या स्वरूपामध्ये थोडासा बदल आणि युजर -अनुकूल वैशिष्ट्यांवर इंटिग्रिटी आणि कमी "संवाद बॉक्स ओव्हरलोड".

ती जलद, अधिक स्थिर आणि वापरण्यास सोपी होती, बहुतेक वापरकर्ते आणि व्यवसाय Windows XP वरून अपग्रेड करणारी ऑपरेटिंग सिस्टीम बनून, Vista पूर्णपणे सोडून.

हस्तलेखन ओळख 7 मध्ये डेब्यू झाली, जसे की स्क्रीनच्या शीर्षस्थानी किंवा बाजूंना विंडो "स्नॅप" करण्याची क्षमता, जलद अधिक ऑटोमॅटिक विंडो आकारमानास अनुमती देते.

Windows 7 ने युरोपमध्ये IE च्या प्री-इंस्टॉलिंगवर अविश्वास तपासणीसह मायक्रोसॉफ्टला फटका बसला, ज्यामुळे नवीन वापरकर्त्यांना ब्राउझर बॅलेट स्क्रीन दर्शविण्यात आली ज्यामुळे त्यांना प्रथम बूट झाल्यावर कोणता ब्राउझर इंस्टाल करायचा हे निवडण्याची परवानगी दिली गेली.

विंडोज 8

विंडोज 8 ने कीबोर्ड आणि माऊसपेक्षा टचस्क्रीन वर अधिक लक्ष केंद्रित केले.

ऑक्टोबर 2012 मध्ये रिलीज झालेला, विंडोज 8 हा मायक्रोसॉफ्टचा विंडोज इंटरफेसचा सर्वात मूलगामी बदल होता, ज्याने स्टार्ट बटण आणि स्टार्ट मेनू अधिक टचफ्रेंडली स्टार्ट स्क्रीनच्या बाजूने सोडला.

जाहिरात

नवीन टाइल केलेल्या इंटरफेसमध्ये प्रोग्राम सिम्बॉल्स आणि थेट टाइल्स दिसल्या, जे सामान्यतः "विजेट्स" शी संबंधित इनफार्मेशन एका दृष्टीक्षेपात

प्रदर्शित करतात, प्रोग्राम आणि चिन्हांच्या लिस्ट बदलतात. एक डेस्कटॉप अजूनही समाविष्ट केला होता, जो विंडोज 7 सारखा दिसत होता.

Windows 8 Windows च्या मागील आवृत्त्यांपेक्षा वेगवान होता आणि नवीन, खूप वेगवान USB 3.0 उपकरणांसाठी सपोर्ट समाविष्ट केले होते.

Windows Store, जे युनिव्हर्सल विंडोज ॲप्स ऑफर करते जे केवळ पूर्ण-स्क्रीन मोडमध्ये चालते, सादर केले गेले. विंडोजच्या इतर पुनरावृत्तींप्रमाणे तृतीय-पक्षांकडून प्रोग्राम अजूनही इंस्टाल केले जाऊ शकतात, परंतु ते फक्त विंडोजच्या पारंपारिक डेस्कटॉप इंटरफेसमध्ये एक्सेस करू शकतात.

मूलगामी दुरुस्तीचे अनेकांनी स्वागत केले नाही. मायक्रोसॉफ्टने टचस्क्रीन सपोर्ट आणि डेस्कटॉप वापरकर्ते यांच्यात एक बारीक रेषा चालवण्याचा प्रयत्न केला, परंतु शेवटी डेस्कटॉप वापरकर्ते पारंपरिक माउस आणि कीबोर्डच्या सहाय्याने विंडोज कंट्रोल करू इच्छित होते आणि टचस्क्रीनने नव्हे तर विंडोज 8 एक पाऊल मागे पडले आहे असे वाटले.

आयपॅड आणि स्मार्टफोन सारख्या टॅब्लेटच्या पॅरलल वाढ असूनही, 2010 च्या अखेरीस पीसीची विक्री सुरू झाली होती, तरीही त्याचा टच-ओरिएंटेड इंटरफेस सबयुक्त किंवा अगदी आवश्यक बनवण्यासाठी वापरात असलेल्या किंवा ऑफरवर असलेल्या काही टचस्क्रीन होत्या.

विंडोज आरटी, जे एआरएम-आधारित प्रोसेसरवर चालते जे परंपरेने स्मार्टफोन आणि नॉन-पीसी टॅब्लेटमध्ये आढळतात, त्याच वेळी मायक्रोसॉफ्ट सरफेस टॅब्लेटसह विंडोज 8 सोबत सादर केले गेले.

ते Windows 8 सारखे दिसले आणि वाटले, परंतु थर्ड पार्टी ॲप्ससाठी पूर्णपणे Windows Store वर अवलंबून राहून पारंपारिक Windows ॲप्लिकेशन्स चालवू शकले नाहीत.

विंडोज ८.१

विंडोज 8.1 आणि स्टार्ट बटणाचे उत्कृष्ट पुनरागमन: ऑक्टोबर 2013 मध्ये विंडोज 8 ला एक फ्री पॉइंट रिलीझ सादर केले गेले, विंडोज 8.1 ने मायक्रोसॉफ्टकडून वार्षिक सॉफ्टवेअर अपडेट्सकडे शिफ्ट केले आणि मायक्रोसॉफ्टच्या नवीन व्हिज्युअल इंटरफेसभोवती यू-टर्न मधील पहिले पाऊल समाविष्ट केले.

विंडोज 8.1 ने स्टार्ट बटण पुन्हा सादर केले, ज्याने विंडोज 8.1 च्या डेस्कटॉप व्हॅमथून स्टार्ट स्क्रीन आणली. वापरकर्ते विंडोज 8.1 च्या डेस्कटॉपवर थेट बूट करणे देखील निवडू शकतात, जे टच-केंद्रित स्टार्ट स्क्रीनपेक्षा माउस आणि कीबोर्डसह डेस्कटॉप संगणक वापरणाऱ्यांसाठी अधिक योग्य होते.

विंडोज १०

Windows 10 सह, Microsoft Windows 8 साठी तयार केलेली काही टच आणि टॅब्लेट वैशिष्ट्ये ठेवण्याचा प्रयत्न करत आहे, त्यांना परिचित स्टार्ट मेनू आणि डेस्कटॉपसह एकत्र करा आणि हे सर्व अधिक सुरक्षिततेसह सुधारित ऑपरेटिंग सिस्टिमच्या शीर्षस्थानी चालवा, नवीन ब्राउझर ,

Cortana सहाय्यक, जाता-जाता संपादनासाठी ऑफिसची स्वतःची आवृत्ती आणि जीवन सोपे बनवण्याच्या हेतूने भरपूर नवीन वैशिष्ट्ये.

अर्थात, याचा अर्थ असा आहे की ते वापरणे खूप वेगळे आहे, तुम्ही Windows 7, Windows 8 किंवा Windows XP वरून येत असलात तरीही. तुमचा पीसी बंद करण्यासाठी तुम्हाला नवीन ठिकाणी पहावे लागेल. त्या वर, Windows 10 ही फक्त पीसी ऑपरेटिंग सिस्टमपेक्षा अधिक आहे; विंडोज फोनवरही तेच चालेल – आणि लहान टॅब्लेटवरही, कारण 6-इंचाचा फोन आणि 7-इंचाचा टॅब्लेट अशी काही वेगळी उपकरणे नाहीत.

मायक्रोसॉफ्टची अपेक्षा आहे की लोकांनी एक अब्ज उपकरणांवर Windows 10 ठेवला पाहिजे (ज्याने अधिक ॲप डेव्हलपरना किमान Windows फोन आणि टॅब्लेट तसेच Xbox One आणि HoloLens साठी त्यांचे ॲप्स तयार करण्यासाठी एक नजर टाकण्यास प्रोत्साहित केले पाहिजे).

स्टार्ट मेनू डेव्हलपिंग

Windows 8 ची पूर्ण-स्क्रीन स्टार्ट स्क्रीन पुन्हा Windows 10 मध्ये स्टार्ट मेनू बनली आहे जी दोन्ही पर्यायांपैकी सर्वोत्तम एकत्र करण्याचा प्रयत्न करते. एक स्क्रॉलिंग स्टार्ट मेनू जो एका कॉलम पुरता मर्यादित आहे, जंप लिस्ट आणि अतिरिक्त पर्यायांसाठी फ्लायआउट मेनूसह, वारंवार वापरल्या जाणाऱ्या आणि अलीकडे इंस्टाल केलेल्या प्रोग्राममध्ये विपार्ट लेला, सर्व ॲप्लिकेशन्सच्या स्क्रॉलिंग दृश्यावर स्विक करण्याच्या पर्यायासह, कॅरक्टर क्रमानुसार क्रमवारी (सॉर्टिंग) लावलेला आहे.

पण एक अतिरिक्त पॅन देखील मिळवा, जिथे विंडोज 8-स्टाइल च्या टाइल्स पिन करू शकतात, थेट टाइल्सच्या 'रोटेटिंग 3D क्यूब' ॲनिमेशनसह पूर्ण करा. स्टार्ट मेनू मोठ्या आकारासाठी ड्रॅग करा किंवा पूर्ण स्क्रीनवर सेट करा.

डेस्कटॉप बॅकग्राऊंड

डेस्कटॉपचा आणखी एक एलिमेंट म्हणजे बॅकग्राऊंड . ही फक्त एक प्रतिमा आहे जी स्क्रीनच्या मागील बाजूस दिसते. बहुतेक संगणक पूर्व-निवडलेल्या बॅकग्राऊंड सह येतात, परंतु ते कोणत्याही प्रतिमेमध्ये बदलतात.



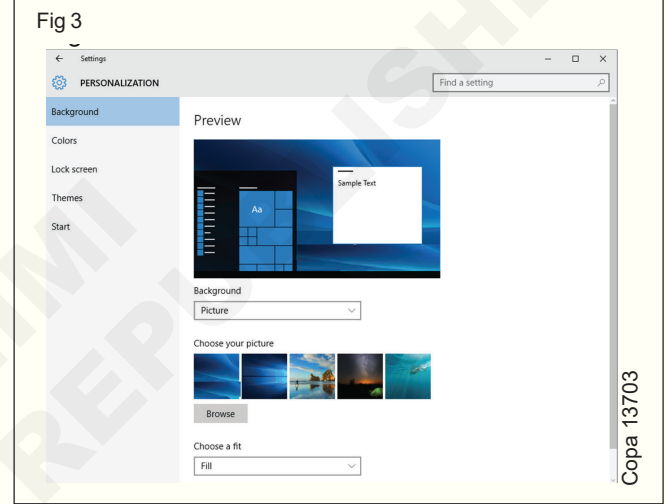
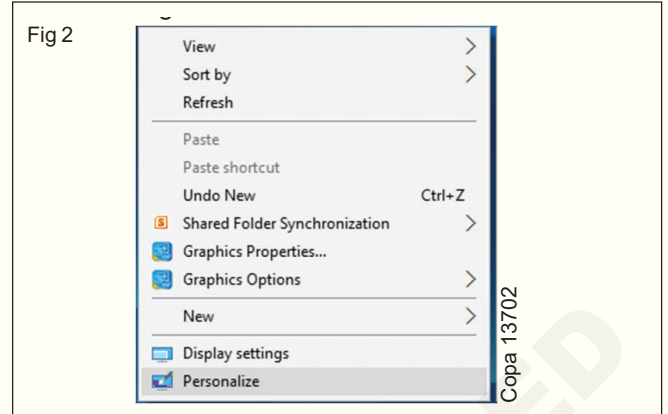
बॅकग्राऊंड बदलण्यासाठी, या स्टेप्स चे अनुसरण करा:

- 1 बॅकग्राऊंड वर उजवे-क्लिक करा आणि "वैयक्तिकृत करा" निवडा
- 2 वैयक्तिकरण c विंडोमधून, पूर्व-निवडलेल्या चित्रांच्या मालिकेतून निवडा किंवा तुमच्या स्वतःसाठी ब्राउझ करा.

पिक्चर निवडल्यानंतर, बॅकग्राऊंड ऑटोमैटिक बदलेल.

सुरुवातीचा मेन्यू

स्पेसिफिक ॲप्लिकेशन सर्च त असल्यास, स्टार्ट मेनू उघडा आणि "सर्व ॲप्लिकेशन " वर क्लिक करा. हे संगणकावर इंस्टाल केलेल्या सर्व ॲप्लिकेशन ची कॅरक्टर माला लिस्ट उघडेल.



फाइल ॲक्सप्लोरर

तुम्ही स्पेसिफिक डॉक्युमेंट्स सर्च त असल्यास, टास्कबारवरील फोल्डर चिन्हावर क्लिक करून फाइल ॲक्सप्लोरर वापरणे हा दुसरा पर्याय आहे.

फाइल ॲक्सप्लोरर विंडोमध्ये, सर्व फोल्डर्स आणि डॉक्युमेंट्स ब्राउझ करा.

व्हर्चुअल डेस्कटॉप

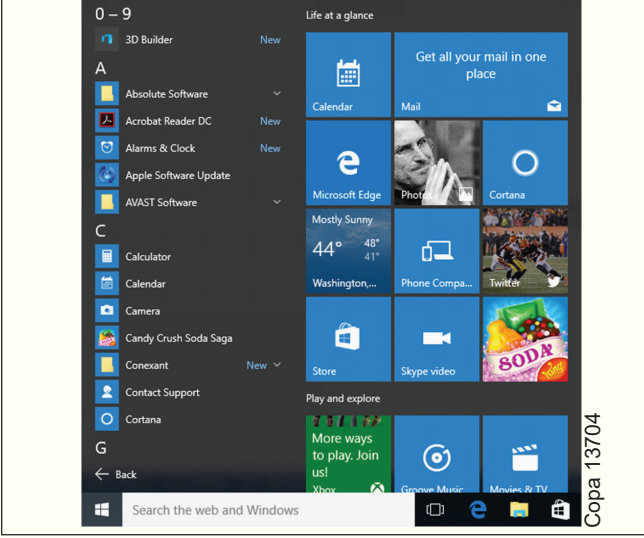
Windows 10 च्या नवीन वैशिष्ट्यांपैकी एक म्हणजे व्हर्चुअल डेस्कटॉपची भर. हे तुम्हाला मल्टिपल डेस्कटॉप स्क्रीन ठेवण्याची परवानगी देते जेथे खुल्या खिडक्या व्यवस्थित ठेवाव्यात.

व्हर्चुअल डेस्कटॉप जोडण्यासाठी, या स्टेप्स चे अनुसरण करा:

- 1 टास्कबारवरील टास्क व्ह्वर क्लिक करा
- 2 खालच्या-उजव्या कोपर्यात "नवीन डेस्कटॉप" पर्यायावर क्लिक करा. तुम्ही Task View वर पुन्हा क्लिक करून नवीन डेस्कटॉपमध्ये एक्सेस करू शकता किंवा हटवू शकता.

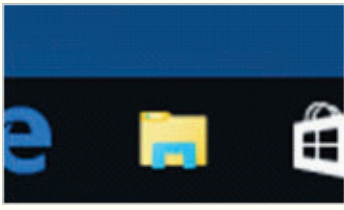
Cortana सर्च आणि नियंत्रणाप्रमाणेच मदत करतेCortana, Windows Phone असिस्टंट, Windows 10 मध्ये टास्कबारवर एक सर्च पॅन म्हणून दर्शविले जाते, जे 'Hey Cortana' म्हणून देखील ट्रिगर करू शकते - आणि स्टार्ट मेनू सर्च ताना.

Fig 4



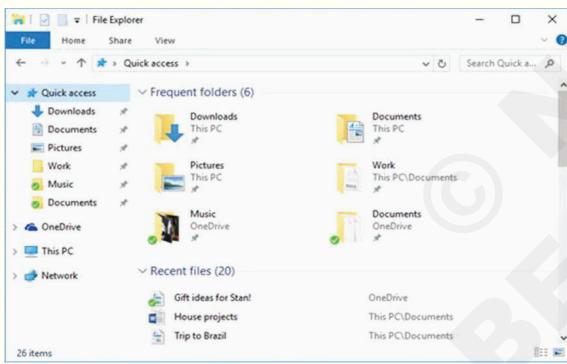
Copa 13704

Fig 5



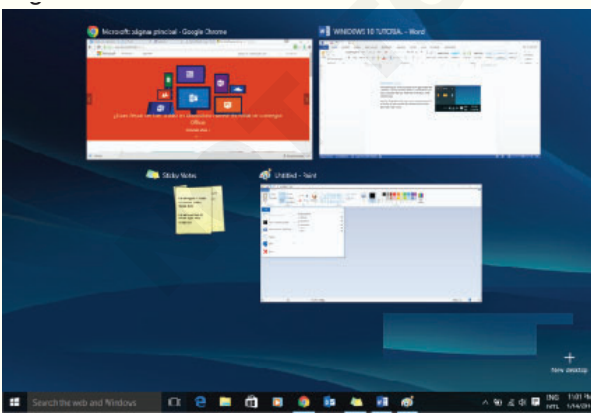
Copa 13705

Fig 6



Copa 13706

Fig 7



Copa 13707

यामुळे ऑप्स इंस्टाल केले आहेत, डॉक्युमेंट्स मध्ये एक्सेस आहे, ऑप्स स्टोअरमधून इंस्टाल करू शकतात, वेबवरील सर्च रिझल्ट आणि Cortana सह समाकलित होणाऱ्या ऑप्स आणि सर्विस सह इतर इनफार्मेशन ची रेन्ज मिळते.

Fig 8



Copa 13708

Cortana ऍक्टिव्ह करणे

Cortana ऍक्टिव्ह नसल्यास, Cortana सेटिंग्जमध्ये एक्सेस करण्यासाठी टास्कबार शोधामध्ये "Cortana" टाइप करून किंवा मेनूच्या डाव्या बाजूला असलेल्या "Gear" चिन्हावर क्लिक करून ते चालू करू शकता.

Fig 9



Copa 13709

Cortana ऍक्टिव्ह केल्यानंतर, ते अनुभव वैयक्तिकृत करण्यासाठी इनफार्मेशन गोळा करण्यास स्टार्ट करेल.

टास्क स्विचर

बऱ्याच Windows वापरकर्त्यांना सर्व चालू ऑप्स पाहण्यासाठी आणि त्यामध्ये स्विच करण्यासाठी Alt-Tab कीबोर्ड कॉम्बिनेशन माहित नाही, म्हणून मोठ्या थंबनेल्ससह पुन्हा डिझाइन केलेले टास्क स्विचर असणे, Windows 10 त्यांना सर्च प्यात मदत करण्यासाठी टास्कबारमध्ये टास्क व्ह आयकॉन देखील ठेवते. ते

Fig 10

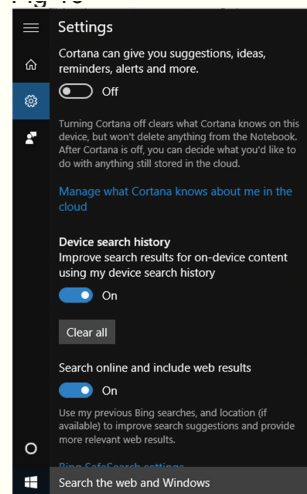
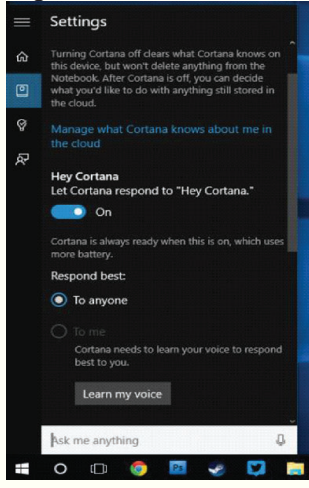


Fig 11

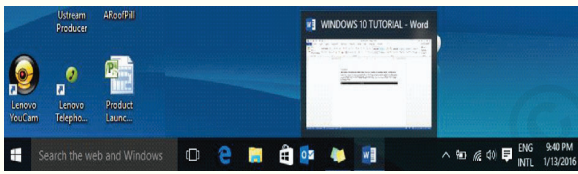


Copa 137011

**टास्कबार:** Windows 10 टास्कबार स्क्रीनच्या तळाशी बसून यूजर ला स्टार्ट मेनू तसेच वारंवार वापरल्या जाणाऱ्या ऍप्लिकेशन्सच्या आयकॉनमध्ये एक्सेस देतो. उजव्या बाजूला, टास्कबारमध्ये नोटिफिकेशन एरिया आहे जे यूजर ला इंटरनेट कनेक्शनची स्थिती किंवा लॅपटॉप बॅटरी चार्ज करण्यासारख्या विविध गोष्टींची इनफॉर्मेशन देते.

टास्कबारच्या मध्यभागी असलेले चिन्ह हे "पिन केलेले" ऍप्लिकेशन्स आहेत, जे तुम्ही वारंवार वापरत असलेल्या ऍप्लिकेशन्समध्ये त्वरित एक्सेस मिळवण्याचा एक पाथ आहे. "पिन केलेले" ऍप्लिकेशन्स तुम्ही "अनपिन" करेपर्यंत टास्कबारमध्ये राहतील.

Fig 12



1 ली स्टेप :मध्ये पिन करू इच्छित ऍप्लिकेशन शोधा

सुरुवातीचा मेन्यू.

स्टेप २: ऍप्लिकेशन वर उजवे-क्लिक करा.

स्टेप ३:मेनूच्या शीर्षस्थानी "अधिक" पर्याय निवडा.

स्टेप ४:"टास्कबारवर पिन करा" पर्याय निवडा.

टास्कबारमधून ऍप्लिकेशन अनपिन करा

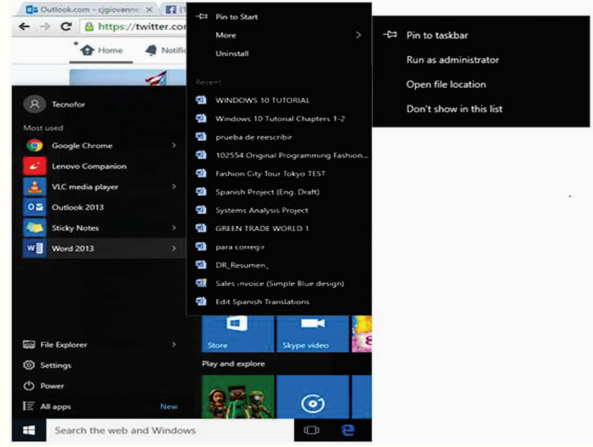
ते "अनपिन" करण्यासाठी, फक्त टास्कबारमधील चिन्हावर उजवे-क्लिक करा आणि "टास्कबारमधून अनपिन करा" निवडा. तुम्हाला पाहिजे तेव्हा तुम्ही ते पुन्हा "पिन" करू शकता.

नोटिफिकेशन एरिया

नोटिफिकेशन एरिया टास्कबारच्या उजव्या बाजूला स्थित आहे. हे तुमच्या संगणकावरून तुमचे इंटरनेट कनेक्शन किंवा व्हॉल्यूम पातळी यांसारख्या विविध टाइप च्या सूचना दाखवते.

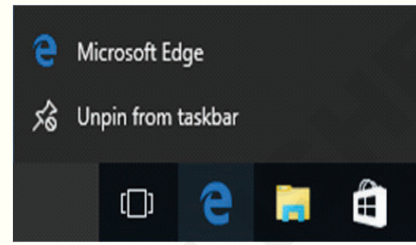
सुरुवातीला, नोटिफिकेशन एरिया मर्यादित प्रमाणात चिन्ह दर्शविते. परंतु इतर सिम्बॉल्स पाहण्यासाठी तुम्ही त्याच्या डाव्या बाजूच्या वरच्या बाणावर क्लिक करू शकता.

Fig 13



Copa 137013

Fig 14

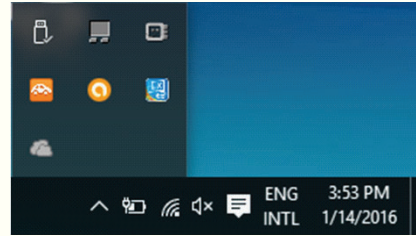


Copa 137014

### सॅप सहाय्य

कारण सर्व ॲप्स आणि प्रोग्राम्स डेस्कटॉपवरील विंडोजमध्ये चालतात, त्याऐवजी स्टोरमधील आधुनिक ॲप्स त्यांच्या स्वतःच्या जागेत असतात आणि यापुढे स्क्रीनवर दुसरे ॲप आणण्यासाठी आणि स्प्लिट व्हू मिळवण्यासाठी स्क्रीनच्या डाव्या काठावर ड्रॅग करू शकत नाहीत. त्याऐवजी, परिचित सॅप व्हू मिळवण्यासाठी विंडो स्क्रीनच्या कोपऱ्यात ड्रॅग करा.

Fig 15



Copa 137015

आता स्क्रीनचे चारही कोपरे वापरा जर प्रत्येक विंडोने अर्धा ऐवजी एक चतुर्थांश स्क्रीन घ्यायची असेल आणि तुम्ही नुकतीच ड्रॅग केलेल्या खिडकीने न भरलेली जागा तुमच्या इतर खिडक्यांची लघुप्रतिमा दाखवते जेणेकरून ते सॅप करणे सोपे होईल. पुढील एक ठिकाणी.

### ॲक्शन सेंटर

जर आम्ही Windows Phone 8.1 (किंवा Android आणि/किंवा iOS) वापरला असेल, तर आम्ही सूचना केंद्र वापरत असल्यास स्क्रीनच्या शीर्षस्थानी खाली ड्रॅग करू शकतो.

Windows 10 स्क्रीनच्या उजवीकडे ठेवते, जेथे चार्ज बार Windows 8 मध्ये होता, शीर्षस्थानी विविध ॲप्सच्या सूचना आणि द्रुत प्रवेशासाठी तळाशी विविध सेटिंग्ज बटणांची निवड.



## कमांड प्रॉम्प्ट

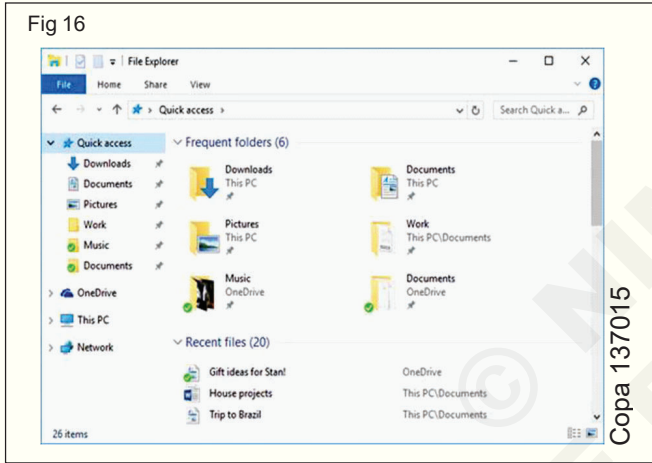
आपल्यापैकी जे कमांड प्रॉम्प्ट वापरतात ते 1990 च्या दशकापासून जवळजवळ समान अनुभवाने अडकले आहेत, परंतु Windows 10 मध्ये शेवटी कमांड प्रॉम्प्ट विंडोचा आकार बदलू शकतो आणि कमांड प्रॉम्प्टवर कॉपी आणि पेस्ट करण्यासाठी परिचित कीबोर्ड शॉर्टकट वापरू शकतो.

हे ग्राउंड ब्रेकिंगपासून दूर आहे परंतु अनेक वर्षांच्या निराशेनंतर ही एक अतिशय स्वागतार्ह सुधारणा आहे.

## फाइल एक्सप्लोरर

फाइल एक्सप्लोरर हे विंडोज ऑपरेटिंग सिस्टमद्वारे फोल्डर्स आणि फाइल्स ब्राउझ करण्यासाठी वापरलेले फाइल मॅनेजमेंट एप्लिकेशन आहे. हे यूजर ला संगणकात साठवलेल्या फाइल्समध्ये नेव्हिगेट करण्यासाठी आणि एक्सेस करण्यासाठी ग्राफिकल इंटरफेस प्रदान करते.

फाइल एक्सप्लोररमध्ये एक्सेस करण्याचा मुख्य पाथ म्हणजे टास्कबारमधील फोल्डर चिन्हावर क्लिक करणे. चिन्हावर क्लिक केल्यानंतर, द



फाइल एक्सप्लोरर विंडो उघडेल.

स्टार्टिंग फाइल एक्सप्लोरर विंडोमध्ये खालील विपार्ट आहेत:

- 1 फाइल एक्सप्लोरर रिबन, जे Microsoft Office मध्ये वैशिष्ट्यीकृत रिबनसारखे दिसते. रिबनमध्ये तुमच्या फाइल्स आणि फोल्डर्ससह सामान्य कार्यासाठी बटणे असतात.
- 2 नेव्हिगेशन पॅन तुम्हाला तुमच्या डॉक्युमेंट्स आणि चित्रांच्या लायब्ररीमध्ये तसेच तुमच्या स्टोरेज डिव्हाइसेसमध्ये एक्सेस देतो. यात वारंवार वापरले जाणारे फोल्डर आणि नेटवर्क उपकरणे देखील आहेत.
- 3 उजवीकडील फ्रिक्वेंट फोल्डर्स विभागात तुम्ही अलीकडे काम केलेल्या फोल्डर्समध्ये द्रुत प्रवेशासाठी अनुमती देण्यासाठी वैशिष्ट्यीकृत केले आहे.
- 4 विंडोच्या खालच्या भागात अलीकडील फाइल्स विभागात तुम्ही अलीकडे उघडलेल्या फाइल्स आणि डॉक्युमेंट्स चा समावेश आहे.

## न्यू एज ब्राउझर

क्रोम आणि फायरफॉक्स सारख्या जलद गतीने चालणाऱ्या ब्राउझरशी संपर्क साधण्यासाठी, मायक्रोसॉफ्टने आपल्या ब्राउझरला बेसिक गोष्टींकडे

नेले, अनेक वर्षांचे कोड जे वेब मानकांमध्ये बसत नाहीत आणि एक दुबळा, वेगवान ब्राउझर बनवला.

हे काम प्रगतीपथावर आहे - त्याला जाहिरात-ब्लॉकिंग एक्स्टेंशनच्या उदाहरणासारख्या गोष्टींसाठी सपोर्ट मिळणार नाही आणि एखाद्या मित्राला पाठवण्यासाठी वेब पेजवर नोट्स स्क्राइबल करू शकतात आणि वेब पेजमधून सबयुक्त इनफार्मेशन काढण्यासाठी Edge मध्ये Cortana अंगभूत आहे, जसे की रेस्टॉरेंटचा फोन नंबर किंवा उघडण्याचे तास.

मध्यम सारख्या साइट्स ज्या IE सह योग्यरित्या कार्य करत नाहीत त्या अधिक चांगल्या दिसल्या पाहिजेत आणि एजमध्ये अधिक वैशिष्ट्ये आहेत.

## मल्टिपल डेस्कटॉप

यूजर ला बऱ्याच विंडोची व्यवस्था करणे आवश्यक आहे आणि मल्टिपल मॉनिटर्स नाहीत, युजर त्यांना मल्टिपल व्हर्च्युअल डेस्कटॉपवर ठेवू शकतो. आणि नेहमीप्रमाणे अॅप्स दरम्यान हलविण्यासाठी Alt-Tab वापरू शकतो आणि नंतर Windows-Ctrl आणि डेस्कटॉप दरम्यान हलविण्यासाठी डाव्या आणि उजव्या बाण की.

## शेड्यूल रीस्टार्ट होते

अपडेट लागू करण्यासाठी रीस्टार्ट होण्यापूर्वी सर्व काही पूर्ण करण्यासाठी तुमच्याकडे पंधरा मिनिटे आहेत हे विंडोजने जाहीर केले नाही. ते कधी करायचे हे ठरवण्यासाठी Windows 10 सोडण्याऐवजी, रीस्टार्ट करण्याची आवश्यकता असल्यास आणि त्यासाठी Windows असू शकते.

अपडेट डाऊनलोड झाल्यानंतरच युजर ते करू शकतो. जर यूजर ला रीस्टार्ट करण्यासाठी ठराविक वेळा बंद-मर्यादा हवी असेल, तर त्यांना Windows Update for Business (Windows 10 Pro आणि Enterprise साठी) मधील वैशिष्ट्यांची आवश्यकता असेल जे रीस्टार्ट ब्लॉक करू देते जेणेकरून ते कामाच्या तासांमध्ये किंवा ठराविक तारखांना होणार नाहीत.

## युनिव्हर्सल अॅप्स - ऑफिससह

Windows 10 ला एक नवीन Windows Store मिळते, जेथे डेस्कटॉप प्रोग्राम तसेच आधुनिक Windows अॅप्स डाऊनलोड केले जातात. यापैकी बरेच अॅप्स सार्वत्रिक अॅप्स असतील जे PC, Windows फोन, Xbox One आणि अगदी HoloLens वर समान कोड आहेत, ज्याचा इंटरफेस वेगवेगळ्या स्क्रीन आकारांनुसार बदलतो. Office for Windows अॅप्स जसे Word आणि Excel हे सार्वत्रिक अॅप्स आहेत, जसे Outlook Mail आणि Calendar अॅप्स आहेत.

## सेटिंग्ज आणि कंट्रोल पॅनेल

Windows 8 सेटिंग्ज अॅपने कंट्रोल पॅनेलमध्ये असलेल्या अनेक सेटिंग्ज ताब्यात घेतल्या आहेत आणि त्यात नेव्हिगेट करण्यासाठी आयकॉनसह कंट्रोल पॅनेल-स्टाइल चा इंटरफेस आहे.

सह परंतु नवीन सेटिंग्ज अॅपमध्ये नसलेल्या सेटिंग्जसाठी (किंवा तुम्हाला तिथे गोष्टी सर्च प्याची सवय असल्यास) जुना कंट्रोल पॅनेल इंटरफेस अजूनही आहे.

## Windows 10 - कीबोर्ड शॉर्टकट यूटीएस

बर्याच Windows ऍप्लिकेशन्सप्रमाणे, काही स्पेसिफिक फंक्शन करणे सोपे किंवा जलद करण्यासाठी तुम्ही वापरू शकता असे अनेक कीबोर्ड शॉर्टकट आहेत.

बहुतेक नवीन विंडोज शॉर्टकट अनेक क्रिया करण्यासाठी Windows की () इतर की सह एकत्रितपणे वापरतात. Windows 10 मध्ये वापरले जाणारे काही सर्वात सामान्य किंवा सबयुक्त शॉर्टकट खालीलप्रमाणे आहेत.

#### Windows 10 नेव्हिगेट करण्यासाठी कीबोर्ड शॉर्टकट

हे दाबा	हे करणे
Windows Logo	Toggle the Start menu
Windows Logo+A	Open the Notifications pane
Windows Logo+B	Activate the notification area's Show Hidden Icons arrow (press Enter to display the hidden icons)
Windows Logo+C	Open Cortana for voice commands
Windows Logo+D	Minimize all open windows to display the desktop
Windows Logo+E	Run File Explorer
Windows Logo+F	Display the Start menu and activate the Search box
Windows Logo+H	Display the Share pane
Windows Logo+I	Run the Settings app
Windows Logo+K	Display the Devices pane
Windows Logo+L	Lock your computer
Windows Logo+M	Minimize all windows
Windows Logo+O	Turn the tablet orientation lock on and off
Windows Logo+P	Display the Project pane to configure a second display
Windows Logo+Q	Open Cortana for voice commands
Windows Logo+R	Open the Run dialog box
Windows Logo+S	Open Cortana for keyboard commands
Windows Logo+T	Activate the taskbar icons (use the arrow keys to navigate the icons)
Windows Logo+U	Open the Ease of Access Center
Windows Logo+W	Activate the Search box
Windows Logo+X	Display a menu of Windows tools and utilities
Windows Logo+Z	Display an app's commands (although this works in only some Modern apps)
Windows Logo+=	Open Magnifier and zoom in
Windows Logo+-	Zoom out (if already zoomed in using Magnifier)
Windows Logo+,	Temporarily display the desktop
Windows Logo+Enter	Open Narrator
Windows Logo+Left	Snap the current app to the left side of the screen

हे दाबा	हे करणे
Windows Logo+Right	Snap the current app to the right side of the screen
Windows Logo+Up	Restore a minimized app; maximize a restored app
Windows Logo+Down	Restore a maximized app; minimize a restored app
Windows Logo+PgUp	Move the current app to the left monitor
Windows Logo+PgDn	Move the current app to the right monitor
Windows Logo+PrtSc	Capture the current screen and save it to the Pictures folder
Windows Logo+Ctrl+D	Create a virtual desktop
Windows Logo+Ctrl+Right	Switch to the next virtual desktop
Windows Logo+Ctrl+Left	Switch to the previous virtual desktop
Windows Logo+Ctrl+F4	Close the current virtual desktop
Windows Logo+Tab	Open Task View, which displays thumbnails for each running app as well as the available virtual desktops

## COPA - संगणक बेसिक आणि सॉफ्टवेअर इन्स्टॉलेशन

## विंडोज ऑपरेटिंग सिस्टम इंस्टाल करा (Install Windows operating system)

उद्दिष्ट: या एक्सरसाईस च्या शेवटी तुम्ही सक्षम व्हाल

- BIOS आणि CMOS च्या कार्यांचे डिस्क्रिप्शन करा
- BIOS सेटिंग्ज पाहण्याच्या आणि बदलण्याच्या मेथड्स चे डिस्क्रिप्शन करा
- पार्टीशन आणि फॉर्मॅटिंग चा अर्थ डिस्क्रिप्शन करा
- विंडोज ऑपरेटिंग सिस्टम इन्स्टॉल करण्याच्या प्रक्रियेचे डिस्क्रिप्शन करा

बेसिक इनपुट/आउटपुट सिस्टम (BIOS), ज्याला सिस्टम BIOS, ROM BIOS किंवा PC BIOS म्हणूनही ओळखले जाते, हे फर्मवेअर इंटरफेस परिभाषित करणारे सामान्यतः स्वीकारलेले स्टॅण्डर्ड आहे.

BIOS चा बेसिक उद्देश म्हणजे सिस्टम हार्डवेअर एलिमेंट सुरू करणे आणि टेस्टिंग करणे आणि मास मेमरी डिव्हाइसवरून ऑपरेटिंग सिस्टम लोड करणे. BIOS हे स्पेशल सॉफ्टवेअर आहे जे संगणकाच्या प्रमुख हार्डवेअर एलिमेंटना ऑपरेटिंग सिस्टमसह इंटरफेस करते. हे सहसा मदरबोर्डवरील फ्लॅश मेमरी चिपवर स्टोर केले जाते, परंतु काहीवेळा चिप हा रॉमचा दुसरा टाइप असतो. BIOS हे फर्मवेअर आहे (सॉफ्टवेअर सूचना तुमच्या मदरबोर्डवर असलेल्या चिपवर कायमस्वरूपी रेकॉर्ड केल्या जातात). (पिक्चर 1 पहा).



BIOS ची फंक्शन

BIOS सॉफ्टवेअरमध्ये विविध भूमिका आहेत, परंतु ऑपरेटिंग सिस्टम लोड करणे ही त्याची सर्वात महत्त्वाची भूमिका आहे. ऑपरेटिंग सिस्टमला कंट्रोल सोपवण्यापूर्वी सिस्टम पॉवर अप झाल्यावर किंवा रीस्टार्ट झाल्यावर प्रत्येक वेळी BIOS पीसी हार्डवेअर तपासते आणि सुरू करते. BIOS करत असलेल्या इतर काही सामान्य कार्यांमध्ये हे समाविष्ट आहे:

- सिस्टममधील सर्व भिन्न हार्डवेअर एलिमेंटसाठी एक पॉवर-ऑन सेल्फ-टेस्ट (POST) सर्वकाही योग्यरित्या कार्य करत असल्याची खात्री करण्यासाठी
- संगणकात इंस्टाल केलेल्या वेगवेगळ्या कार्ड्सवर इतर BIOS चिप्स ऍक्टिव्ह करणे - उदाहरणार्थ ग्राफिक्स कार्ड्समध्ये त्यांच्या स्वतःच्या BIOS चिप्स असतात.
- ऑपरेटिंग सिस्टीम वेगवेगळ्या हार्डवेअर उपकरणांना इंटरफेस

करण्यासाठी वापरत असलेल्या निम्न-स्तरीय नित्यक्रमांचा संच प्रदान करणे. ते कीबोर्ड, स्क्रीन आणि पोर्ट यासारख्या गोष्टी मॅनेज करतात, स्पेशल तः जेव्हा संगणक बूट होत असतो.

- हार्ड डिस्क, घड्याळ इ. साठी सेटिंग्जचा संग्रह मॅनेज करणे.

### CMOS सेटअप

BIOS प्रथम गोष्ट करेल ती म्हणजे कॉम्प्लिमेंटरी मेटल ऑक्सिड सेमीकंडक्टर (CMOS) चिपवर असलेल्या RAM च्या लहान (64 बाइट्स) प्रमाणात स्टोर इनफार्मेशन तपासणे. CMOS सेटअप तुमच्या सिस्टमला तपशीलवार इनफार्मेशन पुरवतो आणि तुमची सिस्टीम बदलल्याप्रमाणे बदलली जाऊ शकते. BIOS ही इनफार्मेशन आवश्यकतेनुसार त्याचे डीफॉल्ट प्रोग्रामिंग सुधारण्यासाठी किंवा पूरक करण्यासाठी वापरते.

### BIOS कॉन्फिगर करत आहे

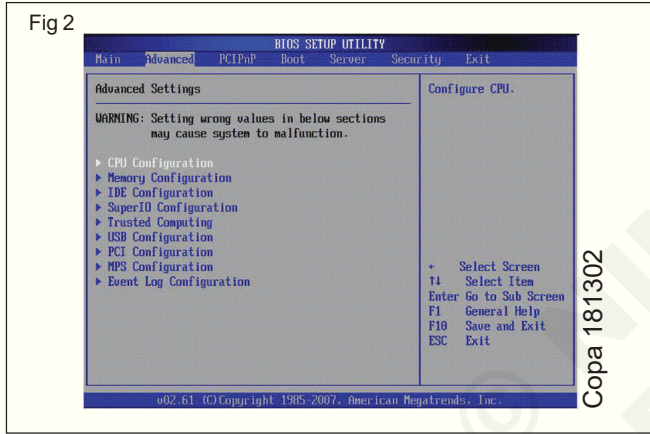
BIOS कस्टम सेटिंग्जसाठी CMOS सेटअप तपासते. CMOS सेटिंग्ज बदलण्यासाठी आम्हाला CMOS सेटअप प्रविष्ट करणे आवश्यक आहे. CMOS सेटअपमध्ये एक्सेस करण्यासाठी, स्टार्टिंग स्टार्टअप क्रमादरम्यान एक स्पेसिफिक की किंवा कीचे संयोजन दाबले जाणे आवश्यक आहे. सेटअप एंटर करण्यासाठी बहुतेक सिस्टम "Esc," "Del," "F1," "F2," "Ctrl-Esc" किंवा "Ctrl-Alt-Esc" वापरतात. डिस्प्लेच्या तळाशी सहसा टेक्स्ट ची एक ओळ असते जी "सेटअप प्रविष्ट करण्यासाठी \_\_\_ दाबा" असे सांगते.

BIOS सेटअप अनेक पर्यायांसह टेक्स्ट स्क्रीनचा संच दाखवतो. यापैकी काही स्टॅण्डर्ड आहेत, तर इतर BIOS निर्मात्यानुसार बदलतात. सामान्य पर्यायांमध्ये हे समाविष्ट आहे:

- सिस्टम वेळ/date - सिस्टम वेळ आणि date सेट करा
- बूट क्रम - BIOS ऑपरेटिंग सिस्टम लोड करण्याचा प्रयत्न करेल असा क्रम
- प्लग आणि प्ले - कनेक्ट केलेल्या उपकरणांचे स्वयं-सर्च प्यासाठी स्टॅण्डर्ड; जर तुमचा संगणक आणि ऑपरेटिंग सिस्टीम दोन्ही त्यास सपोर्ट देत असतील तर "होय" वर सेट केले पाहिजे
- माउस/कीबोर्ड - "नम लॉक सक्षम करा," "कीबोर्ड सक्षम करा," "माऊस ऑटो-डिटेक्ट करा"...
- ड्राइव्ह कॉन्फिगरेशन - हार्ड ड्राइव्ह, CD-ROM आणि फ्लॉपी ड्राइव्ह कॉन्फिगर करा

- मेमरी - BIOS ला स्पेसिफिक मेमरी पत्यावर सावली करण्यासाठी निर्देशित करा
- सुरक्षा - संगणकावर एक्सेस करण्यासाठी पासवर्ड सेट करा
- पॉवर मॅनेजमेंट - पॉवर मॅनेजमेंट वापरायचे की नाही ते निवडा, तसेच "स्टँडबाय" आणि "सस्पेंड" साठी वेळ सेट करा
- बाहेर पडा - तुमचे बदल सेव्ह करा, तुमचे बदल टाकून द्या किंवा डीफॉल्ट सेटिंग्ज रिस्टोर करा

संगणकाच्या सेटिंग्जमध्ये केलेले कोणतेही बदल सेव्ह करण्यासाठी BIOS CMOS तंत्रज्ञान वापरते. या तंत्रज्ञानाद्वारे, एक लहान लिथियम किंवा Ni-Cad बॅटरी वर्षानुवर्षे डेटा ठेवण्यासाठी पुरेशी उर्जा देऊ शकते. प्रमुख BIOS उत्पादकांमध्ये American Megatrends Inc. (AMI), Phoenix Technologies, Winbond इत्यादींचा समावेश आहे. एक स्पेसिफिक BIOS स्क्रीनशॉट आकृती मध्ये दर्शविला आहे. खाली दिले आहे. (पिव्चर 2 पहा)



### विंडोज ऑपरेटिंग सिस्टम इंस्टाल करणे

हार्ड डिस्कचे पार्टिशन करणे आवश्यक आहे (अनिवार्य नसले तरी) आणि आपण त्यावर डेटा संचयित करण्यापूर्वी त्याचे फॉर्मॅटिंग करणे आवश्यक आहे.

### पार्टिशन करणे

पार्टिशन, ज्याला कधीकधी व्हॉल्यूम देखील म्हणतात, हार्ड डिस्कवरील एक एरिया आहे जे फाइल सिस्टमसह फॉर्मॅटिंग केले जाऊ शकते आणि कॅरक्टर मालाच्या अक्षराने ओळखले जाऊ शकते. उदाहरणार्थ, बहुतेक Windows संगणकांवर ड्राइव्ह C हे पार्टिशन आहे. तुम्ही तयार केलेले पहिले तीन पार्टिशन प्रायमरी पार्टिशन आहेत. हे ऑपरेटिंग सिस्टम सुरू करण्यासाठी वापरले जाऊ शकते. तुम्ही तीनपेक्षा जास्त पार्टिशन तयार केल्यास, चौथे पार्टिशन एक्सटेंडेड पार्टिशन म्हणून तयार केले जाते.

एक्सटेंडेड पार्टिशन एक कंटेनर आहे जे एक किंवा अधिक लॉजिकल ड्राइव्ह ठेवू शकते. लॉजिकल ड्राइव्हस प्रायमरी विभाजनांप्रमाणे कार्य करतात त्याशिवाय ते ऑपरेटिंग सिस्टम सुरू करण्यासाठी वापरले जाऊ शकत नाहीत.

हार्ड डिस्कच्या आकाराएवढे सिंगल पार्टिशन म्हणून अनेक संगणकांचे पार्टिशन केले जाते. हार्ड डिस्कचे अनेक लहान विभाजनांमध्ये पार्टिशन

करणे आवश्यक नाही, परंतु ते तुमच्या हार्ड डिस्कवरील डेटा व्यवस्थित करण्यासाठी सबयुक्त ठरू शकते.

एकापेक्षा जास्त पार्टिशन तयार करण्याचे खालील फायदे आहेत:

- ऑपरेटिंग सिस्टीम (OS) आणि प्रोग्राम फाइल्स यूजर च्या फाइल्समधून वेगळे करणे. मेमरी स्वॉपिंग/पेजिंग.
- वारंवार वापरले जाणारे प्रोग्राम आणि डेटा एकमेकांजवळ ठेवणे.
- मल्टी-बूट सेटअपचा वापर, जे वापरकर्त्यांना एकाच संगणकावर एकापेक्षा जास्त ऑपरेटिंग सिस्टम ठेवण्याची परवानगी देतात. उदाहरणार्थ, समान HDD च्या वेगवेगळ्या विभाजनांवर लिनक्स आणि मायक्रोसॉफ्ट विंडोज किंवा इतर ऑपरेटिंग सिस्टम इंस्टॉल करू शकतात आणि पॉवर-अपवर कोणत्याही ऑपरेटिंग सिस्टममध्ये बूट करण्याचा पर्याय असू शकतो.
- करप्टेड फाइल सिस्टम किंवा ऑपरेटिंग सिस्टम इंस्टॉलेशन पुनर्प्राप्त करणे सोपे करण्यासाठी फाइल्सचे प्रोटेक्शन करणे किंवा वेगळे करणे. एक पार्टिशन करप्टेड असल्यास, इतर फाइल प्रणालींवर रिझल्ट होणार नाही.
- जेथे लहान फाइल सिस्टीम अधिक कार्यक्षम आहेत अशा सिस्टीमवर संपूर्ण संगणक कार्यप्रदर्शन वाढवणे.
- सबलब्ध पूर्ण आकारापेक्षा कमी प्रमाणात पार्टिशन केल्याने चेकडिस्क सारख्या निदान साधनांचा वेळ कमी होऊ शकतो.

### फॉर्मॅटिंग

डिस्क फॉर्मॅटिंग ही स्टार्टिंग वापरासाठी हार्ड डिस्क ड्राइव्ह, सॉलिड-स्टेट ड्राइव्ह किंवा USB फ्लॅश ड्राइव्ह सारखे डेटा स्टोरेज डिव्हाइस तयार करण्याची प्रक्रिया आहे. हे व्हॉल्यूमवर फाइल सिस्टम तयार करण्याची क्रिया आहे, ज्यामुळे ऑपरेटिंग सिस्टम त्या व्हॉल्यूमवर डेटा स्टोर आणि पुनर्प्राप्त करू शकते.

डिस्कचे फॉर्मॅटिंग दोन रेन्ज मध्ये आहे:

- 1 लो-लेव्हल फॉर्मॅटिंग (म्हणजे, हार्डवेअरच्या सर्वात जवळ) डिस्कच्या पृष्ठभागांना रेकॉर्डिंग ब्लॉकची सुरुवात दर्शविणाऱ्या मार्करसह चिन्हांकित करते. हे डिस्क कंट्रोलरद्वारे डेटा वाचण्यासाठी किंवा लिहिण्यासाठी भविष्यातील वापरासाठी केलेल्या ब्लॉक चेकबद्दल इनफार्मेशन देखील प्रदान करते. हे डिस्कचा कायमस्वरूपी पाया बनवण्याच्या उद्देशाने आहे आणि बहुतेकदा कारखान्यात पूर्ण केले जाते. हार्ड डिस्कवर डेटा स्टोर करण्यापूर्वी त्याचे पार्टिशन आणि फॉर्मॅटिंग करणे आवश्यक आहे
- 2 हाय-लेव्हल फॉर्मॅटिंग डिस्क पार्टिशन किंवा लॉजिकल व्हॉल्यूममध्ये फाइल सिस्टम फॉर्मॅट तयार करते. या फॉर्मॅटिंगमध्ये OS द्वारे लॉजिकल ड्राइव्ह किंवा पार्टिशन ची कंटेंट ओळखण्यासाठी वापरलेल्या डेटा स्ट्रक्चर्सचा समावेश होतो. हे ऑपरेटिंग सिस्टम इंस्टॉलेशन दरम्यान किंवा नवीन डिस्क जोडताना उद्भवू शकते.

### विंडोज ऑपरेटिंग सिस्टम इंस्टाल करणे

विंडोज इंस्टॉलेशन प्रक्रियेचे तीन बेसिक टाइप खालीलप्रमाणे आहेत:

- अगदी नवीन डिस्क किंवा संगणक सिस्टिम वर इंस्टाल करा

- डिस्क इरेज करा, त्याचे फॉर्मॅटिंग करा आणि इंस्टाल करा.
- ड्युअल-बूटिंगसाठी नवीन डिरेक्ट्री इंस्टाल करा

पहिल्या दोन पद्धतींसाठी, संगणक DVD किंवा इतर कोणत्याही काढता येण्याजोग्या ड्राइव्हवरून बूट होऊ शकतो याची खात्री करणे आवश्यक

आहे. हे करण्यासाठी BIOS मध्ये ड्राइव्ह बूट ऑर्डर बदलणे आवश्यक आहे. नवीनतम विंडोज डीव्हीडी बूट करण्यायोग्य आहेत आणि सेटअप प्रोग्राम चालवतात. ट्रेड प्रॅक्टिकलप्रमाणेच त्यानंतरच्या स्क्रीनवर दर्शविल्यानुसार स्टेप बाय स्टेप प्रक्रियेचे अनुसरण करून केले जाऊ शकते.

© NIMI  
NOT TO BE REPUBLISHED

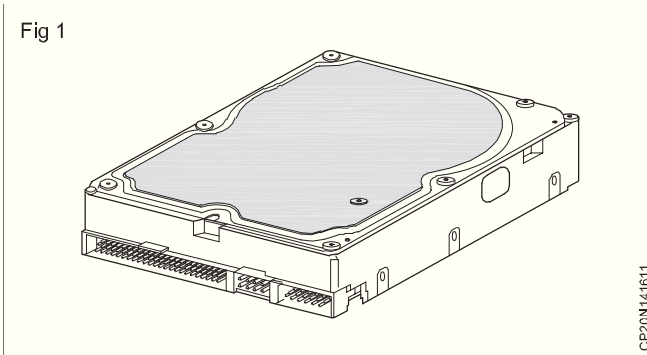
## COPA - संगणक बेसिक आणि सॉफ्टवेअर इन्स्टॉलेशन

## हार्ड डिस्कचे फॉर्मॅटिंग करा आणि पार्टीशन तयार करा (Format hard disk and create partition)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- BIOS आणि CMOS च्या कार्याचे डिस्क्रिप्शन करा
- BIOS सेटिंग्ज पाहण्याच्या आणि बदलण्याच्या मेथड्स चे डिस्क्रिप्शन करा
- पार्टीशन आणि फॉर्मॅटिंग चा अर्थ डिस्क्रिप्शन करा
- विंडोज ऑपरेटिंग सिस्टम इन्स्टॉल करण्याच्या प्रक्रियेचे डिस्क्रिप्शन करा. हार्ड ड्राइव्ह फॉर्मॅटिंग करा

## हार्ड ड्राइव्ह फॉर्मॅटिंग करा



हार्ड ड्राइव्ह फॉर्मॅट करण्याची अनेक कारणे आहेत, जसे की विंडोज फ्रेश इन्स्टॉल करण्यासाठी, व्हायरस किंवा मालवेअरपासून ओपन होण्यासाठी किंवा फक्त विकताना पीसी रिफ्रेश किंवा क्लीन अप केल्यामुळे.

ती फक्त हार्ड ड्राइव्ह आहे की नाही आणि अतिरिक्त पीसी आहे की नाही यावर अवलंबून प्रक्रिया भिन्न असू शकते.

हे फॉर्मॅट केले जाऊ शकत नाही, हार्ड ड्राइव्ह ज्यावर विंडोज चालू आहे. हे करण्यासाठी, विंडोज इन्स्टॉलेशन डिस्क, यूएसबी फ्लॅश ड्राइव्ह किंवा इतर बूट करण्यायोग्य डिस्कवरून पीसी बूट करणे आवश्यक आहे.

## डिस्क फॉर्मॅट करायची?

फॉर्मॅटिंग ही हार्ड ड्राइव्हवरील सर्व डेटा हटविण्याची प्रक्रिया आहे, परंतु 'क्लिक फॉर्मॅट'पासून सावध रहा जे सर्व डेटा जागेवर ठेवते आणि ड्राइव्ह रिक्त असल्याचे दिसते. जर नवीन हार्ड ड्राइव्ह असेल किंवा विंडोज पुन्हा इंस्टाल करण्याची आवश्यकता असेल तर द्रुत फॉर्मॅट ठीक आहे, परंतु डिस्कची विल्हेवात लावत असल्यास किंवा दुसऱ्याला देत असल्यास नाही.

चेतावणीचा एक शब्द: ड्राइव्हचे कोणतेही फोटो, व्हिडिओ, संगीत आणि इतर डॉक्युमेंट्स चे फॉर्मॅटिंग करण्यापूर्वी त्याचा यशस्वीपणे बॅकअप घेतल्याची खात्री करा. हटवलेल्या फायली काही परिस्थितींमध्ये पुनर्प्राप्त केल्या जाऊ शकतात, तरीही सबचारापेक्षा प्रतिबंध नेहमीच चांगला असतो.

## हार्ड ड्राइव्ह पार्टीशन फॉर्मॅटिंग करा

स्टार्ट करण्यापूर्वी विभाजनांबद्दल समजून घेणे महत्त्वाचे आहे. हार्ड ड्राइव्हला लहान विभागांमध्ये विपार्ट ले जाऊ शकते, ज्याला पार्टीशन म्हणतात. इतरांना अस्पर्श ठेवताना एक पार्टीशन फॉर्मॅटिंग करणे शक्य आहे.

संपूर्ण हार्ड ड्राइव्हचे फॉर्मॅटिंग करणे आणि संपूर्ण क्षमता एका ब्लॉकमध्ये वापरणे आवश्यक असल्यास, पार्टीशन इनफार्मेशन डिलेट करा.

BIOS वरून हार्ड ड्राइव्ह फॉर्मॅट करायची?

बरेच लोक विचारतात की BIOS वरून हार्ड डिस्कचे फॉर्मॅटिंग कसे करावे. लहान उत्तर नाही आहे.

डिस्कचे फॉर्मॅटिंग करणे आवश्यक असल्यास आणि आपण ते Windows मधून करू शकत नसल्यास, बूट करण्यायोग्य CD, DVD किंवा USB फ्लॅश ड्राइव्ह तयार करा आणि विनाव्हॅल्यु तृतीय-पक्ष फॉर्मॅटिंग साधन चालवा.

एक पर्याय आहे Darik's Boot and Nuke (DBAN), जो वैयक्तिक वापरासाठी विनाव्हॅल्यु आहे. हा प्रोग्राम तुमची हार्ड डिस्क पूर्णपणे मिटवेल आणि फॉर्मॅट करेल, नवीन OS च्या स्वच्छ इंस्टॉलेशनला अनुमती देईल, परंतु प्रक्रिया पूर्ववत केली जाऊ शकत नाही.

DBAN फक्त बूट करण्यायोग्य CD/DVD-R तयार करण्यास सक्षम आहे,



परंतु जर रिकामी डिस्क किंवा बर्नर नसेल, तर वेगळ्या थर्डपार्टी प्रोग्रामच्या स्वरूपात एक वर्कअराउंड सबलब्ध आहे.

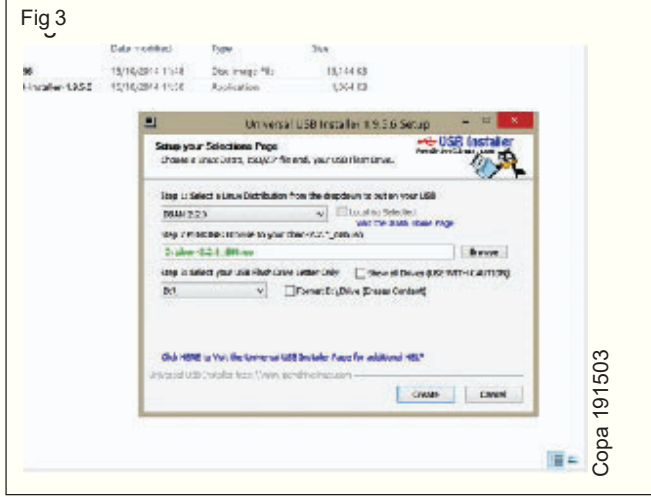
युनिव्हर्सल यूएसबी इंस्टॉलर बूट करण्यायोग्य यूएसबीवरून चालविण्यासाठी डाउनलोड केलेली DBAN ISO प्रतिमा जलद आणि सहजपणे रूपांतरित करेल. फक्त रिक्त USB फ्लॅश ड्राइव्ह इन्सर्ट, युनिव्हर्सल USB इंस्टॉलर सेटअप प्रोग्राम चालवा आणि सूचनांचे अनुसरण करा.

ते स्क्रील करून लिनक्स डिस्ट्रिब्युशनला यूएसबी (या प्रकरणात, डीबीएन ची नवीनतम आवृत्ती) वर इंस्टाल करायचे आहे, त्यानंतर संगणकावर त्याचे पोजिशन आणि USB ड्राइव्हचे पत्र ते इंस्टाल करण्यास सांगेल.

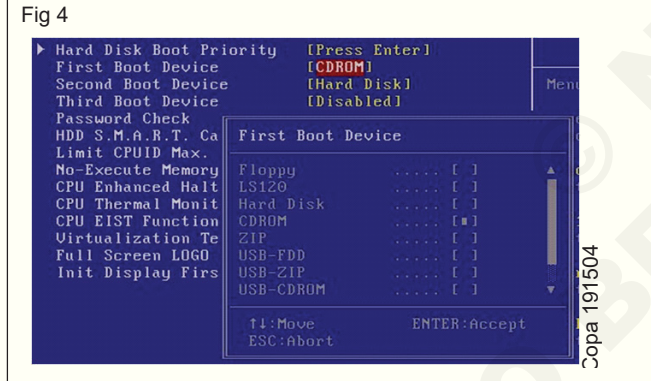
ती इनफार्मेशन पूर्ण झाल्यावर, तयार करा क्लिक करा.

नेहमीच्या बूट इन्फार्मेशन पेक्षा या USB ड्राइववरून बूट करण्यासाठी (बहुतेक प्रकरणांमध्ये, हा हार्ड ड्राइव्ह असेल), BIOS मध्ये काही सेटिंग्ज बदलणे आवश्यक आहे.

BIOS मध्ये, बूट ऑर्डर सेटिंग्जवर नेव्हिगेट करा आणि प्रायमरी बूट डिवाइस USB ड्राइव्हवर बदला (ही निवड करण्यासाठी त्यास प्लग इन करण्याची आवश्यकता नाही). सेटिंग्ज सेव्ह केल्यानंतर आणि BIOS मधून बाहेर पडल्यानंतर, बूट करण्यायोग्य यूएसबी इन्सर्ट, संगणक रीस्टार्ट करा.



Copa 191503



Copa 191504

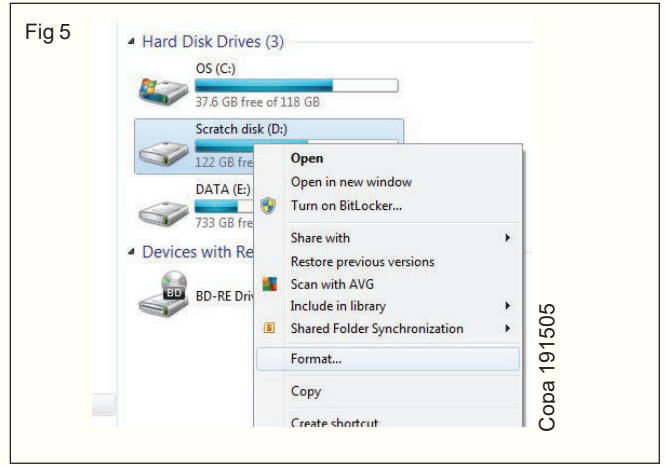
PC ने DBAN सॉफ्टवेअर ऑटोमॅटिक बूट केले पाहिजे, जे हार्ड ड्राइव्ह मिटवण्याच्या प्रक्रियेत पाठ दर्शन करेल, डेटा-वाइपिंगच्या विविध स्तरांसाठी पर्यायांसह.

ते यूएसबीला दुसऱ्या ड्राइव्हसारखे मानेल जेणेकरून अनवधानाने पुसले जाऊ नये म्हणून, डीबीएन मध्ये बूट केल्यानंतर ते काढून टाका.

क्लिक फॉर्मॅटिंग हार्ड ड्राइव्ह?

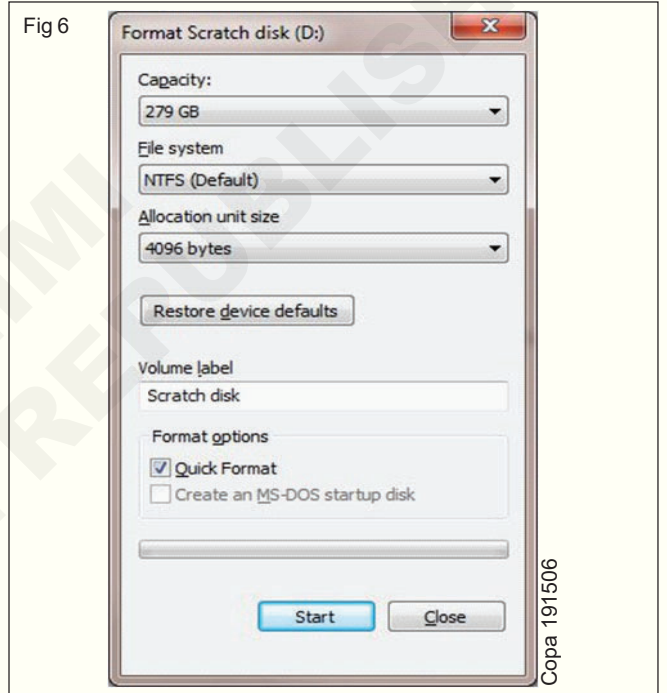
होय, परंतु डेटा कायमचा मिटवायचा असल्यास ही मेथड्स वापरू नका. क्लिक फॉर्मॅटिंग डेटा हटवत नाही परंतु त्याऐवजी फायलीकडे फक्त पॉइंटर मिटवते.

Windows Vista, 7, 8 आणि 10 मध्ये अंगभूत डिस्क मॅनेजमेंट साधन आहे (खाली पहा), परंतु हार्ड ड्राइव्हचे फॉर्मॅटिंग करण्याचा सर्वात जलद पाथ म्हणजे स्टार्ट बटण क्लिक करणे, नंतर संगणक आणि पुसण्यासाठी हार्ड ड्राइव्हवर उजवे-क्लिक करणे. स्पष्ट कारणास्तव ज्या ड्राइव्हवर Windows इंस्टाल केले आहे ते फॉर्मॅटिंग केले जाऊ शकत नाही.



Copa 191505

डीफॉल्टनुसार क्लिक फॉर्मॅट तपासले जाते, आणि फाइल सिस्टम आणि वाटप युनिट आकार तसेच व्हॉल्यूम लेबल (ड्राइव्हचे नाव) बदलणे निवडा. सामान्यतः, सर्व सेटिंग्ज अपरिवर्तित सोडा आणि स्टार्ट बटणावर क्लिक करा. एका मिनिटात हार्ड ड्राइव्ह फॉर्मॅट होईल.



Copa 191506

Windows Vista, 7, 8 किंवा 10 साठी NTFS आधीपासून निवडलेली नसल्यास फाइल सिस्टम म्हणून निवडा आणि वाटप युनिट आकार 'डीफॉल्ट वाटप आकार' वर सेट केला असल्याचे सुनिश्चित करा.

डिस्क मॅनेजमेंट साधन वापरणे

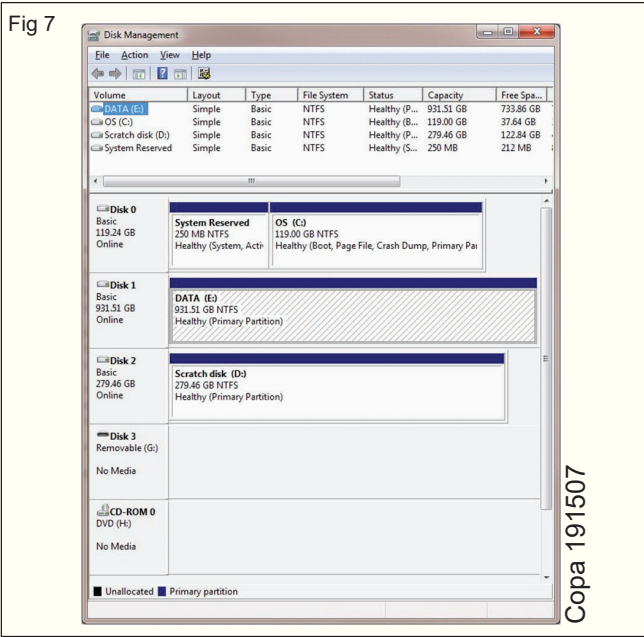
Vista, 7, 8 किंवा 10 मधील सर्च बॉक्समध्ये diskmgmt.msc किंवा डिस्क मॅनेजमेंट टाइप करा आणि नंतर त्याच नावाने वरील मेनूमध्ये दिसणाऱ्या फक्त निकालावर क्लिक करा.

डिस्क मॅनेजमेंट लॉच करण्याचा हा सर्वात सोपा पाथ आहे, परंतु 'डिस्क' सर्च ल्यास आणि 'हार्ड डिस्क पार्टिशन तयार करा आणि फॉर्मॅटिंग करा' निवडल्यास कंट्रोल पॅनेलमध्ये देखील शोधा.

डिस्क मॅनेजमेंट हे स्टँडअलोन पार्टिशन मॅनेजमेंट साधन इतके शक्तिशाली नाही, परंतु तरीही ते डेटा फॉर्मॅटिंग करण्यास सक्षम आहे.



Fig 7

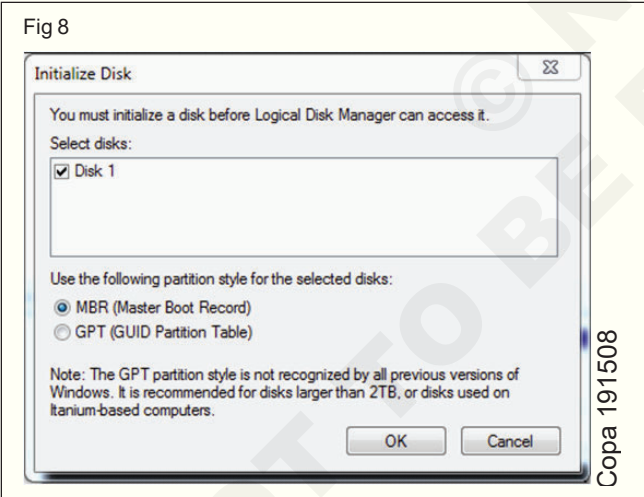


Copa 191507

जर पीसीमध्ये नवीन (अतिरिक्त) हार्ड ड्राइव्ह इंस्टाल करायची असेल, तर ती विंडोज एक्सप्लोररमध्ये का दिसत नाही याबद्दल आश्चर्य वाटेल. याचे कारण असे आहे की ते स्टार्ट करणे आणि फॉर्मॅटिंग करणे आवश्यक आहे - जे डिस्क मॅनेजमेंट मध्ये केले जाऊ शकते.

जेव्हा टूल लोड होईल, तेव्हा ते संगणकाच्या सर्व ड्राइव्हचे विश्लेषण करेल आणि त्याला सापडलेल्या नवीन डिस्कला स्टार्ट करण्यास सूचित करेल.

Fig 8



Copa 191508

डिस्क 2TB पेक्षा मोठी असल्यास, GPT (GUID पार्टीशन टेबल) निवडा. हे सेटिंग चार पेक्षा जास्त पार्टीशन देखील तयार करू देते.

प्रॉम्प्ट दिसत नसल्यास, ड्राइव्हच्या लिस्ट मध्ये पहा आणि 'इनिशियलाइज्ड नाही' असे एक पहा. त्यावर उजवे-क्लिक करा आणि डिस्क स्टार्ट करा निवडा.

एकदा ते पूर्ण झाल्यावर, हॅच केलेल्या अनअलोकेटेड जागेवर उजवे-क्लिक करा आणि नवीन साधा खंड निवडा...

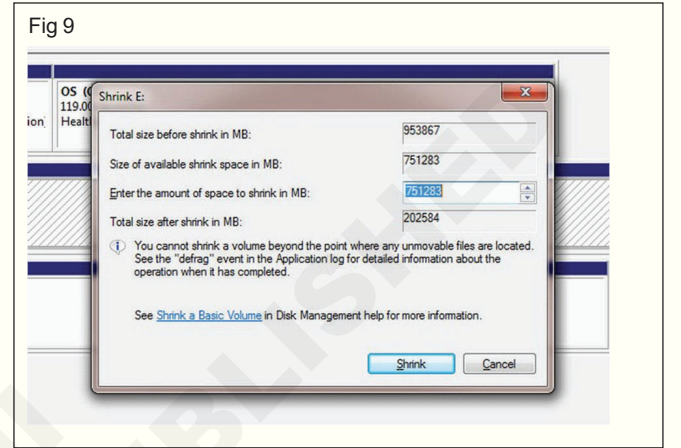
पार्टीशन किती मोठे असावे (MB - 1024MB = 1GB मध्ये) आणि कोणते ड्राइव्ह लेटर नियुक्त करायचे ते निवडून, सूचनांचे अनुसरण करा (एखादे निवडले जाईल, परंतु इच्छित असल्यास ते बदलण्याची निवड करू शकता).

ड्राइव्हच्या एकूण क्षमतेपेक्षा लहान असलेल्या पार्टीशन साठी आकार निवडल्यास, 1TB ड्राइव्हवर 500B म्हणा, ड्राइव्हवर काही न वाटप केलेली जागा असेल जी नुकतीच पूर्ण झालेल्या प्रक्रियेची पुनरावृत्ती करून फॉर्मॅट करू शकते.

पार्टीशन साइज बदला

डिस्क मॅनेजमेंटचा वापर पार्टीशन विस्तृत किंवा कॉम्प्रेस करण्यासाठी केला जाऊ शकतो. फक्त एकावर उजवे-क्लिक करा आणि दिसत असलेल्या मेनूमधून योग्य पर्याय निवडा. कॉम्प्रेस होत असल्यास, त्यात असलेली जागा.

Fig 9



हे थोडे गोंधळात टाकणारे आहे कारण नंबर GB ऐवजी MB मध्ये प्रदर्शित केली जाते, परंतु ते कॉम्प्रेस होण्यासाठी जागेचे प्रमाण अडजस्ट केले जाऊ शकते आणि 'कॉम्प्रेस झाल्यानंतर एकूण आकार' अपडेट केला जाईल, तसेच फायलीच्या पॉइंट च्या पलीकडे पार्टीशन कमी करू शकत नाही. स्थित आहेत - ते प्रथम ड्राइव्ह डीफ्रॅगमेंट करून जागा मोकळी करण्यात सक्षम होऊ शकते.

याउलट ड्राइव्हवर न वापरलेली जागा असेल तरच ते पार्टीशन विस्तृत करू शकते. नसल्यास, पर्याय धूसर होईल.

विंडोज रीसेट करणे

जर पीसीला समस्या येत असतील आणि यूजर ला हार्ड ड्राइव्ह पुसून वैयक्तिक फाइल्स गमावण्याची इच्छा नसेल, तर तो पीसी रिफ्रेश किंवा रीसेट करू इच्छित असेल जे विंडोज 8 आणि 10 मध्ये केले जाऊ शकते.

जर यूजर तुमच्या वैयक्तिक फाइल्स आणि सेटिंग्ज ठेवू इच्छित असेल, परंतु नवीन विंडोज इन्स्टॉल करू इच्छित असेल, तर त्याला सर्व पीसी रिफ्रेश करायचे आहे.

लक्षात ठेवा की रिफ्रेश मशीनवर इंस्टाल केलेले सर्व प्रोग्राम्स आणि ॲप्स काढून टाकेल, परंतु Windows-डिफॉल्ट प्रोग्राम्स कायम ठेवतील.

रीसेट केल्याने विंडोज पुन्हा इंस्टाल होते आणि सर्व फाइल्स, सेटिंग्ज आणि प्रोग्राम हटवल्या जातात, जर पूर्वी सर्व फायलींचा बॅकअप घेतला असेल आणि यूजर च्या वैयक्तिक फायली हस्तांतरित करण्यास हरकत नसेल तर असे करण्यास सुचवले जाते.

Windows 10 चा दृष्टीकोन थोडा वेगळा आहे आणि तो Windows 8 वरून येणाऱ्यांना गोंधळात टाकू शकतो. मायक्रोसॉफ्टने रिफ्रेश पर्याय

काढून टाकला आहे आणि त्याऐवजी रिफ्रेश आणि रीसेट पर्याय एकाच सेटिंगमध्ये एकत्र केले आहेत.

पर्याय सर्व प्यासाठी, स्टार्ट मेनू उघडा, सेटिंग्ज > अपडेट आणि सुरक्षा > रिकव्हरी > गेट स्टार्ट (हा पीसी रीसेट करा पर्याय इंटरनल ) वर क्लिक करा.

पर्याय निवडल्यावर, ते तीन स्वतंत्र पर्यायांसह सादर करेल:

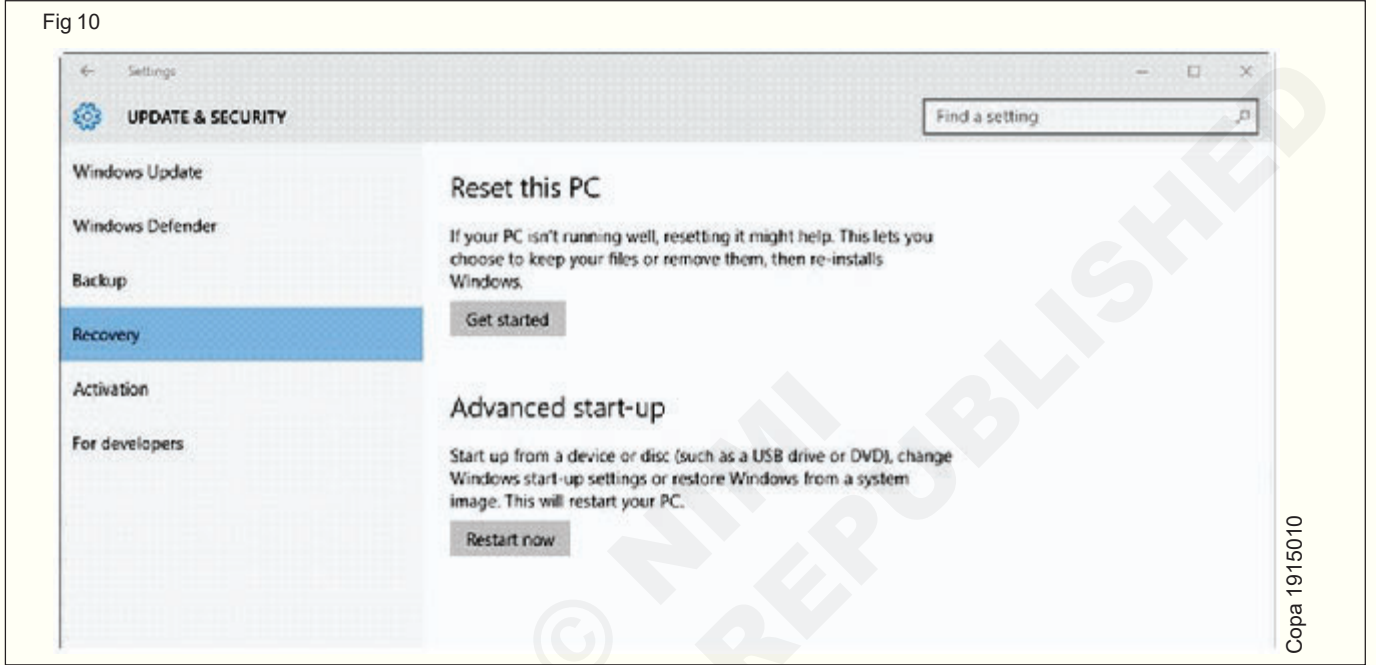
- माझ्या फायली ठेवा
- सर्वकाही काढून टाका
- फॅक्टरी सेटिंग्ज रिस्टोअर करा (सर्व PC वर सबलब्ध नाही)

'कीप माय फाइल्स' वैयक्तिक फाइल्स सेव्ह करते परंतु युजर सेटिंग्ज, ड्रायव्हर्स आणि प्रोग्राम हटवते, तसेच विंडोज 10 पुन्हा इंस्टाल करते.

'सर्व काही काढा' वैयक्तिक फाइल्सपासून देखील ओपन होते आणि 'फॅक्टरी सेटिंग्ज रिस्टोअर करा' हे 'सर्व काही काढा' प्रमाणेच क्रिया करते, परंतु मशीनसह आलेल्या विंडोजच्या आवृत्तीवर पीसी रीसेट करते.

हा पर्याय फक्त त्यांच्यासाठीच सबलब्ध असेल ज्यांनी प्री-बिल्ट पीसी किंवा लॅपटॉप विकत घेतला आहे जे आधीपासूनच प्री-इंस्टॉल केलेले Windows सह आले आहेत.

Fig 10



## OS इंस्टॉलेशन दरम्यान सामान्य हार्डवेअर आणि सॉफ्टवेअर समस्या ओळखा आणि दुरुस्त करा (Identify and rectify common hardware and software issues during OS installation)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- PC ट्रबलशूटींग करण्यात गुंतलेली बेसिक स्टेप सांगा
- समस्या सोडवण्यासाठी बेसिक दृष्टिकोन स्पष्ट करा
- सदोष संगणकातील संभाव्य दोष आणि लक्षणांची यादी करा
- तक्रारीच्या कारणांचे विश्लेषण करा
- दोष दूर करण्यासाठी सर्वात लहान पाथ सांगा.
- सदोष संगणकातील संभाव्य दोष आणि लक्षणांची यादी करा
- प्रॉब्लेम ट्री आणि टीएससीच्या मदतीने "विंडोज सुरू झाल्यावर, सिस्टम पृष्ठभागाची टेस्टिंग घेते आणि सेफमोडवर जाते" या तक्रारीच्या कारणांचे विश्लेषण करा.

### बेसिक ट्रबलशूटींग

पीसीवर समस्यांचे निवारण करताना एक अडचण अशी आहे की बहुतेक प्रकरणांमध्ये ते दिसत नाहीत. पीसी वारंवार हँग होण्यामागील कारण सहा किंवा अधिक चांगल्या प्रकारे परिभाषित क्षेत्रांपैकी एक किंवा डझनभर अज्ञात समस्या असू शकतात. समस्या सॉफ्टवेअर किंवा हार्डवेअरमुळे असू शकते. अनेक वर्षांचा अनुभव आणि प्रशिक्षण असतानाही, PC तंत्रज्ञ समस्यानिवारण प्रक्रियांसह बाहेर पडतात ज्यामुळे वास्तविक समस्या सुटत नाहीत. उदाहरणार्थ, अनेक i/o कार्ड कनेक्ट केलेल्या विंडोज ऑपरेटिंग सिस्टमवर चालणारा वैयक्तिक संगणक स्क्रीन, माउस आणि कीबोर्ड प्रीज करू शकतो आणि प्रतिसाद देण्यापूर्वी 3 मिनिटे लागू शकतो. मेमरी मॉड्यूल बदलणे, नवीन पॅरलल पोर्ट आणि NIC ड्रायव्हर्स इंस्टाल करणे यासारख्या सर्व पर्यायांसह प्रयत्न केल्यानंतर, तंत्रज्ञ शेवटी सिस्टम लॉग तपासतो की एक महत्त्वपूर्ण ऑपरेटिंग सिस्टम लायब्ररी करप्टेड झाली आहे आणि ती पुन्हा इंस्टाल करणे आवश्यक आहे. येथे नमूद करण्याजोगा मुद्दा असा आहे की जर तंत्रज्ञांनी समस्यानिवारणासाठी मेथड्स शीर दृष्टीकोन वापरला, तर कदाचित समस्या खूप लवकर सोडवली गेली असती. एक चांगली समस्यानिवारण योजना राखून ठेवल्याने निश्चितपणे आम्हाला अधिक मेथड्स शीर आणि वैज्ञानिक मेथड्स ने समस्येकडे जाण्याची संधी मिळते. समस्यानिवारण योजना ही एक लेखी चेक लिस्ट आहे जी आम्ही कोणत्याही समस्येसाठी वापरतो. कोणत्याही समस्यानिवारण योजनेमध्ये समाविष्ट केलेले एलिमेंट खालीलप्रमाणे आहेत

- देखभाल रेकॉर्ड
- संभाव्य कारणांची ओळख
- संभाव्य उपायांची ओळख
- उपायांचा एप्लिकेशन आणि टेस्टिंग
- पाठपुरावा

देखरेख रेकॉर्डमध्ये, PC मध्ये इंस्टाल केलेले हार्डवेअर इंस्टाल केल्यावर, सर्व प्रतिबंधात्मक देखभाल क्रियाकलाप, सर्व सॉफ्टवेअर अपडेट किंवा जोडणी आणि सर्व हार्डवेअर इंस्टॉलेशनस आणि अपग्रेड्स रेकॉर्ड करा. पुढे येणाऱ्या कोणत्याही समस्या आणि त्या सोडवण्यासाठी तुम्ही केलेल्या कृतींची नोंद केली पाहिजे. जेव्हा पीसीच्या समस्यानिवारणाचा प्रश्न येतो, तेव्हा देखभाल रेकॉर्डसह एखादी समस्या पिन करू शकते आणि त्यावर उपाय शोधू शकतो. अशा रेकॉर्डमधील पहिली नोंद पीसीची प्रोफाइल असावी, ज्यामध्ये त्याचे कॉन्फिगरेशन, ऑपरेटिंग सिस्टम आणि प्रत्येक एलिमेंट इंस्टाल केल्याची date समाविष्ट असते.

अशा तक्त्यांवरून सिटम कॉन्फिगरेशनमध्ये समाविष्ट केलेल्या इनफार्मेशन च्या प्रकाराची कल्पना येते, जसे की खाली समराइज केले आहे:

- प्रोसेसरचा मेक, मॉडेल आणि वेग.
- सिस्टम मेमरी (RAM) चे प्रमाण आणि मेमरी मॉड्यूल टाइप, आकार आणि मेमरीचे कॉन्फिगरेशन
- हार्ड डिस्कचा आकार आणि इंटरफेसचा टाइप
- कॉम्पॅक्ट डिस्क (CD) किंवा डिजिटल व्हर्सटाइल डिस्क (DVD) ड्राइव्हचे बनवा, मॉडेल आणि गती
- व्हिडिओ किंवा ग्राफिक्स अॅडॉप्टरचा मेमरी आकार
- मॉडेमचे बनवा, मॉडेल, टाइप आणि गती
- ऑपरेटिंग सिस्टमची आवृत्ती क्रमांक
- PC वर इंस्टाल सॉफ्टवेअर एप्लिकेशन ची यादी
- पीसीला जोडलेल्या पेरीफेरल उपकरणांची यादी, ते कोणत्या पोर्टला जोडलेले आहेत हे दर्शविते

प्रत्येक वेळी PC वर देखभालीचे काम चालू असताना देखभाल रेकॉर्ड अपडेट केले जावे. नवीन किंवा बदली हार्डवेअर इंस्टाल केल्यावर,

क्रियाकलाप रेकॉर्ड करा आणि सिस्टम कॉन्फिगरेशन अपडेट करा. ऍक्टिव्हिटी नोंदींचा समावेश असावा

- ऍक्टिव्हिटी ची date किंवा सिस्टममध्ये केलेले बदल
- PC मध्ये काढलेल्या किंवा जोडलेल्या कोणत्याही हार्डवेअरचा मेक, मॉडेल आणि अनुक्रमांक - PC मध्ये जोडलेल्या कोणत्याही सॉफ्टवेअरचे नाव, आवृत्ती आणि प्रकाशक
- बेसिक इनपुट/आउटपुट सिस्टम (BIOS) किंवा नवीन डिव्हाइस किंवा सॉफ्टवेअरसाठी इतर कॉन्फिगरेशनमध्ये केलेल्या कोणत्याही कॉन्फिगरेशन बदलांची तपशीलवार इनफार्मेशन .

### ट्रबलशूटींग दृष्टीकोन

PC शी निगडीत कोणत्याही समस्या सोडवण्यासाठी, प्रथम त्या PC च्या मॅटेनन्स रेकॉर्डमधून जा आणि समस्या वेगळ्या करण्यासाठी मेथड्स शीर प्रक्रियेचे अनुसरण करा. स्टॅण्डर्ड समस्या सोडवण्याच्या प्रक्रियेमध्ये पुढील स्टेप्स चा समावेश आहे

**समस्या ओळखा:** हा प्रक्रियेचा सर्वात कठीण पार्ट आहे. ही स्टेप यशस्वीरित्या पार पाडण्यासाठी समस्येबद्दलचा सर्व डेटा गोळा करा

**संभाव्य कारणे ओळखा:** समस्येच्या सर्व लक्षणांचे विश्लेषण करा आणि संभाव्यतेपासून कमीतकमी संभाव्यतेपर्यंत सर्व कारणांची यादी करण्याचा प्रयत्न करा.

**संभाव्य उपाय ओळखा:** ओळखल्या गेलेल्या प्रत्येक कारणासाठी उपाय ओळखा. संभाव्य कारणामध्ये एकापेक्षा जास्त संभाव्य उपाय असू शकतात.

**संभाव्य उपायांचे विश्लेषण करा:** जर दोन सोल्यूशन्स समान रिझल्ट देत असतील तर, जो अधिक आर्थिक आहे त्याचा विचार करा आणि तेच लागू करा.

वरील स्टेप्स चे अनुसरण केल्यानंतर आणि समस्या सोडवल्यानंतर देखभाल रेकॉर्ड अपडेट करा आणि त्यात आवश्यक नोंदी करा. काही प्रकरणांमध्ये समस्या अगदी स्पष्ट असू शकतात आणि उपाय अगदी पारदर्शक असू शकतात आणि अशा प्रकरणांमध्ये देखील मेथड्स शीर दृष्टिकोनाचा अवलंब करण्याचा सराव करण्यासाठी वर नमूद केलेल्या स्टेप्स चे अनुसरण करण्याचा प्रयत्न करा. जेव्हा जेव्हा पीसीमध्ये समस्या येते तेव्हा, स्टेप्स चे अनुसरण करताना, प्रश्नांची उत्तरे देऊन सिस्टमबद्दल इनफार्मेशन गोळा करण्याचा प्रयत्न करा.

- कोणत्या परिस्थितीत ही समस्या उद्भवली?
- बीप कोड/एरर मेसेज किंवा कोणतीही स्पष्ट लक्षणे या स्वरूपात काही संकेत होते का?
- समस्या आली तेव्हा कोणते सॉफ्टवेअर ऍक्टिव्ह होते?
- हे प्रथमच घडले आहे की भूतकाळातही घडले आहे?
- सत्रादरम्यान कोणतेही कॉन्फिगरेशन बदल केले गेले होते ज्यासाठी रीस्टार्ट करणे आवश्यक होते जे केले गेले नाही?

### पीसी ऑप्टिमाइझ करणे:

एक पीसी जो पूर्णपणे चांगले काम करत होता आणि मंद होण्याचे लक्षण विकसित केले आहे किंवा जर ते नवीन सॉफ्टवेअरच्या मागणीनुसार काम करू शकत नसेल, तर संभाव्य उपायांपैकी एक म्हणजे पीसीचे कार्यप्रदर्शन सुधारण्यासाठी अपडेट करणे किंवा ऑप्टिमाइझ करणे. ऑप्टिमाइझेशन स्टेप्स साठी पैसे खर्च होऊ शकतात, परंतु अनेकांमध्ये तुमच्या मालकीचे सॉफ्टवेअर किंवा वेबवर सहज सबलब्ध असलेले सॉफ्टवेअर समाविष्ट असते.

### BIOS आणि बूट प्रक्रिया ऑप्टिमाइझ करणे:

BIOS सेटअप कॉन्फिगरेशनमध्ये CMOS मध्ये अनेक सेटिंग्ज समाविष्ट आहेत. सिस्टिम किती लवकर बूट होते आणि कार्य करते हे या सेटिंग्जवर अवलंबून असते. इष्टतम कार्यप्रदर्शनासाठी सिस्टिम कॅशिंग किंवा जलद POST प्रक्रिया वापरणे यासारखी मौल्यवान वैशिष्ट्ये सक्षम करणे अत्यंत आवश्यक आहे.

**हार्ड डिस्क ऑप्टिमाइझ करणे:** Windows ScanDisk आणि Disk Defragmenter युटिलिटीज ही हार्ड डिस्क वापरण्याच्या आणि ऍक्सेस गतीच्या दृष्टीने ऑप्टिमाइझ करण्यासाठी सबलब्ध सर्वोत्तम साधने आहेत. स्कॅनडिस्कचा वापर त्रुटींसाठी डिस्क तपासण्यासाठी आणि त्या दुरुस्त करण्यासाठी किंवा भविष्यातील त्रुटी टाळण्यासाठी वापर सारण्यांमधून डिस्कचे अप्राप्य पार्ट काढून टाकण्यासाठी केला जातो. त्याचप्रमाणे डिस्क डीफ्रॅगमेंटर डेटा फाइलच्या तुकड्यांना अधिक ऑप्टिमाइझ आणि लॉजिकल स्वरूपात मॅनेज करते जे जलद ऍक्सेस वेळ आणि कमी हेड हालचाल प्रदान करते.

**एक्सपान्शन कार्ड ऑप्टिमाइझ करणे:** I/O कंट्रोलर आणि इतर एक्सपान्शन कार्ड्स ऑप्टिमाइझ करण्याचा सर्वोत्तम पाथ म्हणजे त्यांना योग्य क्रमाने इंस्टाल करणे. जरी ते क्रमशः इंस्टाल केले असले तरीही कोणतीही हानी होणार नाही, परंतु त्यांना योग्य क्रमाने ठेवल्याने काही फायदा होऊ शकतो. पॅटियम सिस्टिम वर, PCI कार्ड वापरा आणि शक्य असल्यास ISA कार्ड टाळा. व्हिडिओ कार्ड, साउंड कार्ड, NIC, मोडेम आणि SCSI अडॅप्टर्ससह सर्व I/O अडॅप्टर PCI एक्सपान्शन बससाठी सबलब्ध आहेत. मदरबोर्ड दस्तऐवजीकरणाचा सल्ला घ्या आणि पहिल्या PCI स्लॉटमध्ये व्हिडिओ कार्ड इंस्टाल करा, त्यानंतर NIC, मॉडेम आणि साउंड कार्ड, त्या क्रमाने.

**प्रोसेसर ऑप्टिमाइझ करणे:** खालील मार्गांनी प्रोसेसरचा वेग वाढवू शकतो

- ते अधिक वेगवान किंवा उच्च पातळीच्या प्रोसेसरने बदला
- प्रोसेसरच्या लॉजिकवर पॅच किंवा फिक्सेस लागू करण्यासाठी प्रोसेसर उत्पादकाकडून सबयुक्तता वापरा
- प्रोसेसर ओव्हरक्लॉक करा

प्रोसेसरला उच्च स्तरीय किंवा वेगवान प्रोसेसरने बदलण्याची आवश्यकता अशी आहे की तुमचा मदरबोर्ड आणि चिपसेट नवीन प्रोसेसरला लॉजिकल आणि फिजिकल दृष्ट्या सपोर्ट देतील. लॉजिकल दृष्ट्या चिपसेट आणि मदरबोर्डने प्रोसेसरच्या बस गतीला सपोर्ट दिले पाहिजे आणि त्यासाठी

आवश्यक असलेले सपोर्टिंग सर्किट्स असणे आवश्यक आहे. संपूर्ण मदरबोर्ड पुनस्थित करणे बरेचदा चांगले असते. काही वेळा प्रोसेसरचा निर्माता काही सबयुक्तता सोडू शकतो ज्यामुळे प्रोसेसरच्या क्षमतेचे काही पैलू जसे की व्हिडिओ प्रोसेसिंग, बफर हँडलिंग, कॅशिंग आणि इतर प्रोसेसर आधारित फंक्शन्स सुधारतील. प्रोसेसरला ओव्हरक्लॉक करणे म्हणजे प्रोसेसरला सपोर्ट देण्यासाठी सोडल्यापेक्षा वेगाने चालवणे. बहुतेक प्रोसेसर त्यांच्या नाममात्र वेगापेक्षा जास्त वेगाने धावण्यास सक्षम आहेत. प्रोसेसरचा नाममात्र वेग हा एक स्पेसिफिक चिपसेट, मदरबोर्ड, कूलिंग सिस्टम आणि पीसीच्या इतर एलिमेंट्स सह चालण्यासाठी ट्यून केलेला वेग आहे. प्रोसेसरचा वेग वाढवल्याने प्रोसेसरवर हीटिंग समस्या निर्माण होऊ शकतात आणि वारंवार सिस्टम लॉकअप, मेमरी समस्या आणि इतर समस्या उद्भवू शकतात.

### गैर-सॉफ्टवेअर समस्यांचे समस्यानिवारण सोर्स :

कोणत्याही उघड सॉफ्टवेअर कारणास्तव पीसी कधीही अयशस्वी झाल्यास, खालील एरिया तपासा - योग्य एसी पॉवर सुनिश्चित करा

- संगणक व्हायरससाठी पीसी स्कॅन करा

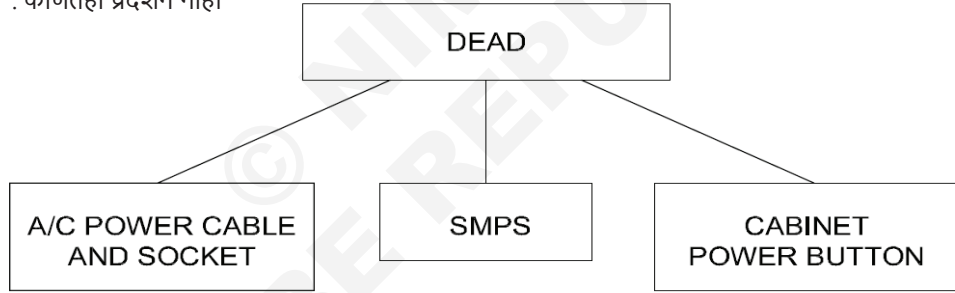
- CPU फॅन फिरत असल्याची खात्री करा
- एक्सटर्नल I/O कनेक्टरचे योग्य कनेक्शन सुनिश्चित करा
- एक्सपान्शन कार्ड रिसेट करा आणि इंटरनल उपकरणांची पॉवर आणि डेटा केबल तपासा
- बहुतेक बूट समस्या अलीकडील बदलाचे रिझल्ट आहेत, BIOS सेटअप कॉन्फिगरेशन डेटा तपासा
- कोणतेही नवीन हार्डवेअर किंवा सॉफ्टवेअर इंस्टाल करण्यासाठी, ज्ञात संघर्ष किंवा विसंगततेसाठी निर्मात्याच्या वेबसाइटला भेट द्या
- कोणतेही नवीन हार्डवेअर किंवा सॉफ्टवेअर इंस्टाल केले असल्यास कोणत्याही संसाधन संघर्षासाठी तपासा.

ट्रबलशूटिंग PC वरील पुढील धडे विविध टाइप च्या समस्यांसाठी प्रॉब्लेम ट्री प्रदान करतात ज्या यूजर ला सामान्यपणे तोंड द्यावे लागते. स्पेसिफिक समस्या असलेले प्रत्येक ट्री संशयित किंवा समस्येचे निराकरण करण्यासाठी सर्च ल्या जाणाऱ्या क्षेत्रांचे विश्लेषण करण्यास वाव देते. या धड्यात मॉनिटरवर डिस्प्ले नसलेल्या पीसीसाठी प्रॉब्लेम ट्री समाविष्ट आहे.

### समस्या ट्री : PT- 01

लक्षात आलेले लक्षण : मृत

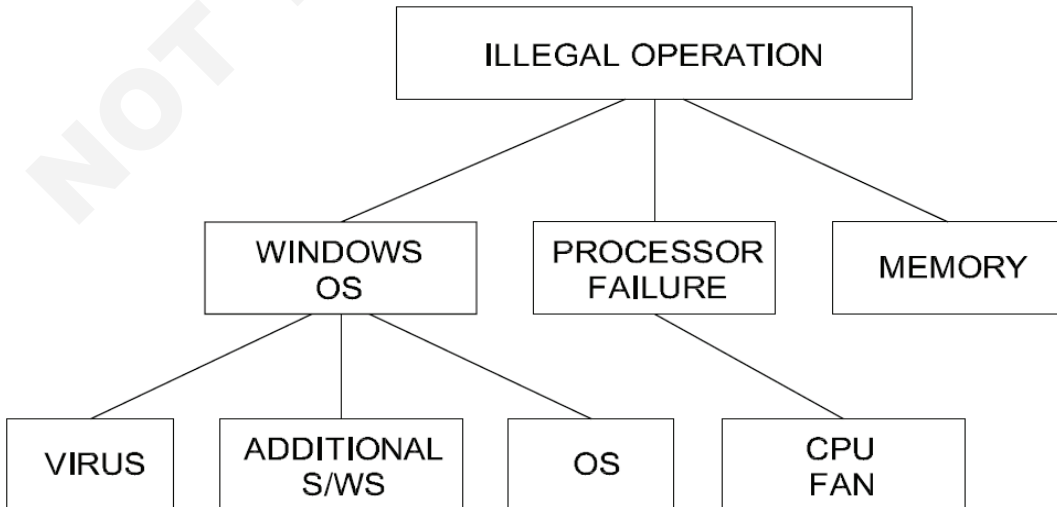
अतिरिक्त लक्षण : कोणतेही प्रदर्शन नाही



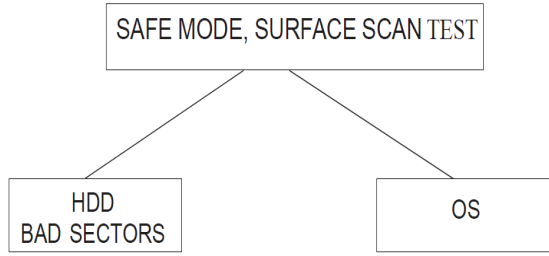
### समस्या ट्री : PT- 02

निरीक्षण केलेले लक्षण : विंडोज "बेकायदेशीर ऑपरेशन" दर्शवते

अतिरिक्त लक्ष : विंडोज काम करत नाही



तक्रारीसाठी ट्रबलशूटिंग चार्ट (TSC-02) आणि सर्विस प्रवाह क्रम (SFS 02) वर चर्चा करा कार्यरत, विंडो त्रुटी दर्शविते - अवैध ऑपरेशन".  
वरील तक्रारीसाठी चर्चा केलेले विविध दोष तुम्हाला दिलेल्या वास्तविक संगणकावर लागू केले जातील व्यावहारिक एक्सरसाईस .

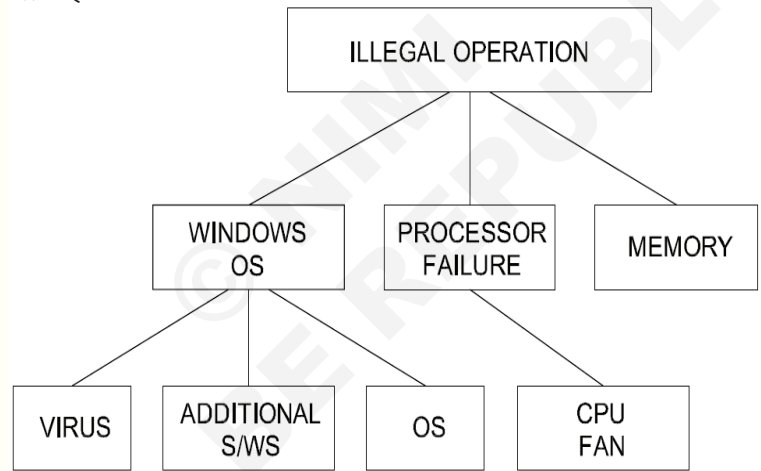


तक्रारीसाठी ट्रबलशूटिंग चार्ट (TSC-01) आणि सर्विस प्रवाह क्रम (SFS 01) वर चर्चा करा "केव्हा विंडोज सुरू झाली आहे सिस्टि पृष्ठपार्ट टेस्टिंग चालवते आणि सुरक्षित मोडवर जाते" एक्सरसाईस 2.33 चा रेफरेन्स देते  
वरील तक्रारीसाठी चर्चा केलेले विविध दोष तुम्हाला दिलेल्या वास्तविक संगणकावर लागू केले जातील व्यावहारिक एक्सरसाईस .

समस्या ट्री : PT- 02

निरीक्षण केलेले लक्षण : विंडोज "बेकायदेशीर ऑपरेशन" दर्शवते

अतिरिक्त लक्षण : विंडोज काम करत नाही

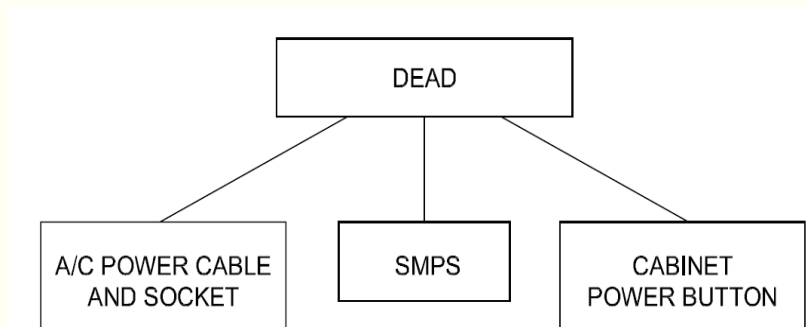


तक्रारीसाठी ट्रबलशूटिंग चार्ट (TSC-02) आणि सर्विस प्रवाह क्रम (SFS 02) वर चर्चा करा कार्यरत, विंडो त्रुटी दर्शविते - अवैध ऑपरेशन"  
वरील तक्रारीसाठी चर्चा केलेले विविध दोष तुम्हाला दिलेल्या वास्तविक संगणकावर लागू केले जातील व्यावहारिक एक्सरसाईस

समस्या ट्री : PT- 03

निरीक्षण केलेले लक्षण : DVD ड्राइव्ह रीड करू शकत नाही

अतिरिक्त लक्षण :



## विंडोजसाठी आवश्यक ॲप्लिकेशन सॉफ्टवेअर इन्स्टॉल करा उदा. ऑफिस पॅकेज, पीडीएफ रीडर, मीडिया प्लेयर (Install necessary application software for Windows i.e. Office Package, PDF Reader, Media Player etc)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- विविध सॉफ्टवेअर प्रकारांचे डिस्क्रिप्शन करा.

### ॲप्लिकेशन सॉफ्टवेअर

ॲप्लिकेशन सॉफ्टवेअर ही एक संज्ञा आहे जी स्पेसिफिक हेतूसाठी तयार केलेल्या सॉफ्टवेअरसाठी वापरली जाते. हा सामान्यतः अंतिम वापरकर्त्याद्वारे वापरल्या जाणाऱ्या प्रोग्रामचा किंवा प्रोग्रामचा संग्रह असतो. याला ॲप्लिकेशन किंवा फक्त ॲप म्हणता येईल.

किंबहुना सिस्टम सॉफ्टवेअर आणि प्रोग्रामिंग सॉफ्टवेअर व्यतिरिक्त इतर सर्व सॉफ्टवेअर्स हे ॲप्लिकेशन सॉफ्टवेअर आहेत.

### ॲप्लिकेशन सॉफ्टवेअर व्याख्या

स्पेसिफिक फंक्शन करण्यासाठी यूजर स मदत करण्यासाठी विकसित केलेल्या सॉफ्टवेअरला ॲप्लिकेशन सॉफ्टवेअर म्हणतात.

### विविध टाइप च्या ॲप्लिकेशन सॉफ्टवेअरमध्ये खालील गोष्टींचा समावेश होतो:

ॲप्लिकेशन सॉफ्टवेअर टाइप	उदाहरणे
प्रोसेसिंग सॉफ्टवेअर	सामग्री प्रवेश सॉफ्टवेअर ब्राउझर
डेटाबेस सॉफ्टवेअर	एमएस वर्ड, वर्डपॅड आणि नोटपॅड
स्प्रेडशीट सॉफ्टवेअर	ओरॅकल, एमएस ॲक्सेस ड
मल्टीमीडिया सॉफ्टवेअर	ॲपल क्रमांक, मायक्रोसॉफ्ट एक्सेल
प्रेसेंटेशन सॉफ्टवेअर	रिअल प्लेयर, मीडिया प्लेयर
ॲंटरप्राइझ सॉफ्टवेअर	मायक्रोसॉफ्ट पॉवर पॉइंट, कीनोट्स
इन्फॉर्मेशन वर्कर सॉफ्टवेअर	ग्राहक संबंध मॅनेजमेन्ट सिस्टिम
शैक्षणिक सॉफ्टवेअर	डॉक्युमेंटेशन साधने, संसाधन मॅनेजमेन्ट साधने
सिम्युलेशन सॉफ्टवेअर	शब्दकोश: एन्कार्टा, ब्रिटानिका मॅथमॅटिकल: MATLAB Others:
ॲप्लिकेशन सूट	Google Earth, NASA World Wind फ्लाइट आणि वैज्ञानिक सिम्युलेटर
अभियांत्रिकी आणि उत्पादन विकासासाठी सॉफ्टवेअर	कन्टेन्ट ॲक्सेस सॉफ्टवेअर ब्राउझर ॲप्लिकेशन सूट
	अभियांत्रिकी आणि प्रॉडक्शन डेव्हलपमेंट साठी सॉफ्टवेअर
	मीडिया प्लेयर्स, वेबद्वारे कंटेंट मध्ये एक्सेस करणे
	ओपनऑफिस, मायक्रोसॉफ्ट ऑफिस
	IDE किंवा इंटीग्रेटेड विकास पर्यावरण

परवानाकृत, विकले जाणारे, फ्रीवेअर, शेअरवेअर आणि ओपन सोर्स असे विविध टाइप चे ॲप्लिकेशन सॉफ्टवेअर आहेत.

ॲप्लिकेशन सॉफ्टवेअर एकतर इंस्टाल करणे आवश्यक आहे किंवा ते ऑनलाइन चालवू शकतात. ॲप्लिकेशन सॉफ्टवेअर खालील वापराच्या आधारावर देखील ओळखले जाऊ शकते:

- युटिलिटी प्रोग्रॅम
- सामान्य प्रोग्रॅम
- इंटीग्रेटेड प्रोग्रॅम
- स्पेसिफिक सॉफ्टवेअर
- बेस्पोक सॉफ्टवेअर
- वर्ड प्रोसेसिंग सॉफ्टवेअर

- डेस्कटॉप प्रकाशन सॉफ्टवेअर
- स्प्रेडशीट सॉफ्टवेअर
- डेटाबेस सॉफ्टवेअर
- प्रेसेंटेशन सॉफ्टवेअर
- इंटरनेट ब्राउझर
- ईमेल प्रोग्राम्स
- ग्राफिक प्रोग्राम (पिक्सेल आधारित)
- ग्राफिक प्रोग्राम (वेक्टर आधारित)
- कम्युनिकेशन सॉफ्टवेअर: ऑडिओ, व्हिडिओ किंवा चॅट आधारित माध्यमांद्वारे कम्युनिकेशन

## ब्लूटूथ आणि वाय-फाय सेटिंग्ज कॉन्फिगर करा (Configure Bluetooth and Wi-Fi settings)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- ब्लूटूथचा अर्थ सांगा
- ब्लूटूथ वापरण्याच्या मेथड्स चे डिस्क्रिप्शन करा
- ब्लूटूथच्या प्रमुख एप्लिकेशन ची यादी करा.

### परिचय

ब्लूटूथ हे निश्चित आणि मोबाईल उपकरणांमधून कमी अंतरावर डेटाची देवाणघेवाण करण्यासाठी (2.4 ते 2.485 GHz रेन्ज तील शॉर्ट-वेव्हलेंथ UHF रेडिओ लहरी वापरून) आणि पर्सनल एरिया नेटवर्क (PAN) तयार करण्यासाठी एक वायरलेस तंत्रज्ञान स्टॅण्डर्ड आहे. हे एक स्टॅण्डर्ड वायर-रिप्लेसमेंट कम्युनिकेशन प्रोटोकॉल आहे जे प्रामुख्याने कमीसाठी डिझाइन केलेले आहे

प्रत्येक यंत्रामध्ये कमी किमतीच्या ट्रान्सीव्हर मायक्रोचिपवर आधारित लहान रेन्ज सह वीज वापर.

ते एकाच वेळी आठ उपकरणे (इलेक्ट्रॉनिक उपकरणांच्या वस्तू) जोडू शकतात. ही चिप संगणक, डिजिटल कॅमेरे, मोबाईल फोन आणि फॅक्स यासारख्या वस्तूंमध्ये प्लग केली जाऊ शकते. ब्लूटूथ स्पेसिफिक परिस्थितींमध्ये स्पेशल तः सोयीस्कर आहे - उदाहरणार्थ, केबल्सशिवाय एका मोबाइल फोनवरून फायली स्थानांतरित करताना. पीसी आणि मोबाईल फोन दरम्यान संगीत आणि फोटो पाठवणे हे आणखी एक सबयुक्त एप्लिकेशन आहे.

उपकरणे रेडिओ (ब्रॉडकास्ट) कम्युनिकेशन सिस्टिम वापरत असल्यामुळे, ते एकमेकांच्या दृष्टीक्षेपात असण्याची गरज नाही, तथापि अर्ध ऑप्टिकल वायरलेस पाथ व्यवहार्य असणे आवश्यक आहे.

रेन्ज ही पॉवर-क्लास- डिपेंडेंट आहे, परंतु प्रभावी रेन्ज 10 ते 100 मीटर पर्यंत बदलू शकतात.

'ब्लूटूथ' हे नाव तंत्रज्ञानाच्या स्कॅन्डिनेव्हियन उत्पत्तीचे प्रतिबिंबित करते. त्याचे नाव 10व्या शतकातील डॅनिश वायकिंग, किंग हॅराल्ड ब्लाटॅंड (इंग्रजीमध्ये 'ब्लूटूथ' म्हणून भाषांतरित) याच्या नावावरून ठेवण्यात आले आहे. त्याने डेन्मार्क आणि नॉर्वे यांना एकत्र केले आणि कंट्रोल केले, म्हणूनच ब्लूटूथद्वारे डिव्हाइसेसना एकत्र करणे ही संड्वेड्स आहे.

### ब्लूटूथ वापरणे

ब्लूटूथ वापरण्यासाठी, डिव्हाइस ब्लूटूथ सक्षम असणे आवश्यक आहे. यासाठी "डोंगल" नावाचे सबकरण वापरले जाऊ शकते. डोंगल हे असे सबकरण आहे जे संगणकाला ब्लूटूथ वापरण्यास सक्षम करण्यासाठी प्लग इन करते. सुसंगत डिव्हाइसेसच्या प्रत्येक निर्मात्याकडे ब्लूटूथमध्ये एक्सेस

करण्यासाठी त्यांच्या स्वतःच्या सूचना असतील. तपशीलवार सूचनांसाठी तुम्हाला मॅन्युअल पाहावे लागेल, परंतु सामान्य पाथ दर्शक म्हणून:

### ब्लूटूथ सेट करण्यासाठी

(डिव्हाइसवर ब्लूटूथ आयकॉन ओळखा.)

- 1 ब्लूटूथ चालू करा किंवा सक्षम करा. तुमचे डिव्हाइस 'व्हिजिबल' आहे आणि 'हिडन नाही' याची खात्री करा, जेणेकरून इतर जवळपासची डिव्हाइस सिग्नल उचलू शकतील.
- 2 इतर सुसंगत उपकरणांशी कनेक्ट करताना ते ओळखण्यासाठी तुमच्या डिव्हाइसला नाव द्या.

जेव्हा मोबाईल फोन, लॅपटॉप, टॅबलेट इत्यादी उपकरणे ब्लूटूथ सक्षम करतात तेव्हा ब्लूटूथ चिन्ह चालू असतो.

### ब्लूटूथ कनेक्शन इंस्टाल करण्यासाठी

- 1 तुम्हाला पाठवायची असलेली फाइल शोधा.
- 2 ते Bluetooth द्वारे पाठवण्याचा पर्याय निवडा - तुमचे डिव्हाइस रेंजमध्ये इतर डिव्हाइसेस शोधेल आणि ते प्रदर्शित करेल.
- 3 तुम्ही ज्या इक्विपमेंट्स शी जोडू इच्छिता त्या इक्विपमेंट्स कडे स्क्रोल करा आणि ते निवडा.
- 4 इतर डिव्हाइसला 'पेरिग'ची आवश्यकता असल्यास, तुम्हाला पासकोड प्रविष्ट करणे आवश्यक आहे - थोडासा पिन क्रमांकासारखा - आणि तो दुसऱ्या डिव्हाइसवर प्रविष्ट केला आहे याची खात्री करा.

कनेक्शन इंस्टाल झाल्यावर, डेटा पाठविणे सुरू होईल. आपल्याला उपकरणांमधील स्पष्ट दृष्टीची काळजी करण्याची आवश्यकता नाही.

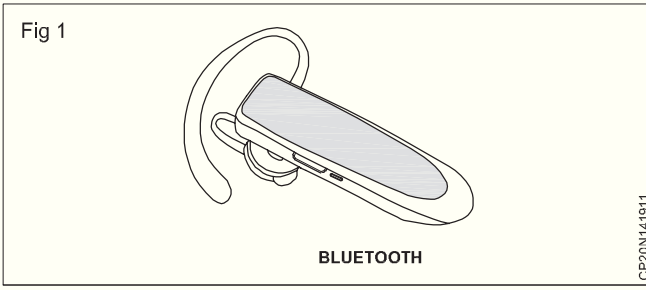
### ब्लूटूथ एप्लिकेशन ची यादी

काही ब्लूटूथ एप्लिकेशन्स खालीलप्रमाणे आहेत:

मोबाईल फोन आणि हँड्सफ्री हेडसेटचे वायरलेस कंट्रोल आणि संवाद. (आकृती क्रं 1)

- मोबाइल फोन आणि ब्लूटूथ सुसंगत कार स्टिरिओ सिस्टम यांच्यातील वायरलेस कंट्रोल आणि संवाद.



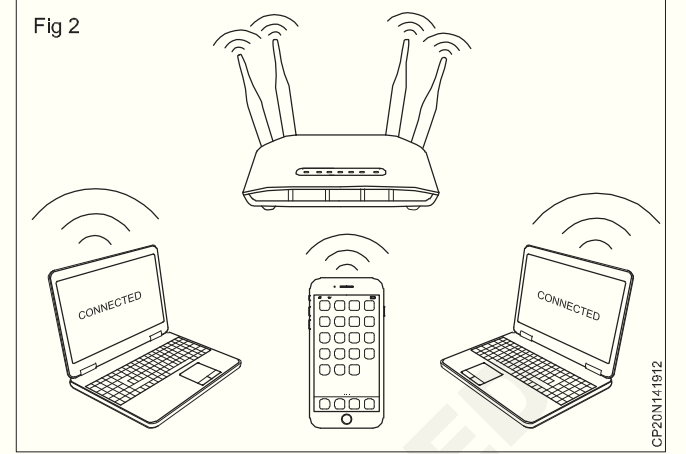


- टॅब्लेट आणि स्पीकर जसे की iPad आणि Android डिव्हाइसेसचे वायरलेस कंट्रोल आणि संवाद.
- मर्यादित जागेत आणि जेथे कमी बँडविड्थ आवश्यक असेल तेथे पीसी दरम्यान वायरलेस नेटवर्किंग.
- PC इनपुट आणि आउटपुट उपकरणांसह वायरलेस कम्युनिकेशन, सर्वात सामान्य म्हणजे माउस, कीबोर्ड आणि प्रिंटर.
- फाइल्स, संपर्क तपशील, कॅलेंडर भेटी, आणि OBEX (ऑब्जेक्ट एक्सचेंज) डिव्हाइसेस दरम्यान स्मरणपत्रांचे हस्तांतरण.
- टेस्टिंग उपकरणे, GPS रिसेव्हर्स, वैद्यकीय उपकरणे, बार कोड स्कॅनर आणि वाहतूक कंट्रोल उपकरणांमध्ये मागील वायर्ड RS-232 सीरियल कम्युनिकेशन्स बदलणे.
- दोन औद्योगिक इन्टरनेट नेटवर्कमधील वायरलेस ब्रिज.
- गेमिंग कन्सोलमध्ये वायरलेस कंट्रोलर.
- चोरी किंवा वस्तू हरवण्यापासून रोखण्यासाठी मोबाइल फोनवर वैयक्तिक सुरक्षा एप्लिकेशन. संरक्षित आयटममध्ये ब्लूटूथ मार्कर (उदा. एक टॅग) असतो जो फोनशी सतत संवाद साधत असतो. जर कनेक्शन तुटले असेल (मार्कर फोनच्या रेन्ज बाहेर असेल) तर अलार्म वाजवला जातो.

### वायफाय

वाय-फाय हे एक लोकप्रिय वायरलेस नेटवर्किंग तंत्रज्ञान आहे. Wi-Fi चा अर्थ "वायरलेस फिडेलिटी" आहे. नेदरलँड्समधील NCR कॉर्पोरेशन/AT&T ने 1991 मध्ये Wi-Fi चा शोध लावला होता. या तंत्रज्ञानाचा वापर करून आपण दोन किंवा अधिक उपकरणांमधील इनफार्मेशन ची देवाणघेवाण करू शकतो. मोबाइल संगणकीय उपकरणांसाठी वाय-फाय विकसित केले गेले आहे, जसे की लॅपटॉप आहेत, परंतु ते आता मोबाइल एप्लिकेशन आणि टेलिव्हिजन, डीव्हीडी प्लेयर आणि डिजिटल कॅमेरे यांसारख्या ग्राहक इलेक्ट्रॉनिक्ससाठी मोठ्या प्रमाणावर वापरले जात आहे. वाय-फाय कनेक्शनशी कम्युनिकेशन करण्याच्या दोन शक्यता असू शकतात जे क्लायंट कनेक्शनच्या अॅक्सेस पॉईंटद्वारे किंवा क्लायंट टू क्लायंट कनेक्शनद्वारे असू शकतात. वाय-फाय हे एक टाइप चे वायरलेस तंत्रज्ञान आहे. याला सामान्यतः वायरलेस लॅन (लोकल एरिया नेटवर्क) असे म्हणतात. वाय-फाय परवानगी देते केबल आणि वायरिंगशिवाय ऑपरेट करण्यासाठी लोकल एरिया नेटवर्क. हे घर आणि व्यवसाय नेटवर्कसाठी

लोकप्रिय निवड करत आहे. संगणकाचे वायरलेस अडॉप्टर डेटा रेडिओ सिग्नलमध्ये हस्तांतरित करतो आणि वापरकर्त्यासाठी अँटेनामध्ये डेटा हस्तांतरित करतो. (पिक्चर 2)

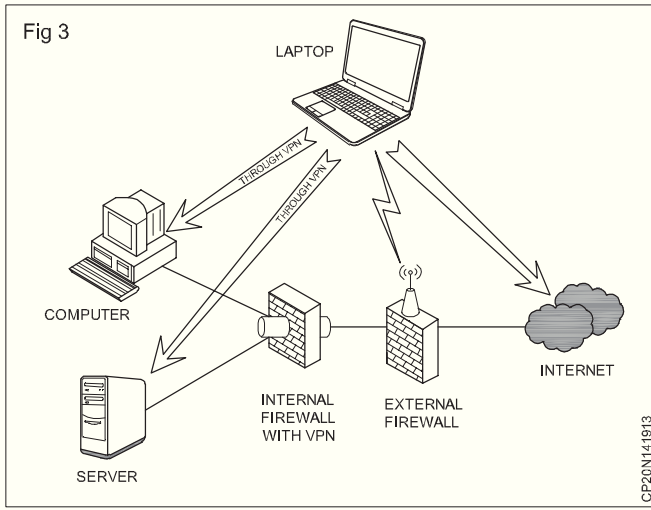


### वोर्किंग प्रिसिपल

वाय-फाय हे हाय स्पीड इंटरनेट कनेक्शन आणि कोणत्याही केबल किंवा वायरचा वापर न करता नेटवर्क कनेक्शन आहे. वायरलेस नेटवर्क रेडिओ सिग्नल, अँटेना आणि राउटर हे तीन आवश्यक एलिमेंट कार्यरत आहे. रेडिओ लहरी या की आहेत ज्यामुळे वाय-फाय नेटवर्किंग शक्य होते. वाय-फाय कार्डसह संगणक आणि सेल फोन तयार आहेत. वाय-फाय सुसंगतता समुदाय नेटवर्कशी जोडलेल्या ग्राउंडमधील एलिमेंटसाठी नवीन निर्मिती वापरत आहे. वास्तविक प्रसारण मध्ये सह कनेक्ट केलेले आहे

खरं तर ते स्टिरिओ सिस्टीम सर्फ तसेच मॉनिटर टू क्लासिफिकेशन प्रोन असलेल्या वायर्सच्या मूल्याद्वारे पूर्ण केले जाते. वाय-फाय व्यक्तीला वास्तविक प्रदान केलेल्या क्षेत्रातील कोणत्याही ठिकाणी वेबवर एक्सेस मिळविण्याची परवानगी देते. आणि आता रिसॉर्ट्स, लायब्ररी, शाळा, महाविद्यालये, कॅम्पस, वैयक्तिक संस्था, तसेच एस्प्रेसो स्टोअर्स तसेच खुल्या सार्वजनिक ठिकाणी कंपनीला अधिक किफायतशीर बनवण्यासाठी तसेच त्यांच्या स्वतःच्या ग्राहकांशी संवाद साधण्यासाठी एक सिस्टिम तयार करू शकतात. केव्हाही वाय-फाय सुसंगतता त्यांच्या प्रेरणादायी केबल टेलिव्हिजनचा वापर करून कंपनीकडे टक लावून सर्फ करू शकते.

रेडिओ सिग्नल अँटेना आणि राउटरमधून प्रसारित केले जातात जे सिग्नल वाय-फाय रिसेव्हर्सद्वारे उचलले जातात, जसे की संगणक आणि सेल फोन आहेत जे वाय-फाय कार्डसह तयार आहेत. जेव्हा जेव्हा संगणकाला राउटरसाठी 100-150 फूट मर्यादित सिग्नल प्राप्त होतात तेव्हा ते डिव्हाइसला त्वरित कनेक्ट करते. वाय-फाय ची रेन्ज वातावरण, घरातील किंवा बाहेरील रेन्ज वर अवलंबून असते. वाय-फाय कार्ड सिग्नल वाचतील आणि युजर आणि नेटवर्क दरम्यान इंटरनेट कनेक्शन तयार करतील. वाय-फाय कनेक्शन वापरणाऱ्या डिव्हाइसचा वेग वाढतो कारण संगणक मुख्य स्त्रोताच्या जवळ जातो आणि वेग कमी होतो आणि संगणक आणखी दूर जातो. (पिक्चर 3)



### सेक्युरीटी:

वाय-फाय तंत्रज्ञानामध्ये सेक्युरीटी हा एलिमेंट आहे. सुरक्षा हा आमचा वैयक्तिक निर्णय आहे परंतु वायरलेस कनेक्शन असल्याने आम्ही आमचे प्राइवेट तपशील सुरक्षित ठेवण्याकडे लक्ष दिले पाहिजे. आम्ही असुरक्षित वायरलेस राउटरशी सहजपणे कनेक्ट करू शकतो. गेम डाउनलोड करणे, ॲप्स डाउनलोड करणे आणि दहशतवादी कारवायांचे नियोजन करणे, बेकायदेशीर संगीत आणि मूव्ही फाइल्स इ. शीरिंग करणे इत्यादी डेटा वापरून कोणीही वायरलेस राउटरशी कनेक्ट केलेले आहे. त्यामुळे वायरलेस तंत्रज्ञानावर आधारित उपकरणांना सुरक्षा प्रदान करणे आवश्यक आहे.

© NIMI  
NOT TO BE REPUBLISHED

## डीव्हीडी, सीडी आणि बर्निंग डीव्हीडी (DVDs, CDs and burning DVDs)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- CDs आणि DVDs च्या वैशिष्ट्यांचे डिस्क्रिप्शन करा
- DVD चे मुख्य फॉरमॅट डिस्क्रिप्शन करा
- CD च्या वैशिष्ट्यांचे डिस्क्रिप्शन करा
- DVD बर्निंगचे डिस्क्रिप्शन करा.

### परिचय

DVD (कधीकधी "डिजिटल व्हिडिओ डिस्क" किंवा "डिजिटल व्हर्सटाइल डिस्क" म्हणून ओळखले जाते) हे डिजिटल ऑप्टिकल डिस्क स्टोरेज फॉरमॅट आहे. PC आणि स्टँडअलोन प्लेयर्ससह अनेक टाइप च्या प्लेअरसह DVDs वापरल्या जाऊ शकतात.

या डिस्कस DVD-ROM म्हणून ओळखल्या जातात, कारण डेटा फक्त वाचला जाऊ शकतो आणि लिहीता किंवा पुसला जाऊ शकत नाही. रिक्त रेकॉर्ड करण्यायोग्य DVD डिस्कस (DVD-R आणि DVD+R) एकदा DVD रेकॉर्डर वापरून रेकॉर्ड केल्या जाऊ शकतात आणि नंतर DVD-ROM म्हणून कार्य करू शकतात. पुनर्लेखन करण्यायोग्य DVDs (DVD-RW, DVD+RW, आणि DVD-RAM) अनेक वेळा रेकॉर्ड आणि मिटवल्या जाऊ शकतात.

### डीव्हीडी वैशिष्ट्ये आणि फॉरमॅट

डीव्हीडीचा वापर डीव्हीडी-व्हिडिओ कन्ड्युमर डिजिटल व्हिडिओ फॉरमॅटमध्ये आणि डीव्हीडी-ऑडिओ कन्ड्युमर डिजिटल ऑडिओ फॉरमॅटमध्ये केला जातो. ते AVCHD फॉरमॅट कॅमकॉर्डरसह वापरल्या जाणाऱ्या स्पेशल AVCHD फॉरमॅट (एडवान्स व्हिडिओ कोडिंग हाय डेफिनिशन) मध्ये देखील वापरले जाऊ शकतात. इतर टाइप ची इनफार्मेशन असलेल्या डीव्हीडीला DVD डेटा डिस्कस म्हणून संबोधले जाऊ शकते.

ठराविक रेकॉर्ड करण्यायोग्य DVD मध्ये सुमारे 4.7 गीगाबाइट्स (GB) असू शकतात. तथापि, डिस्कवर फाइल्स बर्न करण्यासाठी तुम्ही वापरू शकता अशा एकूण डिस्क जागेची नंबर डिस्कवरच लिस्ट बद्ध केलेल्या रकमेपेक्षा कमी आहे. याचे कारण असे की जेव्हा डिस्कची क्षमता संगणकात वापरली जाते तेव्हा ती वेगळ्या मेथड्स ने मोजली जाते. उदाहरणार्थ, 4.7 GB ची लिस्ट बद्ध डिस्क क्षमता असलेली ठराविक DVD-R डिस्कवर फक्त 4.37 GB डेटा संचयित करू शकते. DVDs अंदाजे 4.7 GB ची स्टोरेज क्षमता देतात. डीव्हीडी डिस्क कालांतराने खराब होत नाहीत आणि चुंबकीय क्षेत्रामुळे प्रभावित होत नाहीत.

वापरल्या जाणाऱ्या रेकॉर्ड करण्यायोग्य डिस्कचा टाइप काही भिन्न एलिमेंटवर अवलंबून असतो, जसे की:

- डिस्क बर्नरसह कार्य करणाऱ्या रेकॉर्ड करण्यायोग्य डिस्कचे टाइप .

- संगणक किंवा इक्विपमेंट्स वरील डिस्क ड्राइव्ह डिस्क बर्न केल्यानंतर वाचेल.

- डिस्कवर बर्न केलेल्या सर्व फाइल्सचा एकूण आकार.

डीव्हीडी बर्निंग: ऑप्टिकल डिस्क (CD किंवा DVD) वर सोर्स कंटेंट रेकॉर्ड करण्याच्या प्रक्रियेला बर्निंग/राइटिंग किंवा ऑप्टिकल डिस्क ऑथरिंग म्हणतात. ऑप्टिकल डिस्क तयार करताना प्रथम ऑप्टिकल डिस्कसाठी डिझाइन केलेल्या संपूर्ण फाइल सिस्टमसह डिस्क प्रतिमा तयार करणे समाविष्ट असते आणि नंतर प्रत्यक्षात प्रतिमा डिस्कवर बर्न करणे. डिस्क प्रतिमा तयार करण्यासाठी आणि फाइल्स बर्न करण्यासाठी अनेक प्रोग्राम्स एकत्रित एप्लिकेशन म्हणून सबलब्ध आहेत.

डीव्हीडी ज्या वेगाने लिहिली जाऊ शकते ती गुणक म्हणून व्यक्त केली जाते: 16X म्हणजे ती प्ले करण्यापेक्षा 16 पट वेगवान. 52X पर्यंतचा वेग देखील खूप सामान्य आहे.

### सीडी किंवा डीव्हीडी फॉरमॅट

डीव्हीडी बर्न करण्यासाठी, दोन मुख्य डिस्क फॉरमॅट्स वापरतात:

- लाईव्ह फाइल सिस्टिम आणि
- मास्टर्ड डिस्क फॉरमॅट्स.

### लाईव्ह फाइल सिस्टिम फॉरमॅट

लाईव्ह फाइल सिस्टिम फॉरमॅट वापरणाऱ्या डिस्कस बहुतेक वेळा अधिक सोयीस्कर असतात कारण तुम्ही USB फ्लॅश ड्राइव्हसह करू शकता त्याप्रमाणे तुम्ही निवडलेल्या फाइल्स लगेच आणि तुम्हाला पाहिजे तितक्या वेळा कॉपी करू शकता. तुम्हाला तुमच्या सीडी, डीव्हीडी किंवा ब्ल्यू रे डिस्क बर्नरमध्ये डिस्क ठेवायची असेल आणि गरज पडेल तेव्हा फाइल्स कॉपी करायच्या असल्यास हे सोयीचे आहे. या फॉरमॅटमध्ये तुम्ही फाइल्स पुन्हा पुन्हा कॉपी आणि मिटवू शकता. तथापि, लाईव्ह फाइल सिस्टिम ऑप्टिकल डिस्क फॉरमॅट केवळ Windows 7, Windows Vista आणि Windows XP सिस्टिम द्वारे वाचनीय (रिडेबल) आहे. या डिस्कस फॉरमॅट केल्यानंतर रिक्त (ब्लॅक) नसतात.

या फॉरमॅटमध्ये फॉरमॅट केलेल्या डिस्कमध्ये डिस्क बर्न करा डायलॉग बॉक्समध्ये पर्याय नाव आहे: "USB फ्लॅश ड्राइव्हसारखे."

## मास्टर्ड डिस्क फॉरमॅट

जर आम्हाला एक ऑप्टिकल डिस्क तयार करायची असेल जी डेटा फाइल्स विंडोज ऑपरेटिंग सिस्टमच्या जुन्या आवृत्त्यांमध्ये किंवा दुसऱ्या ऑपरेटिंग सिस्टममध्ये स्थानांतरित करण्यासाठी वापरली जाऊ शकते, तर आम्हाला मास्टर्ड ऑप्टिकल डिस्क फॉरमॅट वापरण्याची आवश्यकता आहे. तसेच जर आम्हाला संगीत किंवा चित्रे बर्न करायची असतील आणि नियमित CD, DVD, किंवा ब्लू रे डिस्क प्लेयर्समध्ये डिस्क वापरायची असेल जी MP3 फाइल्स प्ले करू शकतील आणि डिजिटल चित्रे दाखवू शकतील, तर आम्ही मास्टर्ड ऑप्टिकल डिस्क फॉरमॅट वापरला पाहिजे.

मास्टर्ड फॉरमॅट Windows XP मधील सीडी बर्न करण्यासारखे कार्य करते. दुसऱ्या शब्दांत, जेव्हा आपण डिस्क लिहितो, तेव्हा आम्ही एकाच वेळी एक फाइल किंवा फाइल्सचा समूह ऑप्टिकल डिस्कवर कॉपी करतो. एकदा हे पूर्ण झाल्यावर, डिस्क बंद होते आणि आम्ही डिस्कवर अधिक फायली कॉपी करू शकत नाही किंवा आम्ही विद्यमान फायली हटवू शकत नाही. म्हणून फायली त्वरित कॉपी न करण्याची शिफारस केली जाते; डिस्कवर कॉपी करणे आवश्यक असलेल्या फाइल्सचा संपूर्ण संग्रह एकत्र करणे आणि नंतर त्या सर्व एकाच वेळी बर्न करणे ही चांगली सराव आहे.

मास्टर्ड पर्यायासह फॉरमॅट केलेल्या डिस्कला बर्न अ डिस्क डायलॉग बॉक्समध्ये पर्याय आहे: "सीडी/डीव्हीडी प्लेयरसह."

डेटा, संगीत, व्हिडिओ आणि ऑडिओ डिस्क तयार करण्यासाठी अनेक टाइप ची साधने सबलब्ध आहेत. आम्ही एकापेक्षा जास्त डिस्कवर पसरलेले बॅकअप देखील तयार करू शकतो, ऑडिओ सीडी मधून म्युझिक ट्रॅक रिप करू शकतो आणि डिस्क इमेजेस वेगवेगळ्या फॉरमॅटमध्ये बनवू किंवा बर्न करू शकतो. ते WAV, MP3, FLAC, WMA फाइल्स, डिस्क कॉपी, कॉम्प्रेस्ड फाइल बॅकअप आणि रिस्टोअर, डिस्क मिटवणे, VCD/SVCD सपोर्ट, प्रोजेक्ट बर्निंग इ. मधून ऑटोमॅटिक ऑडिओ रूपांतरण यांसारखी वैशिष्ट्ये देखील प्रदान करू शकतात.

## ब्लू-रे डिस्क

ब्लू-रे, ज्याला ब्लू-रे डिस्क (बीडी) असेही म्हणतात, हे नवीन ऑप्टिकल डिस्क स्वरूपाचे नाव आहे जे वेगाने DVD बदलत आहे. हाय-डेफिनिशन व्हिडिओ (HD) चे रेकॉर्डिंग, पुनर्लेखन आणि प्लेबॅक सक्षम करण्यासाठी तसेच मोठ्या प्रमाणात डेटा संचयित करण्यासाठी हे फॉरमॅट विकसित केले गेले. हे फॉरमॅट पारंपारिक DVD च्या पाचपट स्टोरेज क्षमता देते आणि सिंगल-लेयर डिस्कवर 25GB आणि ड्युअल-लेयर डिस्कवर 50GB पर्यंत ठेवू शकते.

ब्लू-रे हे नाव बेसिक तंत्रज्ञानावरून घेतले गेले आहे, जे डेटा वाचण्यासाठी आणि लिहिण्यासाठी ब्लू व्हायलेट लेसरचा वापर करते. हे नाव "ब्लू" (ब्लू-व्हायलेट लेसर) आणि "रे" (ऑप्टिकल किरण) यांचे मिश्रण आहे. त्यांना "ब्लू-रे" डिस्क किंवा बीडी असे संबोधले जाते.

खालील फॉरमॅट्स सध्याच्या ब्लू-रे डिस्क स्पेसिफिकेशनचा पार्ट आहेत: 1 BD-ROM - HD पिक्चर पट, गेम्स, सॉफ्टवेअर इत्यादींच्या डिस्ट्रीब्युशन साठी केवळ-वाचनीय (रिडेबल) फॉरमॅट . 2 BD-R - HD व्हिडिओ रेकॉर्डिंग आणि पीसी डेटासाठी रेकॉर्ड करण्यायोग्य फॉरमॅट स्टोरेज 3 BD-RE - HD व्हिडिओ रेकॉर्डिंग आणि PC डेटा स्टोरेजसाठी पुनर्लेखन करण्यायोग्य फॉरमॅट .

सध्या, सिंगल-लेयर डिस्क 25GB आणि ड्युअल-लेयर डिस्क 50GB धारण करू शकते. 50GB डिस्कवर 9 तासांपेक्षा जास्त हाय-डेफिनिशन (HD) व्हिडिओ. 50GB डिस्कवर सुमारे 23 तासांचा स्टॅंडर्ड डेफिनिशन (SD) व्हिडिओ बघता येतो.

## COPA - DOS कमांड लाइन इंटरफेस

## डिरेक्टरी लिस्ट साठी बेसिक DOS कमांड वापरा (Use basic DOS commands for directory listing)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- DOS मध्ये रेन्ज बद्ध (हायरार्कीकल) डिरेक्टरी सिस्टिम चे डिस्क्रिप्शन करा
- डिरेक्टरी आणि सबडिरेक्टरी तयार करण्यासाठी dos कमांड्स वापरा
- डिरेक्टरी बदलण्यासाठी आणि लिस्ट बद्ध करण्यासाठी dos कमांड्स वापरा
- स्पेसिफिक फाइल्समध्ये एक्सेस करण्यासाठी dos कमांड वापरा.

रेन्ज बद्ध (हायरार्कीकल) (हायरार्कीकल) डिरेक्टरी सिस्टिम :सोप्या भाषेत पदानुक्रम म्हणजे संस्था किंवा संस्थांची व्यवस्था. वस्तू, फायली, लोक, कल्पना किंवा इतर कोणतीही गोष्ट यासारखी कोणतीही वस्तू असू शकते.

व्यवस्था म्हणजे, उदाहरणार्थ, चलनाची संप्रदायाद्वारे व्यवस्था केली जाऊ शकते. खडे त्यांच्या आकारानुसार व्यवस्थित केले जाऊ शकतात.

पदानुक्रमानुसार संस्थांचे आयोजन करण्याचे इतर अनेक पाथ आहेत. परंतु, रेन्ज बद्ध (हायरार्कीकल) संस्था स्पेशल आहे कारण या व्यवस्थेद्वारे तुम्ही प्रत्येक एलिमेंटला इतर संस्थांशी असलेल्या संबंधानुसार नाव देऊ शकता.

DOS मध्ये, संस्था ही डिरेक्टरी सिस्टीममधील डिरेक्टरी असतात. येथे, पदानुक्रम आवश्यक कोर किंवा मूळ अस्तित्वापासून सुरू होतो. उदाहरणार्थ, कौटुंबिक वृक्षात, आपण महान-आजोबांना मानू शकतो जे आपल्या अस्तित्वाचे मूळ कारण होते. DOS मध्ये, या मुख्य एलिमेंटला root डिरेक्टरी म्हणून संबोधले जाते.

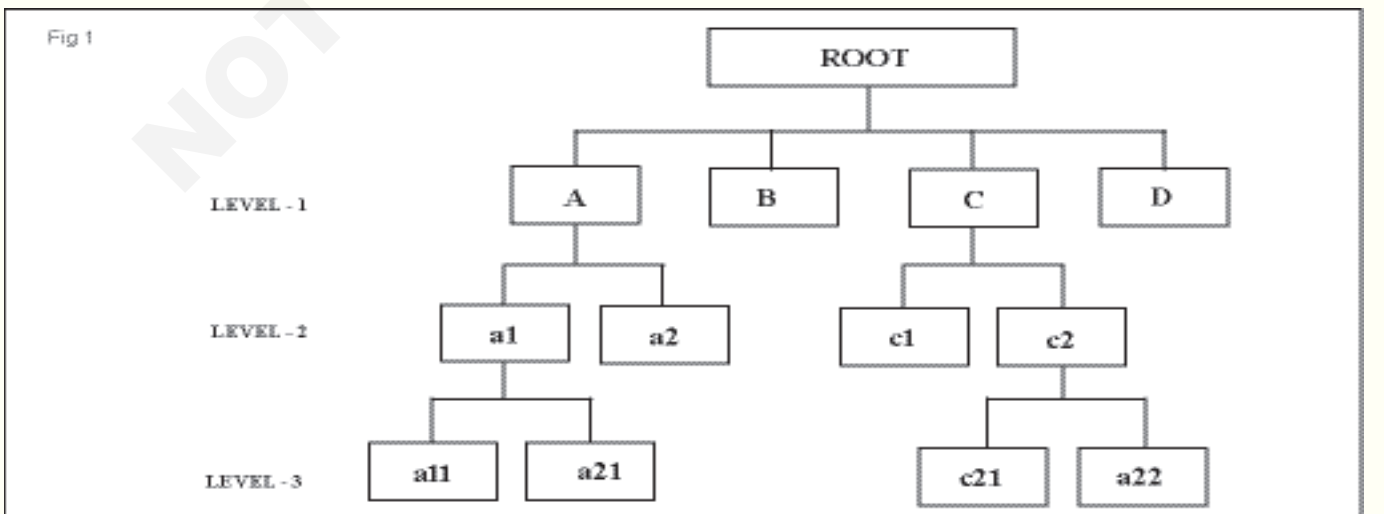
वरील उदाहरणाप्रमाणे, जर आपण ग्रेट-ग्रँड-ग्रँड फादरला मूळ डिरेक्टरी म्हणून विचारात घेतले तर, ग्रेट ग्रँड फादर, ग्रँड फादर, फादर हे सब डिरेक्टरी म्हणून संदर्भित केले जातात. तर root डिरेक्टरी इंटर्नल असलेल्या डिरेक्टरींना DOS मध्ये सबडिरेक्टरी म्हणतात. या सबडिरेक्टरी मूळ डिरेक्टरी कडे त्यांचे पाथ शोधू शकतात.

DOS रेन्ज बद्ध (हायरार्कीकल) फाइल सिस्टमला ट्री-स्ट्रक्चर्ड फाइल

सिस्टम म्हणतात. या झाडाच्या संरचनेच्या पायथ्याशी मूळ डिरेक्टरी आहे. एका वंशवृक्षात, गोविंदा हा राजप्याचा मुलगा आहे, जो रामप्याचा मुलगा आहे जो वेंकाप्याचा मुलगा आहे. वेंकाप्या हे गोविंदासाठी वंशवृक्षाचे प्रमुख किंवा मूळ आहे.

rootपासून अनेक डिरेक्टरी तयार करता येतात. root नंतर या प्रत्येक डिरेक्टरीचे मूळ असेल. तुम्ही सबडिरेक्टरी देखील तयार करू शकता जे इतर सबडिरेक्टरीजमधून स्टेम करतात जे इतर सबडिरेक्टरीजमधून उद्भवतात आणि याप्रमाणे. या नवीन सबडिरेक्टरीजमध्ये त्यांची मूळ डिरेक्टरी म्हणून एक सबडिरेक्टरी आहे. सबडिरेक्टरी rootपासून रेन्ज बद्ध (हायरार्कीकल) मेथड्स ने कशी मांडली जातात हे पिकचर 1 मध्ये स्पष्ट केले आहे. DOS डिरेक्टरी सिस्टीमला अनेकदा ट्री-स्ट्रक्चर्ड डिरेक्टरी सिस्टीम म्हणतात.

सबडिरेक्टरीजचे तीन स्तर आकृती 1 मध्ये दर्शविले आहेत. स्तरांची नंबर कितीही असली तरी सबडिरेक्टरीजचा संबंध महत्त्वाचा आहे. प्रत्येक सबडिरेक्टरी, तसेच root डिरेक्टरीमध्ये युजर फाइल्स असू शकतात. जोपर्यंत फायली वेगवेगळ्या डिरेक्टरीमध्ये राहतात तोपर्यंत दोन फाइल्समध्ये समान फाइल नाव आणि एक्सपान्शन असू शकतो. याचे कारण असे की, तुमची कमांड स्पेसिफाइड केलेल्या दोन समान-नावाच्या फाइल्सपैकी कोणती फाइल DOS ला माहित असणे आवश्यक आहे. यासाठी, DOS ला rootपासून सुरू होणाऱ्या डिरेक्टरींचे नाव आवश्यक आहे, जे इच्छित



फाइलकडे नेईल. फाइलकडे नेणाऱ्या डिरेक्टरी नावांच्या या क्रमाला पाथ (पाथ) म्हणतात

पाथ ही डिरेक्टरी नावांची साखळी आहे जी DOS ला तुम्हाला हवी असलेली फाइल कशी शोधायची हे सांगते. प्रत्येक डिरेक्टरी दुसऱ्यापासून '\ ' वर्णाने विभक्त केली जाते. या '\ ' ला DOS डिरेक्टरी डिलिमिटर म्हणून संबोधले जाते. ड्राइव्ह स्पेसिफायर (C: किंवा D: इ.,) सह फाइल्सचे पूर्ण पाथ नाव हे फाइल कोठे

आहे याचे परिपूर्ण सूचक आहे. ठराविक पाथ नोटेशन खाली दिले आहेत;

D:\Animals\Domestic\Pets\Dog.txt

C:\Admin\Accounts\Tours\Bata.txt

**पाथ आणि डिरेक्टरी रचनेच्या पुढील तपशीलांची चर्चा योग्य धड्यांमध्ये केली जाईल.**

### डॉस कमांड

1 MKDIR नवीन डिरेक्टरी बनवते किंवा तयार करते.

किंवा

MD

सिन्टेक्स

MKDIR:path\dirname

किंवा

MD:path\dirname

कुठे,

C: सबडिरेक्टरी साठी डिस्क ड्राइव्ह आहे

path\ डिरेक्टरीचा पाथ दर्शवितो ज्यामध्ये सबडिरेक्टरी तयार केली जात आहे. dirname हे सबडिरेक्टरीचे नाव आहे.

### स्विच करा

(काहीही नाही)

### महत्वाच्या नोट्स

- MKDIR किंवा त्याचे शॉर्ट फॉर्म MD निवडलेल्या root डिरेक्टरी इंटर्नल नवीन सबडिरेक्टरी बनवते.
- तुम्हाला पाहिजे तितक्या सबडिरेक्टरी तयार करणे शक्य आहे, परंतु लक्षात ठेवा: DOS पाथ नावासाठी बॅकलॅशसह 63 पेक्षा जास्त कॅरक्टर स्वीकारत नाही.
- सबडिरेक्टरीजचे बरेच स्तर आणि लांब नावे तयार करू नका.
- तुम्ही सध्याच्या डिरेक्टरीमधील फाइल नावासारखे डिरेक्टरी नाव तयार करू शकत नाही.

उदाहरणार्थ, जर तुमच्याकडे सध्याच्या डिरेक्टरी FLIES नावाची फाइल

असेल, तर तुम्ही या डिरेक्टरी FLIES नावाने सबडिरेक्टरी तयार करू शकत नाही. तथापि, जर फाईल FLIES ला एक्स्टेंशन दिले असेल तर FLIES.DOC , नंतर नावे विरोध करणार नाहीत आणि तुम्ही FLIES नावाने सबडिरेक्टरी तयार करू शकता.

उदाहरणे

सध्याच्या ड्राइव्ह इंटर्नल ड्रायव्हर्स नावाने सबडिरेक्टरी तयार करण्यासाठी, सूचना असेल,

MKDIR\Drivers

किंवा

MD\ Drivers

C:\Devices>MD \Printers

ही सूचना करंट ड्राइव्ह C: इंटर्नल प्रिंटर नावाने सबडिरेक्टरी तयार करते. लक्षात ठेवा की कमांड डिव्हाइसेस नावाच्या दुसऱ्या सबडिरेक्टरीतून जारी केली जात असली तरी, नवीन तयार केलेली सबडिरेक्टरी प्रिंटर डिव्हाइसेस डिरेक्टरी इंटर्नल तयार होत नाही तर थेट C:\ root इंटर्नल तयार केली जाते. DIR जारी करून याची पडताळणी केली जाऊ शकते डिरेक्टरी इंटर्नल सबडिरेक्टरी तयार करण्यासाठी डिव्हाइसेसची सूचना असेल,

**C:\Devices>MD Printers**

डिरेक्टरी तयार करण्याच्या खालील विविध प्रकारांवर चर्चा करा:

**C:\Devices\Printers>MD C:\Devices\Plotters**

### 2 CHIDR or CD

करंट डिरेक्टरी चा पाथ बदलतो किंवा दाखवतो.

सिन्टेक्स

CHIDRd: path

किंवा, शॉर्ट फॉर्म वापरून:

CD d: path

D:

पाथ वैध डिस्क ड्राइव्ह आणि डिरेक्टरी नावे आहेत.

स्विच

(none)

**तुमच्याकडे CD सह रेन्ज बद्ध (हायरार्कीकल) डिरेक्टरीद्वारे युक्ती चालवण्याच्या दोन मेथड्स आहेत: (1) rootपासून सुरू करणे, किंवा शीर्षस्थानी, डिस्कच्या डिरेक्टरी आणि खाली हलवणे, किंवा (2) करंट डिरेक्टरी पासून सुरू करणे आणि दोन्ही दिशेने हलवणे.**

डिस्कच्या root डिरेक्टरीपासून सुरू करण्यासाठी, तुम्ही पाथ कॅरक्टर (\), जसे की \ किंवा B:\ मध्ये सुरू केले पाहिजे. DOS जेव्हा \ हे पथातील पहिले कॅरक्टर म्हणून पाहते, तेव्हा सिस्टम root डिरेक्टरीने सुरू होते. अन्यथा, DOS करंट डिरेक्टरी सह सुरू होते.

ड्राइव्ह बदलणे:कॉम्प्युटरमध्ये मेमरी, हार्ड डिस्क तयार केलेली असेल आणि त्यात फ्लॉपी डिस्क, कॉम्पॅक्ट डिस्क इत्यादींवरील डेटा स्टोर /वाचण्याची तरतूद देखील असेल. प्रत्येक डिस्क सी ड्राइव्ह, ए ड्राइव्ह, बी ड्राइव्ह इत्यादी नावाने ओळखली जाते. सी ड्राइव्ह C द्वारे दर्शविले जाते: आणि A ड्राइव्हला A: आणि असेच दर्शविले जाते. खालील उदाहरणात दाखविल्याप्रमाणे DOS करंट किंवा डीफॉल्ट ड्राइव्हवरून इच्छित डिस्क ड्राइव्हचे अक्षर ओळख टाईप करून लगेच कोलनद्वारे बदलण्याची परवानगी देते:

C\> a:

ही कमांड C ड्राइव्हवरून D ड्राइव्ह वर कंट्रोल बदलण्याची सूचना देते

फ्लॉपी सबलब्ध नसल्यामुळे किंवा इतर कोणत्याही कारणामुळे डिस्क ड्राइव्हमध्ये एक्सेस न झाल्यास, DOS एक एरर संदेश प्रदर्शित करेल.

ड्राइव्ह A वाचण्यात एरर तयार नाही

निरस्त करा, पुन्हा प्रयत्न करा, अयशस्वी?

खाली परिभाषित केलेल्या A, R किंवा F की दाबणे आवश्यक आहे

A एरर साठी जबाबदार असलेली कमांड रद्द करण्यासाठी DOS ला निर्देश देते. हा पर्याय निवडल्यास DOS कमांड बंद करेल आणि प्रॉम्प्ट पुन्हा प्रदर्शित करेल.

R DOS ला त्रुटी निर्माण करणाऱ्या कमांडचा पुन्हा प्रयत्न करण्याचे निर्देश देते. बऱ्याच प्रकरणांमध्ये हा पर्याय त्रुटी निर्माण करणारी समस्या सुधारण्यासाठी निवडला जातो. (फ्लॉपी डिस्क घातली जाऊ शकत नाही).

F DOS ला त्रुटीकडे दुर्लक्ष करण्यास आणि प्रक्रिया सुरू ठेवण्याचा प्रयत्न करण्याचे निर्देश देते. काही प्रकरणांमध्ये जेव्हा DOS डिस्कचा एक पार्ट वाचतो तेव्हा त्यात त्रुटी असते.

## डॉस कमांड

**आपण** डिरेक्टरीमध्ये फाइल्स आणि सबडिरेक्टरीजची लिस्ट प्रदर्शित करते.

## सिन्टेक्स

DIR C:path/filename [/P] [/W] [/A[:attribs]] [/O[:sortord]]  
[S] [/B] [/L] [/C[H]]

कुठे,

- C: फाइल्स आणि सबडिरेक्टरीजची लिस्ट प्रदर्शित करण्यासाठी डिरेक्टरी धारण करणारी डिस्क ड्राइव्ह आहे
- पाथ / डिरेक्टरी आणि/किंवा लिस्ट साठी फाइल्स स्पेसिफाइड करते.
- फाइलनाम फाइल किंवा प्रदर्शित करण्यासाठी फाइल्सची लिस्ट स्पेसिफाइड करते, जर फाइलचे नाव स्पेसिफाइड केले नसेल तर डिरेक्टरी तील सर्व फाइल्स लिस्ट बद्ध केल्या जातील.
- [/P]/[W] ..... आउटपुट फॉर्मॅट करण्यासाठी स्विकेस स्पेसिफाइड करते.

## स्विच करा

/P इनफार्मेशन च्या प्रत्येक स्क्रीननंतर विराम देतो आणि कोणतीही की दाबण्याची प्रतीक्षा करतो. कोणतीही की दाबल्यावर दुसरी स्क्रीनफुल किंवा उर्वरित इनफार्मेशन प्रदर्शित होईल. कमांड DIR/पी आहे

/W फक्त फाइल नावे प्रदर्शित करण्यासाठी 80-स्तंभांचे विस्तृत फॉर्मॅट वापरते आणि फाइल आकार, date आणि वेळ याविषयी इनफार्मेशन प्रदर्शित केली जात नाही. कमांड DIR/डब्ल्यू आहे

/A स्पेसिफाइड गुणधर्मांसह (अॅट्रिब्यूट्स) फायली प्रदर्शित करते. प्रॉपर्टीज (अॅट्रिब्यूट्स)

D डिरेक्टरी

R केवळ-वाचनीय (रिडेबल) फायली

H लपलेल्या फाइल्स

S सिस्टम फाइल्स

A स्टोर करण्यासाठी तयार फाइल्स - सबसर्ग म्हणजे "Not"

/O यादी क्रमवारी (सॉर्टिंग)त फाइल्स करा.

नावानुसार N क्रमवारी (सॉर्टिंग) लावलेली (कॅरक्टर मालानुसार)

S आकारानुसार (सर्वात लहान प्रथम)

E एक्सटेन्शन नुसार (कॅरक्टर मालानुसार)

D date आणि वेळेनुसार (सर्वात आधी)

G ग्रुप डिरेक्टरीज प्रथम - उलट क्रमासाठी सबसर्ग

C कॉम्प्रेसन रेशोननुसार (सर्वात लहान प्रथम)

/S स्पेसिफाइड डिरेक्टरी आणि सर्व सबनिर्देशिकांमध्ये फाइल्स प्रदर्शित करते.

/B बेअर फॉर्मॅट वापरते (कोणतीही टाईटल इनफार्मेशन किंवा सारांश नाही).

/L लोअरकेस वापरते.

/ C[H] फाइल कॉम्प्रेसन रेशो प्रदर्शित करते; /CH होस्ट वाटप युनिट आकार वापरते.

## महत्वाच्या सूचना:

- वाइल्डकार्ड्स (\* आणि ?) वापरून तत्सम फाइल्स लिस्ट बद्ध केलेल्या डिरेक्टरी, जेथे (\*) Star आणि (?) प्रश्नचिन्हांना वाइल्ड कॅरक्टर किंवा वाइल्ड कार्ड म्हटले जाते. \* उर्वरित अक्षरे बदलू शकतात आणि ? कोणतेही एक कॅरक्टर बदलू शकते.
- जेव्हा DIR पॅरामीटर्स किंवा स्विकशिवाय वापरला जातो, तेव्हा ते डिस्क व्हॉल्यूम लेबल आणि अनुक्रमांक प्रदर्शित करते; प्रति ओळ एक डिरेक्टरी किंवा फाइलनाम, बाइट्समधील फाइल आकार आणि

## DOS कमांड वापरून फाइल्स आणि फोल्डर्स मॅनेज करा (Manage files and folders using DOS commands)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टेक्स्ट फाइलमधील टेक्स्ट प्रदर्शित करण्यासाठी DOS कमांड वापरा
- फायली कॉपी, नाव बदलणे, हटवणे आणि हटवणे रद्द करण्यासाठी DOS कमांड वापरा.

DOS कमांड्स

TYPE टेक्स्ट फाइलची कंटेंट प्रदर्शित करते.

### सिन्टेक्स

TYPE C: पाथ /फाइलनाव

कुठे,

- C: प्रदर्शित करण्यासाठी फाइल धरून ठेवणारा डिस्क ड्राइव्ह आहे.
- पाथ /प्रदर्शित करण्यासाठी फाइलचे पोजिशन स्पेसिफाइड करते.
- फाइलचे नाव प्रदर्शित करण्यासाठी फाइल स्पेसिफाइड करते.

### स्विच

(none)

### महत्वाच्या नोट्स:

- TYPE कमांड ASCII फाइलची कंटेंट इतर प्रोग्राम वापरल्याशिवाय प्रदर्शित करण्याचा एक क्लिक पाथ प्रदान करते. फाइल डिस्कवर ASCII (अमेरिकन स्टँडर्ड कोड फॉर इन्फॉर्मेशन इंटरचेंज) टेक्स्ट म्हणून स्टोर केली जाते. संगणक बायनरी (एक आणि शून्य) अक्षरे, नंबर आणि चिन्हांमध्ये अनुवादित करण्याचा स्टँडर्ड पाथ आहे. जर इनफॉर्मेशन ASCII फॉर्मॅटमध्ये साठवली गेली नसेल, तर TYPE कमांड वापरल्यास इनफॉर्मेशन अस्पष्ट दिसेल.
- कमांड जारी केल्यावर DOS स्पेसिफाइड केलेल्या ड्राइव्हमध्ये दिसेल, स्पेसिफाइड केलेल्या फाइलच्या नावापर्यंत पोहोचण्यासाठी मार्गावर जाईल. नंतर ते फक्त ASCII फॉर्मॅटचे अक्षर, नंबर आणि चिन्हांमध्ये भाषांतर करते आणि मॉनिटरवर प्रदर्शित करते. व्हिडिओ मॉनिटर फक्त 24 ओळींची इनफॉर्मेशन दाखवू शकतो. जर फाइलमध्ये 24 पेक्षा जास्त ओळी असतील तर सुरुवातीच्या ओळी दिसू शकत नाहीत कारण टाइप कमांड स्क्रीनवर सर्व इनफॉर्मेशन स्क्रोल करते. कंट्रोल + S की एकत्र दाबून स्क्रोलिंग कंट्रोल केले जाऊ शकते (कंट्रोल की दाबून ठेवल्यावर S की दाबा आणि दोन्ही की सोडल्यास कंट्रोल + S असे म्हणतात) मॉनिटरवर इनफॉर्मेशन चे स्क्रोलिंग थांबेल. स्क्रीनवरील कंटेंट पाहिल्यानंतर, उर्वरित कंटेंट स्क्रोल करण्यासाठी कोणतीही की दाबली जाऊ शकते कंटेंट पाहण्यासाठी फाइल स्क्रीन पृष्ठ स्क्रीन पृष्ठानुसार, अधिक कमांड देखील वापरली जाऊ शकते. जे स्क्रीन पृष्ठानंतर स्क्रीनवर इनफॉर्मेशन चे स्क्रोल करणे थांबवेल आणि स्क्रीन

पेजवर 24 ओळीवर एक प्रॉम्प्ट संदेश — अधिक — प्रदर्शित होईल. कोणतीही की दाबल्यानंतर दुसरे स्क्रीन पृष्ठ प्रदर्शित होईल. MORE हे फिल्टर आहे उदा. itis हा एक प्रोग्राम आहे जो स्टँडर्ड कॅरक्टर चा प्रवाह फाइलमधील स्टँडर्ड आउटपुट (मॉनिटर) स्क्रीन पेजवर स्क्रीन पृष्ठानुसार हाताळतो.

### उदाहरणे

1 C:\COPA\DOS\PRACT\_3>TEST1.txt टाइप करा

C:\COPA\DOS\PRACT\_3 हा TEST.txt फाइलचा पाथ आहे आणि TYPE ही DOS द्वारे कार्यान्वित केलेली कमांड आहे.

2 C:\>TYPE C:\COPA\DOS\PRACT\_3\>TEST1.txt

यामुळे उदाहरण 1 प्रमाणेच आउटपुट मिळते. C: (C drive) वरून काम करत असताना ही कमांड डिरेक्टरी न बदलता जारी केली जाऊ शकते.

3 C:\COPA\DOS\PRACT\_3>TYPE TEST1.txt | अधिक

याचा रिझल्ट देखील समान आउटपुटमध्ये होईल परंतु स्क्रीन पृष्ठानुसार स्क्रीन पृष्ठ प्रदर्शित होईल. कोणतीही की दाबल्यावर स्क्रीन पेज बदलता येते. MORE सोबत आणखी एक कॅरक्टर प्रीफिक्स आहे '|

### फाइलचे नाव बदलणे

RENAME हा कमांड फाइलचे नाव बदलण्याची परवानगी देतो.

किंवा

REN

### सिन्टेक्स

REN C: PATH\filename1.ex1 filename2.ex2कुठे,

- C: प्रदर्शित करण्यासाठी फाइल धरून ठेवणारा डिस्क ड्राइव्ह आहे.
- PATH/प्रदर्शित करण्यासाठी फाइलचे पोजिशन स्पेसिफाइड करते.
- filename1.ex1 ही फाइल पुनर्नामित करायची आहे
- filename2.ex2 हे नवीन फाइलनाव आहे

### महत्वाच्या नोट्स

- जर ड्राइव्ह स्पेसिफाइड नसेल तर करंट डिस्क ड्राइव्ह वापरली जाईल.



- जर पाथ स्पेसिफाइड केला नसेल तर करंट डिरेक्टरी वापरली जाईल
- फाईलचे नाव बदलण्यासाठी एक्स्टेंशनसह अचूक फाईल नाव दिले पाहिजे. - नवीन फाईलनावासाठी योग्य एक्स्टेंशन सह वैध फाईल नाव दिले पाहिजे.
- फाईल नावांमध्ये वाइल्ड कॅरक्टर ना परवानगी आहे ज्याद्वारे फाईल्सच्या आवश्यक गटाचे नाव बदलले जाऊ शकते.
- फक्त फाईलची नावे बदलली जातील आणि कंटेंट समान राहिल.
- डिरेक्टरीमध्ये आधीपासून अस्तित्वात असलेल्या नावावर फाईलचे नाव बदलण्याचा प्रयत्न केल्यास, DOS एक त्रुटी संदेश सूचित करते  
डुप्लिकेट फाईल नाव किंवा फाईल नॉट फाउंड
- जर इन्व्हीड फाईल नाव किंवा नवीन नाव दिलेले नसेल, तर DOS देखील एक एरर संदेश सूचित करते

### फाईल नावांसाठी नियम.

- फाईलच्या नावात 1 ते 8 कॅरक्टर असणे आवश्यक आहे.
- 1 ते 3 कॅरक्टर चा पर्यायी एक्सपान्शन
- एक्सपान्शन वापरला असल्यास नाव आणि एक्स्टेंशन चे नाव यांच्यातील कालावधी (.).
- A ते Z पर्यंतची सर्व अक्षरे (लोअर केस अक्षरे ऑटोमेटिक अपरकेसमध्ये हस्तांतरित केली जातात), 0 ते 9 नंबर आणि स्पेशल कॅरक्टर आणि सिम्बॉल्स \$ # & @ ! ^ ( ) \_ - { } ' ~ फाईल नावात परवानगी आहे.
- Esc, Del, किंवा space bar सारखी कण्ट्रोल कॅरक्टर फाईल नावामध्ये वापरली जाऊ शकत नाहीत.
- कॅरक्टर + = / [ ] ; : ? \* < > : परवानगी नाही.
- डिरेक्टरी तील प्रत्येक फाईलचे नाव यूनिक असणे आवश्यक आहे.

उदाहरणे:

- 1 C:\COPA\DOS\PRACT\_3\>REN TEST2.txt CHECKED.txt  
C:\COPA\DOS\PRACT\_3\ हा TEST2.txt फाईलचा ड्राइव्ह आणि पाथ आहे TEST2.txt हे नाव बदलायचे आहे.  
CHECKED.txt हे नवीन फाईलनाव आहे
- 2 : \COPA\DOS\PRACT\_3\>REN \*.pic \*.jpg पिक्चर एक्सपान्शन jpg एक्स्टेंशन फाईल्समध्ये बदलला जाईल.

### फायली कॉपी करणे:

कॉपी एक किंवा अधिक फायली दुसऱ्या स्थानावर कॉपी करते.

### सिन्टेक्स

कॉपी [/A | /B] सोर्स [/A | /B] [+ सोर्स [/A | /B] [+ ...]] [डेस्टिनेशन [/A | /B]] [/V] [/Y | /- Y] सोर्स कॉपी करायच्या फाईल किंवा फाईल्स स्पेसिफाइड करतो. डेस्टिनेशन नवीन फाईल(स) साठी डिरेक्टरी आणि/

किंवा फाईलनाव स्पेसिफाइड करते.

स्विचेस

- /A ASCII टेक्स्ट फाईल दर्शवते.
- /B बायनरी फाईल दर्शवते.
- /V नवीन फाईल्स योग्यरित्या लिहिल्या गेल्या आहेत याची पडताळणी करते.
- /Y तुम्हाला विद्यमान डेस्टिनेशन फाईल ओव्हरराईट करायची आहे याची कन्फर्म करण्यासाठी प्रॉम्प्टिंग करते.
- /-Y तुम्ही विद्यमान डेस्टिनेशन फाईल ओव्हरराईट करू इच्छित आहात याची कन्फर्म करण्यासाठी सूचित करते.

**इंस्ट्रक्टर प्रत्येक बाबतीत किमान दोन उदाहरणांसह साध्या स्विचेसची चर्चा करेल.**

**कॉपी कमांड स्विचच्या अधिक तपशीलांसाठी डॉसवरील कोणतेही ट्यूटोरियल किंवा हँड बुक पहा**

### महत्त्वाच्या सूचना:

- DOS कमांड कॉपी एक किंवा अधिक फायली डुप्लिकेट करू शकते. एकाच डिरेक्टरीमध्ये वेगवेगळ्या नावांनी किंवा एका डिरेक्टरीमधून दुसऱ्या डिरेक्टरीमध्ये एकाच नावाने किंवा वेगळ्या नावाने.
- जर ड्राइव्ह स्पेसिफाइड नसेल तर करंट डिस्क ड्राइव्ह वापरली जाईल.
- जर पाथ स्पेसिफाइड केला नसेल तर करंट डिरेक्टरी वापरली जाईल.
- फाईल कॉपी करण्यासाठी एक्स्टेंशनसह तंतोतंत फाईलचे नाव दिले पाहिजे
- नवीन कॉपी केलेल्या फाईलनावासाठी योग्य एक्स्टेंशनसह वैध फाईल नाव दिले पाहिजे
- फाईल नावांमध्ये वाइल्ड कॅरक्टर ना परवानगी आहे ज्याद्वारे फाईल्सचा आवश्यक ग्रुप कॉपी केला जाऊ शकतो
- कॉपी केल्यावर, दोन्ही सोर्स आणि टारगेट फाईल्समध्ये समान कंटेंट असेल. - कॉपी त्याच नावाने टारगेट फाईल ओवरराईडिंग करते
- कॉपी ही फाईल स्वतः कॉपी करू देणार नाही जी सोर्स आहे आणि टारगेट फाईल्स समान नसावीत
- संकलित करताना डेस्टिनेशन फाईल नाव स्पेसिफाइड केले नसल्यास प्रथम फाईल नाव डेस्टिनेशन नाव होईल. पहिल्या फाईलच्या नावानंतर B

### उदाहरण

1 C:\COPA\DOS\PRACT\_3\>COPY TEST2.txt

TRIAL.txt

वरील आदेशानुसार C:\COPA\DOS\PRACT\_3 डिरेक्टरी TEST2.txt फाइल TRIAL.txt फाइल म्हणून कॉपी केली जाईल. डिरेक्टरी लिस्ट बद्ध केल्यावर दोन्ही फाईल्समध्ये समान तपशील असतील आणि दोन्ही फाईलची कंटेंट पाहिल्यावर समान असेल. डॉस प्रॉम्प्ट कॉपी केल्यानंतर 1 फाइल कॉपी केली आहे

2 C:\COPA\DOS\PRACT\_3>COPY \*.bmp \*.pic

वरील आदेशानुसार C:\COPA\DOS\PRACT\_3 डिरेक्टरीमध्ये bmp एक्स्टेंशन फाइल असलेल्या सर्व फाइल्स त्याच डिरेक्टरीमध्ये pic एक्सपान्शन फाइल्स म्हणून कॉपी केल्या जातात. डॉस कॉपी करताना कॉपी केलेल्या फाइलचे नाव प्रॉम्प्ट करते आणि कॉपी पूर्ण केल्यानंतर कॉपी केलेल्या फाइल्सची नंबर सूचित करते.

3 C:\COPA\DOS\PRACT\_3>COPY \*.pic

C:\COPA\DOS\PRACT\_4\

C:\COPA\DOS\PRACT\_3 डिरेक्टरी तील पिक्चर एक्सटेंशन सह सर्व फायली त्याच नाव आणि एक्सटेंशन सह C:\COPA\DOS\PRACT\_4 डिरेक्टरी कॉपी केल्या जातात.

कॉपी कमांडनंतर \*\* वापरल्याने सर्व एक्सटेंशन सह सर्व फायली डेस्टिनेशन स्थानावर कॉपी केल्या जातात.

**एकत्र करणे कॉपी करा :** सोर्स फाईल्समध्ये + चा वापर करून मल्टिपल फाईल एकत्र करून एकच फाईल बनवता येते आणि याला संयोग म्हणतात

उदाहरण ४ C:\COPA\DOS\PRACT\_3\TEST2.txt + TRIAL.txt  
CONCAT1.txt

वरील आदेशाने TEST2.txt आणि TRIAL.txt एकत्र केले जातात आणि CONCAT1.txt फाइल तयार केली जाईल ज्यामध्ये पहिल्या दोन सोर्स फाईल्सची कंटेंट असेल.

**फाइल डिलेट करणे**

**DEL** या स्पेसिफाइड केलेल्या फाइल्स हटवते.

किंवा

**Erase**

**सिन्टेक्स**

DEL C:path/filename [/P]

ERASE C:path/filename [/P]

कुठे,

- C:डिलीट करण्याची फाइल धरून ठेवणारा डिस्क ड्राइव्ह आहे.
- path / हटवल्या जाणाऱ्या फाईलचे पोजिशन स्पेसिफाइड करते.
- फाइलनाम ही फाइल डिलेट करायची आहे

**स्विच करा**

/P स्पेसिफाइड फाइल हटवण्यापूर्वी कन्फर्म करणासाठी सूचित करते.

/P स्विच वापरणे

जर /P स्विच वापरला असेल तर, DEL फाईलचे नाव दाखवते आणि खालील फॉर्मॅटमध्ये संदेशासह सूचित करते:

**filename, delete (Y/N)?**

हटवण्याची कन्फर्म करण्यासाठी Y दाबा, हटवणे रद्द करण्यासाठी N दाबा आणि पुढील फाइलनाम प्रदर्शित करा (जर फाइल्सचा समूह स्पेसिफाइड केला असेल), किंवा DEL कमांड थांबवण्यासाठी CTRL+C दाबा

**महत्वाच्या नोट्स**

- जर ड्राइव्ह स्पेसिफाइड नसेल तर करंट डिस्क ड्राइव्ह वापरली जाईल
- जर पाथ स्पेसिफाइड केला नसेल तर करंट डिरेक्टरी वापरली जाईल
- फाइल हटवण्यासाठी एक्स्टेंशनसह अचूक फाईलचे नाव दिले पाहिजे
- फाइल नावांमध्ये वाइल्ड कॅरक्टर ना परवानगी आहे ज्याद्वारे फाइल्सचा आवश्यक ग्रुप हटविला जाऊ शकतो
- डिलीट केल्यावर, फाईल्सची नावे डिरेक्टरीमधून काढून टाकली जातात.
- डिरेक्टरीमधील सर्व फाईल्स del कमांड टाईप करून डिलीट केल्या जाऊ शकतात आणि त्यानंतर [drive:]path. वाइल्डकार्डचा वापर (\* आणि ?) एका वेळी एकापेक्षा जास्त फाइल हटवण्यासाठी देखील केला जाऊ शकतो. तथापि, अनावधानाने फायली हटवू नयेत म्हणून वाइल्डकार्डसचा वापर DEL कमांडसह सावधपणे केला पाहिजे.

सर्व फाईल्स डिलीट करण्यासाठी खालील कमांड दिलेली आहे.

del \*.\*

DEL खालील प्रॉम्प्ट प्रदर्शित करते:

डिरेक्टरी तील सर्व फायली हटविल्या जातात! तुम्हाला खात्री आहे (Y/N)?

**करंट डिरेक्टरी तील सर्व फायली हटवण्यासाठी Y आणि नंतर ENTER दाबा किंवा**

हटवणे रद्द करण्यासाठी N दाबा आणि नंतर ENTER दाबा.

- डिरेक्टरी del कमांडसह काढल्या जाऊ शकत नाहीत डिरेक्टरी काढून टाकण्यासाठी वेगळी कमांड सबलब्ध आहे.
- एकदा फाइल हटवल्यानंतर मेमरी स्पेस नवीन फाइलने व्यापली असल्यास ती पुनर्प्राप्त केली जाऊ शकत नाही. चुकून फाइल (ज्या) ताबडतोब हटवल्या गेल्यास, युटिलिटी कमांड वापरून ते पुनर्प्राप्त केले जाऊ शकते.
- डिस्क किंवा डिस्कटवरील हटविलेल्या फाइलने व्यापलेली जागा मोकळी केली जाते.
- डिलीट कमांड ऍक्टिव्ह करण्यासाठी एंटर की दाबण्यापूर्वी फाइलच्या नावांमध्ये टायपोग्राफिक त्रुटी तपासा.

उदाहरण

- 1 C:\COPA\DOS\PRACT\_3\>DEL TEST2.txt  
वरील आदेशाने TEST2.txt फाइल C:\COPA\DOS\PRACT\_3 डिरेक्ट्रीमधून हटवली जाईल. TEST2 डिरेक्टरी लिस्ट बद्ध केल्यावर, txt सबलब्ध होणार नाही.
- 2 C:\COPA\DOS\PRACT\_4\>DEL \*.txt  
C:\COPA\DOS\PRACT\_4 डिरेक्टरी तील वरील आदेशाने txt एक्सटेन्शन सह सर्व फाईल्स हटवल्या जातील.
- 3 C:\COPA\DOS\PRACT\_3\TEMP \>DEL \*.\*  
C:\COPA\DOS\ PRACT\_3\TEMP डिरेक्टरी तील कोणत्याही एक्सटेन्शन सह सर्व फायली हटवल्या जातील.

**डिलीट केलेल्या फायली पुनर्प्राप्त करणे:**

**UNDELETE प्रोटेक्शन सुविधा डिलीट करा**

**सिन्टेक्स**

```
UNDELET C:path/filename [/DT | /DS | /DOS]
UNDELET [/LIST | /ALL | /PURGE[DRIVE]] | /STATUS | /LOAD
| /UNLOAD /UNLOAD | /S[DRIVE] | /T[DRIVE]-entrys]]
```

**कुठे,**

- C:डिस्क ड्राईव्ह आहे ज्यामध्ये फाईल्स न हटवल्या जातील.
- path /हटवल्या जाणाऱ्या फाईलचे पोजिशन स्पेसिफाइड करते.
- फाइलनाव ही फाइल रद्द केली जाणार आहे

**स्विचेस**

/LIST पुनर्प्राप्त करण्यासाठी सबलब्ध हटवलेल्या फाईल्सची लिस्ट देते.

/ALL कन्फर्म करणाऱ्यासाठी सूचित न करता फायली पुनर्प्राप्त करते.

/DOS MS-DOS ने हटवलेल्या फाईल्स रिकव्हर करते.

/DT Delete Tracker द्वारे संरक्षित फाईल्स पुनर्प्राप्त करते.

/DS Delete Sentry द्वारे संरक्षित फाईल्स पुनर्प्राप्त करते.

/लोड लोड डिलीट करा संरक्षणासाठी मेमरीमध्ये अनडिलीट करा.

/UNLOAD अनलोड मेमरीमधून अनडिलीट.

/PURGE[ ड्राइव्ह] डिलीट सेंट्री डिरेक्टरीमधील सर्व फाईल्स पुस करते.

/STATUS प्रत्येक ड्राइव्हसाठी प्रभावी प्रोटेक्शन मेथड्स प्रदर्शित करा.

/S[ड्राइव्ह] संरक्षणाची डिलीट सेन्ट्री मेथड्स सक्षम करते.

/T[ड्राइव्ह]-[एंट्रीज] संरक्षणाची डिलीट करा ट्रॅकिंग मेथड्स सक्षम करते.

**महत्वाच्या नोट्स**

एकदा डिस्कवरून फाइल हटवली की ती पुन्हा मिळवणे शक्य होणार नाही. जरी UNDELETE कमांड हटविलेल्या फाईल्स पुनर्प्राप्त करू शकते, परंतु डिस्कवर इतर फाईल्स तयार केल्या गेल्या नाहीत किंवा बदलल्या गेल्या नाहीत तरच ते निश्चितपणे असे करू शकते. जर एखादी फाईल चुकून डिलीट झाली असेल आणि ती ठेवणे आवश्यक असेल तर, संगणकावरील इतर सर्व क्रियाकलाप थांबवा आणि फाइल पुनर्प्राप्त करण्यासाठी त्वरित UNDELETE कमांड वापरा.

**उदाहरण**

1 C:\COPA\DOS\PRACT\_3\>UNDELETE TEST2.txt

वरील आदेशाने TEST2.txt फाइल पुनर्प्राप्त होईल. लिस्ट बद्ध केल्यावर TEST2.txt फाइल C:\COPA\DOS\PRACT\_3 डिरेक्ट्री सबलब्ध होईल.

2 C:\COPA\DOS\PRACT\_4\TEMP\> UNDELETE

वरील आदेशाने अनेक फाईल्स रिकव्हर करता येतात. DOS प्रत्येक फाईल हटवण्याची कन्फर्म करेल आणि फाइलचे पहिले अक्षर टाइप करण्यास सांगेल. C:\COPA\DOS\PRACT\_4 directory डिलीट आणि लिस्ट रद्द केल्यानंतर, न हटवलेली फाइल नावे पाहिली जाऊ शकतात.

3 C:\COPA\DOS\PRACT\_4\TEMP\> UNDELETE /ALL

वरील आदेशाने अनेक फाईल्स रिकव्हर करता येतात. DOS प्रत्येक फाईल रद्द करणे रद्द करण्याच्या कन्फर्म साठी सूचित करणार नाही. रद्द करणे आणि लिस्ट केल्यानंतर C:\COPA\DOS\PRACT\_4 directory, रद्द न केलेली फाइल नावे पाहिली जाऊ शकतात.

## लिनक्स ऑपरेटिंग सिस्टमचा परिचय (Introduction to Linux operating system)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- लिनक्सचे ओवरविव्ह
- लिनक्सचे फ्युचर्स परिभाषित करा
- लिनक्सचे एप्लिकेशन एरिया स्पष्ट करा
- कर्नल बदल

### लिनक्सचे ओवरविव्ह

#### ऑपरेटिंग सिस्टम

कर्नलशी बोलू शकणारे ऍप्लिकेशन्स लिहिण्यासाठी डेव्हलपरना स्पेशल साधने (जसे की कंपाइलर आणि GNU मधील कमांड लाइन्स) आवश्यक असतात. एप्लिकेशन लिहिल्यानंतर आणि इंस्टाल केल्यानंतर कर्नलमध्ये एक्सेस करणे एक्सटर्नल एप्लिकेशन साठी सोपे करण्यासाठी त्यांना साधने आणि एप्लिकेशन देखील आवश्यक आहेत.

कर्नलसह एकत्रित साधनांचा हा सामूहिक संच, ऑपरेटिंग सिस्टम म्हणून ओळखला जातो. हा साधारणपणे संगणकाच्या सॉफ्टवेअरचा सर्वात खालचा स्तर असतो जो सरासरी यूजर द्वारे एक्सेस करता येतो. सामान्य वापरकर्ते जेव्हा कमांड लाइनमध्ये एक्सेस करतात तेव्हा ते ऑपरेटिंग सिस्टमवर जातात.

लिनक्स त्यांचे ऍप्लिकेशन लिहिण्यासाठी शक्तिशाली साधने प्रदान करते. डेव्हलपर इनव्हायरमेंट, एडिटर आणि कंपायलर हे विकसकांचा कोड घेण्यासाठी आणि कर्नलमध्ये एक्सेस करू शकतील आणि फंक्शन पूर्ण करू शकतील अशा गोष्टीमध्ये रूपांतरित करण्यासाठी डिझाइन केलेले आहेत.

कर्नल प्रमाणे, लिनक्स ऑपरेटिंग सिस्टम देखील मॉड्यूलर आहे. डेव्हलपर वापरकर्ते आणि विकासकांना स्पेसिफिक फंक्शन पूर्ण करण्यासाठी डिझाइन केलेल्या Linux च्या नवीन फ्लेवर सह प्रदान करण्यासाठी ऑपरेटिंग साधने निवडू शकतात.

### लिनक्सचा परिचय

Linux (उच्चारित Lih-nucks) ही UNIX सारखी ऑपरेटिंग सिस्टीम आहे जी अनेक वेगवेगळ्या संगणकांवर चालते. जरी बरेच लोक लिनक्सला ऑपरेटिंग सिस्टम म्हणून संबोधत असले आणि त्यात सॉफ्टवेअर समाविष्ट केले असले तरी, काटेकोरपणे बोलायचे तर, लिनक्स ही ऑपरेटिंग सिस्टम कर्नल आहे, जी सॉफ्टवेअरच्या डिस्ट्रीब्युशन सह येते.

लिनक्स प्रथम 1991 मध्ये त्याचे लेखक लिनस टोरवाल्ड्स यांनी हेलसिंकी विद्यापीठात प्रसिद्ध केले. तेव्हापासून आजूबाजूला प्रोग्रामर म्हणून त्याची लोकप्रियता प्रचंड वाढली आहे

एक विनाव्हॅल्यु ऑपरेटिंग सिस्टम तयार करणे, वैशिष्ट्ये जोडणे आणि समस्यांचे निराकरण करण्याचा त्यांचा प्रकल्प जगाने स्वीकारला.

लिनक्स आजच्या संगणक वापरकर्त्यांमध्ये लोकप्रिय आहे त्याच कारणास्तव UNIX ऑपरेटिंग सिस्टमच्या सुरुवातीच्या आवृत्त्यांनी 20 वर्षांपूर्वी चाहत्यांना मोहित केले होते. Linux हे पोर्टेबल आहे, याचा अर्थ तुम्हाला नेम-ब्रँड किंवा क्लोन पीसी, Apple Macintoshes, Sun workstations किंवा Digital Equipment Corporation Alpha वर चालणाऱ्या आवृत्त्या सापडतील.

आधारित संगणक. लिनक्समध्ये सोर्स कोड देखील येतो, त्यामुळे तुम्ही तुमच्या गरजेनुसार सॉफ्टवेअर बदलू किंवा कस्टमाइझ करू शकता. शेवटी, लिनक्स ही एक उत्तम ऑपरेटिंग सिस्टम आहे, जी युनिक्सच्या इतर आवृत्त्यांमधून स्वीकारलेल्या वैशिष्ट्यांनी समृद्ध आहे.

### लिनक्स कुठे आहे?

लिनक्सच्या सर्वात प्रसिद्ध गुणधर्मपैकी एक म्हणजे ते कुठे वापरले जाऊ शकते. Windows आणि OS X प्रामुख्याने वैयक्तिक संगणकीय उपकरणांवर जसे की डेस्कटॉप आणि लॅपटॉप संगणकांवर आढळतात. सिम्बियन सारख्या इतर ऑपरेटिंग सिस्टीम फोन आणि PDA सारख्या छोट्या उपकरणांवर आढळतात, तर प्रमुख शैक्षणिक आणि कॉर्पोरेट लॅबमध्ये आढळणारे मेनफ्रेम आणि सुपरकॉम्प्युटर AS/400 आणि Cray OS सारख्या स्पेशल ऑपरेटिंग सिस्टम वापरतात.

लिनक्स, ज्याचे अस्तित्व सर्व्हर OS म्हणून सुरू झाले आणि डेस्कटॉप OS म्हणून सबयुक्त झाले, ते या सर्व उपकरणांवर देखील वापरले जाऊ शकते. 'मनगटाच्या घड्याळांपासून ते सुपरकॉम्प्युटरपर्यंत, लिनक्सच्या क्षमतांचे लोकप्रिय डिस्क्रिप्शन आहे.

### लिनक्सचे भविष्य

लिनक्स आधीच विविध टाइप च्या उपकरणांवर यशस्वी आहे, परंतु अनेक तांत्रिक एरिया देखील आहेत जिथे लिनक्स पुढे जात आहे, जरी डेस्कटॉप आणि सर्व्हरचा विकास आज इतर कोणत्याही ऑपरेटिंग सिस्टमपेक्षा वेगाने वाढत आहे.

लॅपटॉप आणि नोटबुक संगणकांच्या BIOS सिस्टिम वर Linux इंस्टाल केले जात आहे, जे वापरकर्त्यांना त्यांचे डिव्हाइस काही सेकंदात चालू करण्यास सक्षम करेल, एक सुव्यवस्थित Linux वातावरण आणेल. या वातावरणात वेब ब्राउझर आणि ई-मेल क्लायंट सारखी इंटरनेट कनेक्टिव्हिटी साधने असतील, जे वापरकर्त्यांना त्यांच्या डिव्हाइसच्या प्रायमरी ऑपरेटिंग सिस्टममध्ये बूट न करता इंटरनेटवर कार्य करण्यास अनुमती देतात- जरी ती ऑपरेटिंग सिस्टम विंडोज असली तरीही.

त्याच वेळी, लिनक्स मोबाईल इंटरनेट उपकरणांवर (MIDs) दिसत आहे. यामध्ये एम्बेडेड उपकरणे जसे की स्मार्ट फोन आणि PDA, तसेच नेटबुक उपकरणे समाविष्ट आहेत लहान लॅपटॉप-टाइप मशीन जे त्यांच्या मोठ्या समकक्षांची मुख्य कार्यक्षमता एका लहान, अधिक ऊर्जा-कार्यक्षम पॅकेजमध्ये वैशिष्ट्यीकृत करतात.

लिनक्ससाठी क्लाउड कंप्युटिंगची वाढ नैसर्गिकरित्या योग्य आहे, जे आधीपासून इंटरनेटचे अनेक वेब सर्व्हर चालवते. लिनक्स अॅमेझॉनच्या A3 सारख्या क्लाउड सर्विस ना ऑनलाइन ॲप्लिकेशन्स आणि वापरकर्त्यांना इनफार्मेशन वितरीत करण्यासाठी उत्कृष्ट क्षमतेसह कार्य करण्यास सक्षम करते.

क्लाउड कॉम्प्युटिंगमध्ये लिनक्सच्या वाढीशी संबंधित हे सुपरकॉम्प्युटर्सवरील लिनक्सचे सुप्रसिद्ध यश आहे, उच्च-कार्यक्षमता संगणन (HPC) आणि उच्च-उपलब्धता (HA) क्षेत्रांमध्ये, जेथे फिजिकल शास्त्र आणि जैव अभियांत्रिकीमधील शैक्षणिक रिसर्च न आणि कंपनी वित्तीय आणि ऊर्जा उद्योगांना त्यांची उद्दिष्टे पूर्ण करण्यासाठी विश्वसनीय आणि स्केलेबल संगणकीय शक्तीची आवश्यकता असते.

इंटरनेटवरील अनेक लोकप्रिय वेब 2.0 सर्विस , जसे की Twitter, Linked In, YouTube आणि Google या सर्व त्यांची ऑपरेटिंग सिस्टम म्हणून Linux वर अवलंबून असतात. भविष्यात नवीन वेब सर्विस येत असल्याने, लिनक्स हे या नवीन तंत्रज्ञानाला चालना देणारे व्यासपीठ बनणार आहे.

### लिनक्स ऑपरेटिंग सिस्टमचा सध्याचा ॲप्लिकेशन

आज लिनक्स डेस्कटॉप मार्केटमध्ये सामील झाले आहे. लिनक्स डेव्हलपर्सनी सुरवातीला नेटवर्किंग आणि सर्विस वर लक्ष केंद्रित केले आणि ऑफिस ॲप्लिकेशन्स हा शेवटचा अडथळा होता. मायक्रोसॉफ्ट या मार्केटवर राज्य करत आहे हे मान्य करायला त्यांना आवडत नाही, त्यामुळे वर्कस्टेशन म्हणून लिनक्सला स्वीकाराई पर्याय बनवण्यासाठी, वर्ड प्रोसेसर सारखे सोपे यूजर इंटरफेस आणि एमएस कॅंपॅटिबल ऑफिस ॲप्लिकेशन प्रदान करण्यासाठी गेल्या काही वर्षांत भरपूर पर्याय सुरू झाले आहेत. , स्प्रेडशीट, प्रेजेंटेशन आणि सारखे. सर्व्हरच्या बाजूने, लिनक्स हे एक स्थिर आणि विश्वासार्ह व्यासपीठ म्हणून प्रसिद्ध आहे, जे Amazon, सुप्रसिद्ध ऑनलाइन बुकशॉप, यूएस पोस्ट ऑफिस, जर्मन सैन्य आणि इतर अनेक कंपन्यांसाठी डेटाबेस आणि व्यापार सर्विस प्रदान करते. स्पेशल तः इंटरनेट प्रदाते आणि इंटरनेट सर्विस प्रदात्यांनी फायरवॉल, प्रॉक्सी-आणि वेब सर्व्हर म्हणून लिनक्सची आवड वाढवली आहे आणि तुम्हाला प्रत्येक UNIX सिस्टम प्रशासकाच्या आवाक्यात एक लिनक्स बॉक्स मिळेल जो आरामदायक मॅनेजमेंट स्टेशनची प्रशंसा करतो. पोस्ट ऑफिसमध्ये, ते तंत्रिका केंद्र आहेत जे मेलला पाथ देतात आणि मोठ्या सर्व इंजिनमध्ये, क्लस्टर्सचा वापर इंटरनेट सर्व करण्यासाठी केला जातो. हजारो हेवी-ड्युटी नोकऱ्यांपैकी या फक्त काही आहेत ज्या लिनक्स जगभरात दररोज करत आहे. हे देखील लक्षात घेण्यासारखे आहे की आधुनिक लिनक्स केवळ वर्कस्टेशन्स, मिड- आणि हाय-एंड सर्व्हरवरच चालत नाही, तर PDA, मोबाईल, एम्बेडेड ॲप्लिकेशन्सचे शिपलोड आणि प्रायोगिक मनगटी घड्याळे सारख्या "गॅंजेट्स" वर देखील चालते. यामुळे हार्डवेअरची एवढी विस्तृत रेंज कव्हर करणारी लिनक्स ही जगातील एकमेव ऑपरेटिंग सिस्टम बनते.

### कोड

लिनक्स हे इतर ऑपरेटिंग सिस्टीमपेक्षा वेगळे आहे कारण त्याचा एकच मालक नाही. Torvalds अजूनही लिनक्स कर्नलच्या विकासाचे मॅनेजमेंट करतात, परंतु व्यावसायिक आणि प्राइवेट विकासक संपूर्ण लिनक्स ऑपरेटिंग सिस्टम बनवण्यासाठी इतर सॉफ्टवेअरचे योगदान देतात.

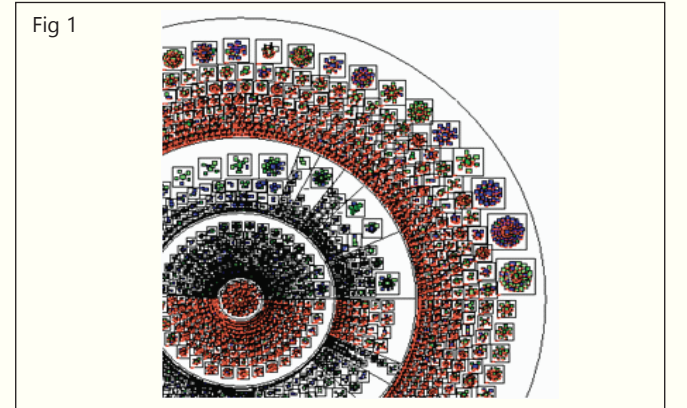
### कर्नल

सर्व ऑपरेटिंग सिस्टीममध्ये कर्नल असतात, जे आर्किटेक्चरल रूपकांच्या आसपास तयार केले जातात की डिव्हाइस हार्डवेअर निर्देशित करण्यासाठी निर्देशांचा एक मध्यवर्ती संच असणे आवश्यक आहे, कार्यक्षमतेच्या विविध मॉड्यूलर स्तरांनी वेढलेले आहे. लिनक्स कर्नल अद्वितीय आणि लवचिक आहे कारण ते निसर्गात देखील मॉड्यूलर आहे.

मॉड्यूलरिटी इष्ट आहे कारण ते विकसकांना कर्नलचे पार्ट शेड करण्यास अनुमती देते जे त्यांना वापरण्याची आवश्यकता नाही. सामान्यतः लहान कर्नल हा वेगवान कर्नल असतो, कारण तो आवश्यक नसलेल्या प्रक्रिया चालवत नाही.

जर एखाद्या डिव्हाइस डेव्हलपरला लिनक्सची आवृत्ती सेल फोनवर चालवायची असेल, तर तिला डिस्क ड्राइव्ह, इथरनेट डिव्हाइसेस किंवा मोठ्या मॉनिटर स्क्रीनशी संबंधित कर्नल कार्यक्षमतेची आवश्यकता नाही. फोनसाठी वापरण्यासाठी फक्त ऑप्टिमाइझ कर्नल सोडून ती ते तुकडे (आणि इतर) काढू शकते.

विंडो ऑपरेटिंग सिस्टीमचे कर्नल (ज्याला Microsoft च्या बाहेरील काही लोकांना विशेषाधिकार न भरता पाहण्याची परवानगी आहे) कोडचा एक घट्टपणे जोडलेला तुकडा आहे, जो सहजपणे तुकडे करता येत नाही. फोनवर बसण्यासाठी विंडोज कर्नल कमी करणे कठीण (अशक्य नसल्यास) आहे.



लिनक्सच्या यशासाठी ही मॉड्यूलरिटी महत्त्वपूर्ण आहे. स्पेसिफिक प्लॅटफॉर्मच्या गरजा पूर्ण करण्यासाठी स्केल कमी (किंवा वर) करण्याची क्षमता हा फक्त काही संभाव्य प्लॅटफॉर्मवर मर्यादित असलेल्या इतर ऑपरेटिंग सिस्टीमपेक्षा एक मोठा फायदा आहे.

मॉड्यूलरिटी स्थिरता आणि सुरक्षिततेवर देखील रिझल्ट करते. कर्नल कोडचा एक तुकडा अयशस्वी झाल्यास, उर्वरित कर्नल क्रॅश होणार नाही. त्याचप्रमाणे, कर्नलच्या एका भागावर (किंवा उर्वरित ऑपरेटिंग सिस्टम) बेकायदेशीर हल्ला कोडच्या त्या भागाला बाधा आणू शकतो, परंतु संपूर्ण इन्फॉर्मेशनच्या सुरक्षिततेची तडजोड करू नये.

## वातावरण (इनव्हअरमेंट)

विंडोज, मेनू आणि डायलॉग बॉक्स ज्यांना बहुतेक लोक ऑपरेटिंग सिस्टमचा पार्ट मानतात ते प्रत्यक्षात वेगळे स्तर असतात, ज्यांना विंडोइंग सिस्टम आणि डेस्कटॉप वातावरण म्हणतात.

हे स्तर मानव-केंद्रित ग्राफिकल यूजर इंटरफेस (GUI) प्रदान करतात जे वापरकर्त्यांना ऑपरेटिंग सिस्टममधील ॲप्लिकेशन्स आणि ऑपरेटिंग सिस्टमवर इंस्टाल केल्या जाणाऱ्या थर्ड पार्टी ॲप्लिकेशन्ससह सहजपणे कार्य करण्यास सक्षम करतात.

लिनक्समध्ये, विंडोइंग सिस्टम आणि डेस्कटॉप वातावरण वापरले जाऊ शकते अशा अनेक पर्याय आहेत, जे लिनक्स वापरकर्त्यांना ठरवू देते. हे Windows मध्ये केले जाऊ शकत नाही आणि OS X मध्ये करणे कठीण आहे.

ऑपरेटिंग सिस्टीम आणि कर्नल प्रमाणे, अशी साधने आणि कोड लायब्ररी सबलब्ध आहेत जी ॲप्लिकेशन डेव्हलपरना या वातावरणात अधिक सहजतेने कार्य करू देतात (उदा. GNOME साठी gtk+, KDE साठी Qt).

## ॲप्लिकेशन्स

ऑपरेटिंग सिस्टीममध्ये दोन टाइप चे ॲप्लिकेशन्स असतात: ते जे ऑपरेटिंग सिस्टीमचेच आवश्यक एलिमेंट आहेत आणि जे वापरकर्ते नंतर इंस्टॉल करतील. Windows आणि OS X सारख्या बंद ऑपरेटिंग सिस्टम वापरकर्त्यांना (किंवा विकसकांना) ते वापरू शकतील असे आवश्यक एलिमेंट ॲप्लिकेशन निवडू देत नाहीत. विंडोज डेव्हलपर्सनी मायक्रोसॉफ्टचे कंपाइलर, विंडोइंग सिस्टीम इत्यादी वापरणे आवश्यक आहे.

लिनक्स ॲप्लिकेशन डेव्हलपर्सकडे त्यांचे ॲप्लिकेशन विकसित करण्यासाठी निवडीचा मोठा संच असतो. हे ॲप्लिकेशन तयार करण्यासाठी अधिक लवचिकतेला अनुमती देते, परंतु याचा अर्थ विकासकाला कोणते Linux एलिमेंट वापरायचे हे ठरवावे लागेल.

## डिस्ट्रिब्यूशन

लिनक्स डिस्ट्रिब्यूशन हे लिनक्स कर्नलच्या शीर्षस्थानी (सामान्यतः ओपन सोर्स ) सॉफ्टवेअरचा संग्रह आहे. वितरण (किंवा लहान, डिस्ट्रो) सर्व्हर सॉफ्टवेअर, सिस्टम मॅनेजमेंट टूल्स, डॉक्युमेंटेशन आणि अनेक डेस्कटॉप ॲप्लिकेशन्स केंद्रीय सुरक्षित सॉफ्टवेअर रिपॉजिटरीमध्ये एकत्रित करू

शकतात. डिस्ट्रोचे उद्दिष्ट एक सामान्य फॉर्मेट आणि अनुभव, सुरक्षित आणि सोपे सॉफ्टवेअर मॅनेजमेंट आणि बऱ्याचदा स्पेसिफिक ऑपरेशनल उद्देश प्रदान करणे आहे.

## चला काही लोकप्रिय वितरणांवर एक नजर टाकूया.

**RED Hat:** रेड हॅट ही एक अब्ज डॉलरची व्यावसायिक लिनक्स कंपनी आहे जी लिनक्स विकसित करण्यासाठी खूप प्रयत्न करते. त्यांच्याकडे शेकडो लिनक्स स्पेशल ज्ञ आहेत आणि ते त्यांच्या उत्कृष्ट समर्थनासाठी ओळखले जातात. ते त्यांची उत्पादने (Red Hat Enterprise Linux आणि Fedora) मोफत देतात. Red Hat Enterprise Linux (RHEL) ची रीलिझ होण्यापूर्वी चांगली टेस्टिंग केली जाते आणि प्रकाशनानंतर सात वर्षांपर्यंत समर्थित असताना, Fedora हे वेगवान अद्यतनांसह परंतु समर्थनाशिवाय डिस्ट्रो आहे.

## उबंटू

कॅनोनिकलने 2004 मध्ये उबंटू लिनक्ससह मोफत कॉम्पॅक्ट डिस्क पाठवण्यास सुरुवात केली आणि घरगुती वापरकर्त्यांसाठी (बरेचजण मायक्रोसॉफ्ट विंडोज वरून स्विच करत) त्वरीत लोकप्रिय झाले. उबंटूला कधीही कमांड लाइन न पाहता ग्राफिकल लिनक्स डेस्कटॉप वापरण्यास सोपा असावा अशी कॅनोनिकलची इच्छा आहे. अर्थात त्यांना उबंटूला सपोर्ट विकूनही नफा मिळवायचा आहे.

**डेबियन:** डेबियनच्या मागे कोणतीही कंपनी नाही. त्याऐवजी हजारो सुसंघटित विकासक आहेत जे दर दोन वर्षांनी डेबियन प्रोजेक्ट लीडर निवडतात. डेबियनला सर्वात स्थिर लिनक्स वितरणांपैकी एक म्हणून पाहिले जाते. उबंटूच्या प्रत्येक रिलीझचा आधार देखील आहे. डेबियन तीन आवृत्त्यांमध्ये येते: स्थिर, टेस्टिंग आणि अस्थिर. प्रत्येक डेबियन रिलीजला टॉय स्टोरी पिक्चर पटातील पात्राचे नाव दिले जाते.

## इतर

Cent OS, Oracle Enterprise Linux आणि Scientific Linux सारखे वितरण Red Hat Enterprise Linux वर आधारित आहेत आणि अनेक समान तत्त्वे, डिरेक्टरी आणि सिस्टम एडमिनिस्ट्रेशन तंत्र सामायिक करतात. लिनक्स मिंट, एडुबंटू आणि इतर अनेक उबंटू नावाचे वितरण उबंटूवर आधारित आहेत आणि अशा प्रकारे डेबियनसह बरेच काही सामायिक करतात. इतर शेकडो लिनक्स वितरणे आहेत.

## ओपन सोर्स सॉफ्टवेअरचा परिचय (Introduction to Open source software)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- OSS ची व्याख्या
- OSS चा इतिहास
- ओपन सोर्स सबक्रम
- OSS ची उदाहरणे.

## व्याख्या

ओपन सोर्स सॉफ्टवेअर (OSS) हे सॉफ्टवेअर आहे जे त्याच्या सोर्स कोडसह वितरित केले जाते, ते वापरण्यासाठी, बदल करण्यासाठी आणि

त्याच्या मूळ अधिकारांसह डिस्ट्रीब्यूशन साठी सबलब्ध करून देते. सोर्स कोड हा सॉफ्टवेअरचा पार्ट आहे जो बहुतेक संगणक वापरकर्ते कधीही पाहत नाहीत; हा कोड आहे संगणक प्रोग्रामर प्रोग्राम किंवा ॲप्लिकेशन

कसे वागतात हे कंट्रोल करण्यासाठी हाताळतात. प्रोग्रामर ज्यांना सोर्स कोडमध्ये एक्सेस आहे ते प्रोग्राममध्ये जोडून, बदलून किंवा योग्यरित्या कार्य करत नसलेले पार्ट निश्चित करून बदलू शकतात. OSS मध्ये सामान्यतः एक परवाना समाविष्ट असतो जो प्रोग्रामरना त्यांच्या गरजा पूर्ण करण्यासाठी आणि सॉफ्टवेअरचे वितरण कसे करता येईल यावर कंट्रोल ठेवण्यासाठी सॉफ्टवेअरमध्ये सुधारणा करण्यास अनुमती देतो.

### **OSS चा इतिहास काय आहे?**

सोर्स कोड ओपन पणे सबलब्ध करून देण्याची कल्पना 1983 मध्ये एमआयटीमधील प्रोग्रामर रिचर्ड स्टॉलमन यांनी अनौपचारिकपणे स्थापन केलेल्या वैचारिक चळवळीतून उद्भवली. स्टॉलमनचा असा विश्वास होता की सॉफ्टवेअर प्रोग्रामरसाठी एक्सेस योग्य असावे जेणेकरून ते त्यांच्या इच्छेनुसार ते सुधारू शकतील, ते समजून घेणे, त्याबद्दल शिकणे आणि त्यात सुधारणा करणे या उद्देशाने. स्टॉलमनने त्याच्या स्वतःच्या परवान्याखाली विनाव्हॅल्यु कोड जारी करण्यास सुरुवात केली, ज्याला GNU सार्वजनिक परवाना म्हणतात. सॉफ्टवेअर निर्मितीच्या सभोवतालचा हा नवीन दृष्टिकोन आणि विचारधारा पकडली गेली आणि अखेरीस 1998 मध्ये ओपन सोर्स पुढाकाराची इन्स्टॉलेशन झाली.

### **ओपन सोर्स इनिशिएटिव्ह म्हणजे काय?**

ओपन सोर्स इनिशिएटिव्ह (OSI) ची निर्मिती ओपन सोर्स सॉफ्टवेअर आणि कम्युनिटीजचा प्रचार आणि प्रोटेक्शन करण्यासाठी करण्यात आली.ii थोडक्यात, ओएसआय हे ओपन सोर्स सॉफ्टवेअरचे केंद्रीय इनफार्मेशन आणि नियमन भांडार म्हणून काम करते. हे OSS कसे वापरावे आणि त्याच्याशी संवाद कसा साधावा यासाठी नियम आणि पाथ दर्शक तत्त्वे प्रदान करते, तसेच कोड परवाना इनफार्मेशन प्रदान करते.

ओपन सोर्सचा वापर आणि सबचार समजण्यायोग्य आणि नैतिक बनवण्यासाठी सपोर्ट , व्याख्या आणि सामान्य समुदाय कोलायब्रेशन .

### **OSS कसे कार्य करते?**

ओपन सोर्स कोड सहसा सार्वजनिक भांडारात (public repository) स्टोर केला जातो आणि सार्वजनिकरित्या सामायिक केला जातो. कोड स्वतंत्रपणे वापरण्यासाठी किंवा एकूण प्रकल्पाच्या डिझाइन आणि कार्यक्षमतेमध्ये सुधारणा करण्यासाठी कोणीही रेपॉजिटरीमध्ये एक्सेस करू शकतो. OSS

सहसा वितरण परवान्यासह येतो. या परवान्यात विकासक सॉफ्टवेअर कसे वापरू शकतात, त्याचा अभ्यास करू शकतात, त्यात सुधारणा करू शकतात आणि सर्वात महत्वाचे म्हणजे ते कसे वितरीत करू शकतात हे परिभाषित करतात. iii Synopsys Black Duck® Knowledgebase नुसार, पाच सर्वात लोकप्रिय परवाने आहेत

- MIT परवाना
- GNU जनरल पब्लिक लायसन्स (GPL) 2.0- हे अधिक प्रतिबंधात्मक आहे आणि त्यासाठी सुधारित कोडच्या प्रती सार्वजनिक वापरासाठी सबलब्ध करून दिल्या पाहिजेत.
- अपाचे परवाना 2.0
- GNU जनरल पब्लिक लायसन्स (GPL) 3.0
- BSD परवाना 2.0 (3-खंड, नवीन किंवा सुधारित)-हे कमी प्रतिबंधात्मक आहे

जेव्हा सोर्स कोड बदलला जातो, तेव्हा OSS मध्ये काय बदलले होते तसेच त्यात समाविष्ट असलेल्या मेथड्स समाविष्ट केल्या पाहिजेत. परवाना अटींवर अवलंबून, या सुधारणांमुळे तयार होणारे सॉफ्टवेअर विनाव्हॅल्यु सबलब्ध करून देण्याची आवश्यकता असू शकते किंवा नाही.

### **OSS ची काही उदाहरणे आहेत?**

- GNU/Linux
- Mozilla Firefox
- VLC मीडिया प्लेयर
- शुगर सीआरएम
- GIMP
- VNC
- Apache वेब सर्व्हर
- लिबरऑफिस
- jQuery

## आज्ञा आणि विविध एडिटर हाताळणे (Handling commands and various editors)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टर्मिनल बद्दल इनफार्मेशन
- कमांड शेल स्पष्ट करा
- लिनक्सच्या डिरेक्टरी सिन्टेक्स ची यादी करा
- लिनक्स कमांड्स परिभाषित करा
- linux OS च्या स्पेशल कॅरेक्टर ची यादी करा
- लिनक्स OS मधील विविध एडिटर चे स्पष्टीकरण द्या.

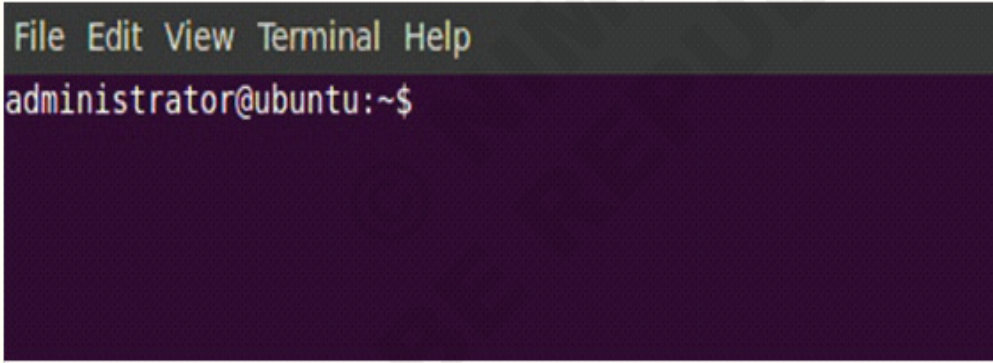
### टर्मिनल सुरू करणे

शेलमध्ये एक्सेस करण्यासाठी आम्ही शेल सारखी ऍप्लिकेशन वापरू, ज्याला टर्मिनल एमुलेटर देखील म्हणतात. GNOME किंवा KDE, किंवा Yakuake, Guake, rxvt आणि इतर मधील डीफॉल्ट ऍप्लिकेशन सह तेथे मोठ्या संख्येने चांगले टर्मिनल ऍप्लिकेशन आहेत. आता साठी फक्त डीफॉल्ट सह रहा तुमच्या सिस्टमसह येतो. तुम्ही GNOME वापरत

### काही सर्वात लोकप्रिय शेल आहेत:

- bash -बॉर्न-अगेन शेल, बहुतेक लिनक्स सिस्टम्सवरील डीफॉल्ट शेल.
- sh -बॉर्न शेल, एक जुने शेल जे आता इतके मोठ्या प्रमाणावर वापरले जात नाही.
- csh- 'C' शेल, जे 'C' प्रोग्रामिंग भाषेसारखे दिसणारे वाक्यरचना स्वीकारते.

Fig 1



असाल तर तुम्ही जाऊन टर्मिनलमध्ये एक्सेस करू शकता ऍप्लिकेशन्स -> अॅक्सेसरीज -> टर्मिनल किंवा Alt+F2 दाबा आणि दिसणाऱ्या रन बॉक्समध्ये gnome-terminal टाईप करा, त्यानंतर Enter. जर तुम्ही KDE चालवत असाल तर तुम्ही त्याऐवजी टाईप करू शकता कन्सोल Alt+F2 दाबल्यानंतर.

डिस्ट्रीब्युशन वर अवलंबून, प्रॉम्प्ट असे काहीतरी दिसू शकते user@host\$. अॅपरसँडच्या आधीचा पहिला पार्ट लॉगिन युजर नाव आहे आणि दुसरा पार्ट संगणकाचे होस्टनाव आहे.

**कमांड शेल:** एक कवच आहे कमांड इंटरप्रिटर जे तुम्हाला संगणकाशी संवाद साधण्याची परवानगी देते. गोष्टींची कार्य करण्याची मेथड्स अगदी सोपी आहे: तुम्ही कमांड टाईप करता, शेल त्यांचा अर्थ लावतो, त्याला सांगितलेली कामे पूर्ण करतो आणि शेवटी ते रिझल्ट स्टॅण्डर्ड आउटपुटवर पाठवते, जे सहसा स्क्रीन असते.

ही root डिरेक्ट्रीमधील फाइल्सची लिस्ट आहे. root डिरेक्टरी हे फाइल सिस्टम ट्री पदानुक्रमातील पहिले पोजिशन आहे, आणि ते स्लॅश कॅरेक्टरद्वारे दर्शविले जाते : /.

- tcsh -'C' शेलची सुधारित आवृत्ती.
- ksh -कॉर्न शेल, सुरुवातीला 1980 च्या दशकात विकसित झाले.
- dash- डेबियन आल्मक्रिस्ट शेल, डेबियन डिस्ट्रीब्युशन द्वारे तयार केलेले शेल.सिस्टममध्ये सबलब्ध शेलची लिस्ट

### \$ cat /etc/shells/

वरील कमांड आकृती 2 प्रमाणे खालील आउटपुट प्रदर्शित करेल.

या ट्यूटोरियलमध्ये आपण यावर लक्ष केंद्रित करू bash ,कारण ते सर्वात जास्त वापरले जाते आणि तिथल्या सर्वात शक्तिशाली शेलपैकी एक आहे. बॅश ही जुन्या बॉर्न शेलची आधुनिक अंमलबजावणी आहे (sh), GNU प्रकल्पाद्वारे विकसित केले गेले आहे, जे मोठ्या प्रमाणात साधने प्रदान करते आणि जे लिनक्स कर्नल, GNOME किंवा KDE सारखे डेस्कटॉप वातावरण आणि त्यांच्या वर चालणारे ऍप्लिकेशन , संपूर्ण लिनक्स प्लॅटफॉर्म समाविष्ट करतात. डेबियन किंवा उबंटू डिस्ट्रीब्युशन वर, सिस्टमद्वारे वापरलेले डीफॉल्ट शेल फाइलमध्ये स्पेसिफाइड केले आहे/ etc/passwd(डीफॉल्ट म्हणजे बॅश).



Fig 2

```

File Edit View Terminal Help
administrator@ubuntu:~$ cat /etc/shells
# /etc/shells: valid login shells
/bin/csh
/bin/sh
/usr/bin/es
/usr/bin/ksh
/bin/ksh
/usr/bin/rc
/usr/bin/tcsh
/bin/tcsh
/usr/bin/esh
/bin/dash
/bin/bash
/bin/rbash
/usr/bin/screen
administrator@ubuntu:~$

```

सिस्टममध्ये डीफॉल्ट शेल कसे प्रदर्शित करावे

`$ echo $SHELL`

टर्मिनलमध्ये खालील कमांड टाईप करा

आणि एंटर की दाबा. डीफॉल्ट शेल आकृती 3 प्रमाणे प्रदर्शित केले जाईल.

Fig 3

```

File Edit View Terminal Help
administrator@ubuntu:~$ echo $SHELL
/bin/bash
administrator@ubuntu:~$

```

लिनक्स डिरेक्टरी लेआउट

डिरेक्टरी	डिस्क्रिप्शन
	फाइल सिस्टमचा निनावी आधार. इतर सर्व डिरेक्टरी, फाइल्स, ड्राइव्हस् आणि डिव्हाइस या rootशी संलग्न आहेत. सामान्यतः (परंतु चुकीच्या मेथड्स ने) "स्लॅश" किंवा "/" डिरेक्टरी म्हणून संदर्भित. "/" ही फक्त एक डिरेक्टरी विभाजक आहे, स्वतःच डिरेक्टरी नाही.
/bin	अत्यावश्यक कमांड बायनरी (प्रोग्राम) येथे स्टोर आहेत (bash, ls, mount, tar, इ.)
/boot	बूट लोडरच्या स्थिर फाइल्स

/dev	डिवाइस फायली. लिनक्समध्ये, हार्डवेअर उपकरणे इतर फायलींप्रमाणेच अॅक्सेस केली जातात आणि ती या डिरेक्टरी खाली ठेवली जातात.
/etc	होस्ट-स्पेसिफिक सिस्टम कॉन्फिगरेशन फाइल्स.
/home	वापरकर्त्यांच्या वैयक्तिक होम निर्देशिकांचे पोजिशन (उदा. /home/Susan).
/lib	आवश्यक सामायिक लायब्ररी आणि कर्नल मॉड्यूल.
/proc	प्रक्रिया इनफार्मेशन स्यूडो-फाइल सिस्टिम . कर्नल डेटाचा इंटरफेस स्ट्रक्चर
/root	root (सुपर यूजर) होम डिरेक्टरी.
/sbin	आवश्यक सिस्टिम बायनरी (fdisk, fsck, init, इ.).
/tmp	तात्पुरत्या फाइल्स. सर्व वापरकर्त्यांना येथे तात्पुरत्या फाइल्स ठेवण्याची परवानगी आहे.
/usr	सर्वात शेअर करण्यायोग्य, केवळ-वाचनीय (रिडेबल) डेटासाठी (प्रोग्राम, लायब्ररी, दस्तऐवजीकरण आणि बरेच काही) बेस डिरेक्टरी.
/usr/bin	बहुतेक युजर प्रोग्राम येथे ठेवले जातात (cc, find, du, इ.).
/usr/include	सी प्रोग्राम्स संकलित करण्यासाठी हेडर फाइल्स
/usr/lib	बहुतेक बायनरी प्रोग्रामसाठी लायब्ररी
/usr/local	“लोकल रित्या” इंस्टाल फायली. ही डिरेक्टरी फक्त त्या वातावरणातच महत्त्वाची आहे जिथे फाइल्स नेटवर्कवर साठवल्या जातात. लोकल -इंस्टाल फाइल्स /usr/local/bin, /usr/local/lib, इ.) मध्ये जातात. स्त्रोतावरून इंस्टाल केलेल्या सॉफ्टवेअर पॅकेजेससाठी किंवा डिस्ट्रीब्युशन सह अधिकृतपणे पाठवलेले सॉफ्टवेअर देखील वापरले जाते.
/usr/sbin	अ-महत्वपूर्ण सिस्टिम बायनरी (lpd, useradd, इ.)
/usr/share	आर्किटेक्चर-स्वतंत्र डेटा (चिन्ह, बॅकग्राउंड, डॉक्युमेंटेशन, टर्मिनो, मॅन पेजेस इ.).
/usr/src	प्रोग्राम सोर्स कोड. उदा. लिनक्स कर्नल, सोर्स आरपीएम इ.
/usr/X11R6	एक्स विंडो सिस्टिम
/var	व्हेरिअबल डेटा: मेल आणि प्रिंटर स्पूल, लॉग फाइल्स, लॉक फाइल्स इ.

## लिनक्स कमांड्स काय आहेत?

लिनक्स कमांड एक्झिक्युटेबल बायनरी फाइल्स आहेत ज्या स्पेसिफिक फंक्शन करण्यासाठी रन केल्या जाऊ शकतात, उदाहरणार्थ संपूर्ण ग्राफिकल ॲप्लिकेशन चालवणाऱ्या डिरेक्टरीमध्ये फाइल्सची लिस्ट करणे. वारंवार वापरल्या जाणाऱ्या कमांडची उदाहरणे म्हणजे ls, cd, pwd, date किंवा cat. एक्झिक्युटेबल फाइल्सचा अपवाद वगळता, शेल बिल्ट-इन नावाची एक रेन्ज देखील आहे,

जे स्वतः शेलद्वारे प्रदान केलेल्या कमांड्स आहेत (आमच्या बाबतीत बॅश). आम्ही त्या नंतर हाताळू.

## लिनक्स कमांडचे सामान्य फॉर्मट आहे:

कमांड पर्याय(ले) फाइलनाव(ने) जे कमांड स्पेसिफाइड करते, त्यानंतर एक किंवा अधिक पॅरामीटर्स आणि वैकल्पिकरित्या त्यावर लागू करण्यासाठी एक किंवा अधिक फाइल्स. उदाहरणार्थ:

```
$ echo -e 'Hello, world!\n'
```

‘हॅलो, वर्ल्ड!’ असा टेक्स्ट आउटपुट करेल. त्यानंतर एक नवीन कॅरक्टर . -e पॅरामीटर (याला आर्ग्युमेंट किंवा स्विक देखील म्हणतात) echo कमांडला एस्केप केलेल्या कॅरक्टर चा अर्थ लावण्यासाठी सांगते, जसे की ट्रेलिंग \n, जे सिंगल कोट्समधील टेक्स्ट नंतर नवीन ओळ जोडेल. अग्रगण्य डॉलर चिन्हाकडे दुर्लक्ष करा, ते फक्त शेल प्रॉम्प्ट सूचित करते.

कमांड मध्ये आर्ग्युमेंट्स असू शकतात किंवा नसू शकतात. आर्ग्युमेंट्स हा पर्याय किंवा फाइलनाव असू शकतो.

## लिनक्स ऑपरेटिंग सिस्टिममधील स्पेशल कॅरक्टर

हे जाणून घेणे महत्त्वाचे आहे की अनेक सिम्बॉल्स आणि कॅरक्टर आहेत ज्यांचा शेल स्पेशल प्रकारे अर्थ लावतो. याचा अर्थ असा होतो की स्पेसिफिक टाइप केलेले कॅरक्टर : अ) स्पेसिफिक परिस्थितीमध्ये वापरले जाऊ शकत नाही, ब) स्पेशल ऑपरेशन्स करण्यासाठी वापरले जाऊ शकते किंवा, क) जर तुम्हाला ते सामान्य मार्गाने वापरायचे असतील तर ते “एस्केप केलेले” असणे आवश्यक आहे.

कॅरक्टर	डिस्क्रिप्शन
\	एस्केप कॅरक्टर . तुम्हाला एखाद्या स्पेसिफिक वर्णाचा रेफरेन्स घ्यायचा असल्यास, तुम्ही प्रथम बॅकस्लॅशसह "एस्केप" केले पाहिजे. उदाहरण: स्पर्श /tmp/filename\*
/	डिरेक्टरी सेपरेटर, डिरेक्टरी नावांची स्ट्रिंग विभक्त करण्यासाठी वापरला जातो. उदाहरण: /usr/src/linux
.	करंट डिरेक्टरी . फाइलनाममधील पहिले कॅरक्टर असताना फायली "लपवू" शकतात
..	पालक डिरेक्टरी
~	यूजर ची होम डिरेक्टरी
*	फाइलनामामध्ये 0 किंवा अधिक कॅरक्टर चे प्रतिनिधित्व करते, किंवा स्वतःच, डिरेक्टरी तील सर्व फाइल्स. उदाहरण: pic*2002 हे pic2002, picJanuary2002, picFeb292002 इत्यादी फाईल्सचे प्रतिनिधित्व करू शकते.
?	फाईल नावात एकच कॅरक्टर दर्शवतो. उदाहरण: hello?.txt हे hello1.txt, helloz.txt चे प्रतिनिधित्व करू शकते, परंतु hello22.txt नाही
[ ]	व्हॅल्यु च्या रेन्ज चे प्रतिनिधित्व करण्यासाठी वापरले जाऊ शकते, उदा. [0-9], [A-Z], इ. उदाहरण: hello[0-2].txt हे hello0.txt, hello1.txt आणि hello2.txt या नावांचे प्रतिनिधित्व करते.
	"पाईप". एका कमांडचे आउटपुट दुसऱ्या कमांडमध्ये पुनर्निर्देशित करा. उदाहरण: ls   अधिक
>	कमांडचे आउटपुट नवीन फाइलमध्ये पुनर्निर्देशित करा. फाइल आधीपासून अस्तित्वात असल्यास, ती ओव्हर-राइट करा. उदाहरण: ls > myfiles.txt
>>	विद्यमान फाइलच्या शेवटी कमांडचे आउटपुट पुनर्निर्देशित करा. उदाहरण: इको .मेरी 555-1234. >> phonenumbers.txt
<	प्रोग्राममध्ये इनपुट म्हणून फाइल पुनर्निर्देशित करा. उदाहरण: अधिक < phonenumbers.txt
;	कमांड सेपरेटर. तुम्हाला एकाच ओळीवर मल्टिपल कमांड कार्यान्वित करण्याची अनुमती देते. उदाहरण: cd /var/log ; कमी संदेश

## सीडी कमांड

लिनक्स आणि इतर युनिक्स सारखी ऑपरेटिंग सिस्टीममध्ये सध्याची डिरेक्टरी (म्हणजेच युजर सध्या काम करत असलेली डिरेक्टरी) बदलण्यासाठी cd कमांडचा वापर केला जातो. हे MS-DOS मधील CD आणि CHDIR कमांड सारखेच आहे.

### cd कमांड

#### cd [option] [directory]

चौरस कंसातील आयटम ऐच्छिक आहेत. कोणत्याही डिरेक्टरी चे नाव स्पेसिफाइड न करता वापरल्यास, cd यूजर ला मागील करंट डिरेक्टरी परत करते. हे दोन डिरेक्टरी दरम्यान टॉगल करण्याचे सोयीस्कर माध्यम प्रदान करते.

जेव्हा डिरेक्टरीचे नाव दिले जाते, तेव्हा cd करंट डिरेक्टरी बदल करते. नाव अब्सोल्युट पाथ नाव (म्हणजे मूळ डिरेक्टरी शी संबंधित पोजिशन ) किंवा लोकल पाथ नाव (म्हणजे करंट डिरेक्टरी शी संबंधित पोजिशन ) म्हणून व्यक्त केले जाऊ शकते. करंट डिरेक्टरी च्या सबडिरेक्टरीमध्ये बदलताना लोकल पाथ नाव वापरणे अधिक सोयीचे असते.

एक उदाहरण म्हणून, खालील करंट डिरेक्टरी बदल करेल, ती सिस्टमवर कोठे आहे याची पर्वा न करता (कारण तो एक अब्सोल्युट पाथ आहे), root डिरेक्टरीमध्ये (ज्याला फॉरवर्ड स्लॅश द्वारे दर्शविले जाते):

### cd /

त्याचप्रमाणे, खालील करंट डिरेक्टरी, तिचे पोजिशन काहीही असले तरी, /usr/sbin डिरेक्टरीमध्ये बदलेल (ज्यात सिस्टम प्रशासकाद्वारे वापरल्या जाणाऱ्या अ-महत्वाच्या सिस्टम युटिलिटीज असतात):

#### cd/usr/sbin

सध्या डिरेक्टरीमध्ये यूजर ला /usr/local/share/man/ डिरेक्टरीमध्ये बदलायचे असल्यास /usr/local/share/man/ man2, जी सध्याच्या डिरेक्टरीची सबडिरेक्टरी आहे, वापरून बदलणे शक्य होईल. परिपूर्ण पाथ नाव, म्हणजे,

#### cd/usr/local/share/man/man2

तथापि, रिलेटिव्ह पाथ नाव वापरणे स्पष्टपणे कमी कंटाळवाणे असेल, म्हणजे,

#### cdman2

युनिक्स सारख्या ऑपरेटिंग सिस्टीमवर सध्याची डिरेक्टरी सिंगलडॉट द्वारे दर्शविली जाते आणि तिची मूळ डिरेक्टरी (म्हणजे ती असलेली डिरेक्टरी) सलग दोन बिंदूनी दर्शविली जाते. अशा प्रकारे, खालील वापरून करंट डिरेक्टरी च्या मूळमध्ये बदलणे शक्य आहे (आणि बरेचदा सोयीस्कर आहे):

cd ..

cd चे आणखी एक सोयीस्कर वैशिष्ट्य म्हणजे कोणत्याही यूजर साठी केवळ टिल्डचा आर्ग्युमेंट्स म्हणून थेट त्याच्या होम डिरेक्टरीवर परत येण्याची क्षमता. होम डिरेक्टरी, ज्याला लॉगिन डिरेक्टरी देखील म्हटले जाते, ही युनिक्स सारखी ऑपरेटिंग सिस्टमवरील डिरेक्टरी आहे जी यूजर च्या वैयक्तिक फाइल्स, डिरेक्टरी आणि प्रोग्राम्ससाठी भांडार म्हणून काम करते. सिस्टीममध्ये लॉग इन केल्यानंतर युजर प्रथम इन केलेला डिरेक्टरी देखील आहे. टिल्ड हा एक लहान, लहरी, आडवा रेषेचा कॅरक्टर आहे जो करंट यूजर च्या होम डिरेक्टरीचे प्रतिनिधित्व करतो. म्हणजेच, कोणताही युजर खालील टाइप करून आणि नंतर एंटर की दाबून त्याच्या होम डिरेक्टरीमध्ये त्वरित परत येऊ शकतो:

cd ~

यूजर च्या होम डिरेक्टरीचे पूर्ण नाव टाइप करण्यापेक्षा हे सोपे आहे, उदाहरणार्थ, josephine नावाच्या यूजर च्या बाबतीत /home/josephine. (आणि युनिक्स सारख्या ऑपरेटिंग सिस्टीमवर कमांड लाइन वापरण्यास सोपी बनवणाऱ्या असंख्य शॉर्टकटपैकी हा फक्त एक आहे.)

त्यानंतर स्पेस आणि नंतर हायफन आल्यावर, cd दोन्ही यूजर ला मागील करंट डिरेक्टरी परत करते आणि नवीन ओळीवर त्या डिरेक्टरी च्या परिपूर्ण पाथ नावाचा रिपोर्ट देते. हे cd ची आधीच सोयीस्कर टॉगल क्षमता आणखी वाढवू शकते. टॉगल करणे स्पेशल तः सोयीचे असते जेव्हा दोन डिरेक्टरीपैकी किमान एक लांब संपूर्ण पाथ नाव असते, जसे की /usr/local/share/man/man2.

cd मध्ये फक्त दोन पर्याय आहेत आणि त्यापैकी एकही सामान्यपणे वापरला जात नाही. -P पर्याय cd ला प्रतिकात्मक दुव्यांचे अनुसरण करण्याऐवजी फिजिकल डिरेक्टरी स्ट्रक्चर वापरण्याची सूचना देतो. -L पर्याय प्रतिकात्मक दुवे फॉलो करण्यास पार्ट पाडतो.

### pwd कमांड

pwd कमांड सध्याच्या डिरेक्टरीचा संपूर्ण पाथ दाखवते.

सध्याची डिरेक्टरी म्हणजे कमांड लाइन इंटरफेस वापरताना युजर सध्या कार्यरत असलेली डिरेक्टरी आहे. कमांड लाइन इंटरफेस हा सर्व-टेक्स्ट डिस्प्ले मोड आहे आणि तो कन्सोलद्वारे प्रदान केला जातो (म्हणजे, संपूर्ण स्क्रीन फक्त टेक्स्ट असतो) किंवा टर्मिनल विंडोद्वारे (म्हणजे, GUI मधील केवळ टेक्स्ट विंडो).

संपूर्ण पाथ, ज्याला अब्सोल्युट पाथ देखील म्हणतात, डिरेक्टरी किंवा फाइलचा root डिरेक्टरीपासून त्या डिरेक्टरी किंवा फाइलपर्यंत आणि त्यासह डिरेक्टरीचा संपूर्ण पदानुक्रम असतो.

root डिरेक्टरी, जी फॉरवर्ड स्लॅश (/) द्वारे नियुक्त केली जाते, ही फाइल सिस्टमवरील बेस डिरेक्टरी आहे (म्हणजे, डिरेक्टरीचा पदानुक्रम), आणि त्यामध्ये सिस्टमवरील इतर सर्व डिरेक्टरी, सबडिरेक्टरीज आणि फाइल्स असतात. अशा प्रकारे, कोणत्याही डिरेक्टरी किंवा फाइलचा पूर्ण पाथ नेहमी फॉरवर्ड स्लॅशने सुरू होतो.

pwd ही लिनक्स आणि इतर युनिक्स सारखी ऑपरेटिंग सिस्टीममधील सर्वात बेसिक कमांड्सपैकी एक आहे. Is सोबत, जी करंट डिरेक्टरी तील कंटेंट लिस्ट बद्ध करण्यासाठी वापरली जाते आणि cd, जी करंट डिरेक्टरी बदलण्यासाठी वापरली जाते.

### pwd चे syntax

#### pwd [option]

बऱ्याच कमांड्सच्या विपरीत, pwd जवळजवळ नेहमीच स्वतःच वापरला जातो, म्हणजे,

#### Pwd

म्हणजेच, ते क्वचितच त्याच्या पर्यायांसह वापरले जाते आणि वितर्कांसह कधीही वापरले जात नाही (म्हणजे, फाइल नावे किंवा इनपुट म्हणून प्रदान केलेली इतर इनफार्मेशन). pwd नंतर समान ओळीवर टाइप केलेली कोणतीही गोष्ट, एक पर्याय वगळता, दुर्लक्ष केले जाते आणि कोणतेही त्रुटी संदेश परत केले जात नाहीत.

उदाहरण म्हणून, जर युजर नाव janis असलेला युजर त्याच्या होम डिरेक्टरीमध्ये असेल, तर वरील कमांड सामान्यतः /home/janis/ परत करेल (कारण, डिफॉल्टनुसार, सर्व होम डिरेक्टरी /home डिरेक्टरी असतात). त्याचप्रमाणे, जर युजर सध्या डिरेक्टरी /usr/share/ config मध्ये काम करत असेल (ज्यात अनेक प्रोग्राम कॉन्फिगरेशन फाइल्स आहेत), तर तीच कमांड /usr/share/ config परत करेल.

pwd हे कन्फर्म करण्यासाठी सबयुक्त आहे की सध्याची डिरेक्टरी सीडी वापरल्यानंतर यूजर च्या हेतूनुसार बदलली गेली आहे. उदाहरणार्थ, करंट डिरेक्टरी /home/janis वरून /usr/share/ config मध्ये बदलण्यासाठी cd कमांड जारी केल्यानंतर, pwd चा वापर कन्फर्म करणाऱ्यासाठी केला जाऊ शकतो; म्हणजेच, खालील कमांड चा क्रम जारी केला जाईल:

#### cd/usr/share/config/

#### pwd

pwd च्या स्टॅण्डर्ड आवृत्तीमध्ये फक्त दोन पर्याय आहेत, जे दोन्ही फक्त क्वचितच वापरले जातात. --help पर्याय खालीलप्रमाणे वापरला जातो:

```
pwd --help
```

हा पर्याय pwd बद्दल इनफार्मेशन प्रदर्शित करतो, ज्यामध्ये फारच कमी आहे कारण ती इतकी सोपी कमांड आहे (म्हणजे, त्यात फक्त दोन पर्याय आहेत आणि कोणतेही आर्ग्युमेंट्स स्वीकारत नाहीत).

दुसरा पर्याय --version आहे, जो आवृत्ती क्रमांक प्रदर्शित करतो, म्हणजे,

#### pwd --version

जरी हे बऱ्याचदा सध्याच्या कार्यरत डिरेक्टरी साठी उभे आहे असे मानले जात असले तरी, pwd हे प्रिंट वर्किंग डिरेक्टरीचे संक्षिप्त रूप आहे. प्रिंट हा शब्द लिहिण्यासाठी किंवा प्रदर्शनासाठी पारंपारिक UNIX शब्दावली आहे आणि जेव्हा संगणक आउटपुट सामान्यतः डीफॉल्टनुसार कागदावर प्रिंटेड केले जाते तेव्हा ते उद्भवले कारण CRT (कॅथोड रे ट्यूब) डिस्प्ले मॉनिटर्स अद्याप मोठ्या प्रमाणावर सबलब्ध नव्हते.

## इको कमांड

इको ही बॅश आणि सी शेल्समधील अंगभूत कमांड आहे जी त्याचे आर्ग्युमेंट्स स्टॅण्डर्ड आउटपुटवर लिहिते.

शेल हा एक प्रोग्राम आहे जो लिनक्स आणि इतर युनिक्सलाइक ऑपरेटिंग सिस्टमवर कमांड लाइन (म्हणजे सर्व-टेक्स्ट डिस्प्ले यूजर इंटरफेस) प्रदान करतो. ते त्यामध्ये टाइप केलेल्या कमांड्स देखील कार्यान्वित करते (म्हणजे, चालते) आणि रिझल्ट प्रदर्शित करते. Linux वर bash हे डीफॉल्ट शेल आहे.

कमांड म्हणजे संगणकाला काहीतरी करायला सांगणारी सूचना. आर्ग्युमेंट्स म्हणजे कमांडसाठी इनपुट डेटा. स्टॅण्डर्ड आउटपुट डिफॉल्टनुसार डिस्प्ले स्क्रीन आहे, परंतु ती फाइल, प्रिंटर इ. वर पुनर्निर्दिशित केली जाऊ शकते.

इको साठी वाक्यरचना आहे

**\$ echo \$USER**

**\$ इको "हॅलो वर्ल्ड"**

चौरसकंसातील आयटम ऐच्छिक आहेत. स्ट्रिंग म्हणजे कॅरक्टर चा कोणताही मर्यादित क्रम (म्हणजे अक्षरे, अंक, सिम्बॉल्स आणि विरामसिम्बॉल्स).

कोणत्याही पर्यायाशिवाय किंवा स्ट्रिंगशिवाय वापरल्यास, इको डिस्प्ले स्क्रीनवर एक रिक्त ओळ दर्शवते आणि त्यानंतरच्या ओळीवर कमांड प्रॉम्प्ट येते. याचे कारण असे की ENTER की दाबणे हा सिस्टीमला नवीन ओळ सुरू करण्याचा सिग्नल आहे आणि अशा प्रकारे इको या सिग्नलची पुनरावृत्ती करतो.

जेव्हा एक किंवा अधिक स्ट्रिंग आर्ग्युमेंट्स म्हणून प्रदान केल्या जातात, तेव्हा डीफॉल्टनुसार echo स्क्रीनवर त्या स्ट्रिंगची पुनरावृत्ती करते. अशा प्रकारे, उदाहरणार्थ, खालील टाईप केल्याने आणि ENTER की दाबल्याने इको हा पेन आहे या वाक्यांशाची पुनरावृत्ती करेल. पडद्यावर:

**echo This is a pen.**

कोट्ससह स्ट्रिंग वेढणे आवश्यक नाही, कारण स्क्रीनवर काय लिहिले आहे यावर त्याचा रिझल्ट होत नाही. अवतरण (एकतर किंवा दुहेरी) वापरले असल्यास, ते स्क्रीनवर पुनरावृत्ती होत नाहीत.

सुदैवाने, इको केवळ शब्दशः पुनरावृत्ती करण्यापेक्षा बरेच काही करू शकते जे त्याचे अनुसरण करते. म्हणजेच, जर व्हेरिअबलच्या नावाच्या आधी थेट (म्हणजेच मध्यस्थी न करता) डॉलर कॅरक्टर (\$) असेल तर ते स्पेसिफिक व्हेरिअबलचे व्हॅल्यू देखील दर्शवू शकते, जे शेलला व्हेरिअबलचे व्हॅल्यू बदलण्यास सांगते. त्याचे नाव.

उदाहरणार्थ, x नावाचे व्हेरिअबल तयार केले जाऊ शकते आणि त्याचे व्हॅल्यू खालील कमांडसह 5 वर सेट केले जाऊ शकते:

x = 5

x चे व्हॅल्यू नंतर खालील द्वारे परत बोलावले जाऊ शकते:

echo The number is \$x

environmental व्हेरिअबलची व्हॅल्यू दाखवण्यासाठी इको स्पेशल तः सबयुक्त आहे, जे शेलला कमांड लाइनवर किंवा स्क्रिप्टमध्ये (लहान प्रोग्राम्स) काम करत असताना कसे वागायचे ते सांगते.

उदाहरणार्थ, HOME चे व्हॅल्यू पाहण्यासाठी, environmental व्हॅल्यू जे कॅरंट यूजर ची होम डिरिक्टरी दर्शवते, खालील वापरले जाईल:

**echo \$HOME**

त्याचप्रमाणे, यूजर चे PATH पर्यावरणीय व्हेरिअबल दर्शविण्यासाठी इकोचा वापर केला जाऊ शकतो, ज्यामध्ये यूजर ने जारी केलेल्या आदेशाशी संबंधित एक्झिक्युटेबल प्रोग्राम सर्च प्यासाठी सिस्टम सर्च त असलेल्या निर्देशिकांची कोलन-विभक्त लिस्ट असते:

**echo \$PATH**

echo, डीफॉल्टनुसार, नवीन लाइन कॅरक्टरसह कोणतेही आउटपुट फॉलो करते. हे एक नॉन-प्रिंटिंग (म्हणजे, अव्ह.) कॅरक्टर आहे जे टेक्स्ट च्या एका ओळीचा शेवट आणि पुढील स्टार्ट दर्शवते. हे युनिक्स सारख्या ऑपरेटिंग सिस्टीममध्ये \n द्वारे दर्शविले जाते. रिझल्ट असा होतो की त्यानंतरचे कमांड प्रॉम्प्ट echoद्वारे परत आलेल्या आउटपुटच्या समान ओळीवर न जाता नवीन ओळीवर सुरू होते.

-e पर्यायाचा वापर नवीन ओळीच्या कॅरक्टर च्या अतिरिक्त उदाहरणांचे echo तसेच इतर स्पेशल कॅरक्टर चे स्पष्टीकरण सक्षम करण्यासाठी केला जातो, जसे की क्षैतिज टॅब, जे \t द्वारे दर्शविले जाते. अशा प्रकारे, उदाहरणार्थ, खालील फॉर्मॅटिंग आउटपुट तयार करेल:

**echo -e "\n Projects: \n\n\tplan \n\tcode \n\ttest\n"**

(वरील आज्ञा एका ओळीवर लिहिली जावी, जरी ती लहान डिस्प्ले स्क्रीनवर दोन ओळींप्रमाणे रेंडर केली जाऊ शकते.) आउटपुटमध्ये नवीन लाइन जोडण्यापासून echo थांबवण्यासाठी -n पर्याय वापरला जाऊ शकतो.

आउटपुट रीडायरेक्शनचा वापर करून, इको टेक्स्ट असलेली नवीन फाइल तयार करण्याचा एक अतिशय सोपा पाथ प्रदान करतो. इको टाईप करून इच्छित टेक्स्ट , आऊटपुट रीडायरेक्शन ऑपरेटर (जो उजव्या बाजूस पॉइंटिंग अँगल ब्रॅकेट आहे) आणि शेवटी नवीन फाइलचे नाव टाकून हे पूर्ण केले जाते. फाईल देखील स्पेशल कॅरक्टर वापरून फॉर्मॅटिंग केली जाऊ शकते. अशाप्रकारे, उदाहरणार्थ, वरील उदाहरणातील फॉर्मॅट केलेले आउटपुट प्रोजेक्ट1 नावाची नवीन फाइल तयार करण्यासाठी वापरले जाऊ शकते:

echo -e "\n Project1: \n\n\tplan \n\twrite \n\ttest\n" > project1

नवीन फाइलची कंटेंट , कोणत्याही फॉर्मॅट नासह, कॅट किंवा त्यापेक्षा कमी, म्हणजे, कमांड वापरून सत्यापित केली जाऊ शकते.

**less project1**

echo हा फाईलच्या शेवटी टेक्स्ट जोडण्याचा एक सोयीस्कर पाथ असू शकतो ज्याचा वापर अॅपेंड ऑपरेटरसह केला जातो, जो सलग दोन उजवीकडील पॉइंटिंग अँगल ब्रॅकेटद्वारे दर्शविला जातो. तथापि, चुकून दोन ऐवजी एकच ब्रॅकेट वापरण्याचा धोका नेहमीच असतो, ज्यामुळे

फाईलमधील सर्व कंटेंट ओव्हरराईट होते आणि अशा प्रकारे, हे वैशिष्ट्य स्क्रिप्टमध्ये वापरण्यासाठी सर्वोत्तम आरक्षित आहे.

echo चा वापर पॅटर्न मॅचिंगसह देखील केला जाऊ शकतो, जसे की वाइल्डकार्ड कॅरेक्टर, जे स्टार कॅरेक्टरद्वारे दर्शविले जाते. उदाहरणार्थ, Gif फाईल्स या कंरंट डिरेक्टरी तील सर्व .gif इमेज फाईल्सच्या नावांनंतर खालील वाक्यांश परत करेल:

```
echo -e gif filesare *.gif
```

### cal कमांड

चालू महिन्याचे कॅलेंडर प्रदर्शित करते.

```
$ cal
```

### जुलै 2012

Su	Mo	Tu	We	Th	Fr	Sa
१	२	३	४	५	६	७
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
२९	३०	३१				

'cal' स्पेसिफाइड महिन्यासाठी आणि वर्षासाठी कॅलेंडर प्रदर्शित करेल.

```
$ cal 08 1991
```

### ऑगस्ट १९९१

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3				
४	५	६	७	८	९	१०
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

### Date command

कंरंट वेळ आणि date प्रदर्शित करा.

```
$ date
```

```
शुक्र 6 जुलै 01:07:09 IST 2012
```

तुम्हाला फक्त वेळेत स्वारस्य असल्यास, तुम्ही 'date +%T' वापरू शकता

(hh:mm:ss मध्ये):

```
$ date +%T
```

```
01:13:14
```

tty कमांड

कंरंट टर्मिनल दाखवते.

```
$tty
```

```
/dev/pts/0
```

### whoami command

हा कमांड कंरंट लॉग इन केलेल्या यूजर स प्रकट करतो.

```
$ whoami
```

```
raghu
```

### id कमांड

ही आज्ञा युजर आणि ग्रुप (UID आणि GID) चे प्रिंटेड करते

कंरंट युजर .

```
$ id
```

```
uid=1000(raghu) gid=1000(raghu)
```

```
groups = 1000 (raghu), 4(adm), 20(dialout), 24(cdrom), 46(plugdev), 112(lpadmin), 120(admin),122(sambashare)
```

कंरंट यूजर बदल डीफॉल्ट इनफार्मेशन प्रदर्शित केली जाते. जर दुसरे युजर नाव आर्ग्यूमेंट्स म्हणून प्रदान केले असेल, तर त्या यूजर ची इनफार्मेशन प्रिंटेड केली जाईल:

```
$ id root
```

```
uid=0(root) gid=0(root) groups=0(root)
```

### Clear command

ही आज्ञा स्क्रीन क्लियर करते.

### Getting help command

त्याच्या सर्व फायद्यांसाठी, कमांड लाइनचा एक मोठा तोटा असा आहे की तेथे भरपूर कमांड्स आहेत आणि त्याहूनही अधिक पर्याय आणि वापर आहेत. परंतु कोणीही सर्व आज्ञा लक्षात ठेवू शकत नाही. कमांड लाइन वापरण्याचे काही हुशार पाथ आहेत. लिनक्स आम्हाला येथे चर्चा केलेली अनेक रिसोर्सेस प्रदान करते:

--help option

जवळजवळ प्रत्येक कमांडसह, '--help' पर्याय वापर दर्शवतो

त्या आदेशाचा सारांश.

```
$ date --help
```

```
Usage: date [OPTION]... [+FORMAT]
```

किंवा: date[-u|--utc|--universal] [MMDDhhmm[[CC]YY][.ss]] दिलेल्या फॉर्मॅटमध्ये कंरंट वेळ प्रदर्शित करा किंवा सिस्टम date सेट करा.

### whatis command

whatis कमांड कमांड लाइन प्रोग्राम्स (म्हणजे, सर्व-टेक्स्ट मोड प्रोग्राम्स) आणि लिनक्स आणि इतर युनिक्स-सारख्या ऑपरेटिंग सिस्टमशी संबंधित इतर विषयांचे संक्षिप्त डिस्क्रिप्शन प्रदान करते.

आर्ग्युमेंट्स (म्हणजे इनपुट डेटा) म्हणून प्रदान केलेल्या प्रत्येक कीवर्डसाठी whatis डेटाबेसमधील लहान कॅरक्टर ने शोधून ते हे साध्य करते. या डेटाबेसमध्ये मॅन्युअलमधील प्रत्येक पृष्ठाच्या NAME विभागातील फक्त टाईटल, विपार्ट क्रमांक आणि डिस्क्रिप्शन आहे जे बहुतेक युनिक्स सारख्या प्रणालींमध्ये तयार केले आहे.

### Whatis साठी syntax:

#### कीवर्ड काय आहे

उदाहरणार्थ, खालील हेडकमांडचा सिंगल ओळ सारांश प्रदान करते (जे डीफॉल्टनुसार प्रत्येक फाईलच्या पहिल्या दहा ओळी प्रदर्शित करते जे त्यास आर्ग्युमेंट्स म्हणून प्रदान केले जाते):

#### head tall काय आहे

whatis चा वापर एकाच वेळी मल्टिपल विषयांबद्दल इनफार्मेशन सर्च प्यासाठी केला जाऊ शकतो. उदाहरणार्थ, खालील डोके आणि शेपटी दोन्हीबद्दल इनफार्मेशन प्रदान करेल (जे डीफॉल्टनुसार फाईल्सच्या अंतिम दहा ओळी वाचते):

#### whatis काय आहे

whatis चे आउटपुट मर्यादित आहे कारण ते डेटाबेसमध्ये आढळणाऱ्या प्रत्येक कीवर्डसाठी फक्त एकच ओळ प्रदान करते; अशा प्रकारे ते अगदी मध्यम गुंतागुंतीच्या कमांडबद्दल अपूर्ण इनफार्मेशन पुरवते. उदाहरणार्थ, cat कमांडबद्दल इनफार्मेशन मिळवण्यासाठी whatis चा खालील वापर केल्याने "concatenate files and print on standard output" असे आउटपुट तयार होते:

#### whatis काय आहे

तथापि, यामुळे मांजरीबद्दल काही महत्त्वाची इनफार्मेशन वगळण्यात आली आहे, स्पेशल त: फायली वाचण्यासाठी वापरणे अतिशय सोयीचे आहे आणि ती फाईल्स तयार करण्यासाठी आणि लिहिण्यासाठी देखील वापरली जाते.

whatis apropos कमांड सारखे आहे. तथापि, apropos अधिक शक्तिशाली आहे कारण त्याचे आर्ग्युमेंट्स नाहीत

पूर्ण शब्दांपुरते मर्यादित परंतु स्ट्रिंग्स (म्हणजे, कॅरक्टर चे कोणतेही मर्यादित क्रम) देखील असू शकतात ज्यात शब्दांचे काही पार्ट असतात. दोन्ही कमांड असामान्य आहेत कारण त्यांच्याकडे कोणतेही पर्याय नाहीत.

मॅन कमांड (जी अंगभूत मॅन्युअल पेजेज वाचण्यासाठी वापरली जाते), जेव्हा त्याच्या -f पर्यायासह वापरली जाते, तेव्हा whatis प्रमाणेच आउटपुट तयार करते. अशा प्रकारे, उदाहरणार्थ,

#### man -f cat

च्या समतुल्य आहे

whatis cat

इनफार्मेशन पेजेज

इनफार्मेशन डॉक्युमेंट्स कधीकधी मॅन पेजपेक्षा अधिक विस्तृत असतात. परंतु काही कमांड साठी, इनफार्मेशन पेजेज मॅन पेजसारखीच असतात. हे वेब पेजसारखे आहेत. इनफार्मेशन च्या पेजमध्ये इंटरनल दुवे सबस्थित आहेत. या लिंक्सना नोड्स म्हणतात. या नोड्सद्वारे इनफार्मेशन पेजेज एका पृष्ठावरून दुसऱ्या पेजवर नेव्हिगेट केली जाऊ शकतात.

\$ info date

लिनक्स वातावरणात वर्ड प्रोसेसर

टेक्स्ट एडिटर चा वापर विविध टाइप चे लोक करतात. कॅज्युअल वापरकर्ते, लेखक, प्रोग्रामर आणि सिस्टम एडमिनिस्ट्रेटर हे सर्व लिनक्समध्ये एका वेळी किंवा दुसऱ्या वेळी टेक्स्ट एडिटर वापरतील.

text editor चा वापर

टेक्स्ट एडिटर अनेक वैशिष्ट्यांशिवाय वर्ड प्रोसेसरसारखा असतो. सर्व ऑपरेटिंग सिस्टम बेसिक टेक्स्ट संपादकासह येतात. लिनक्स अनेकांसह येतो. टेक्स्ट संपादकाचा मुख्य वापर म्हणजे साध्या टेक्स्ट मध्ये कोणतेही फॉर्मॅटिंग न करता काहीतरी लिहिणे जेणेकरून दुसरा

प्रोग्राम ते वाचू शकतो. त्या फाईलमधून मिळालेल्या इनफार्मेशन च्या आधारे, प्रोग्राम एक ना एक मार्गाने चालेल.

#### Movement commands:

तुमच्या सिस्टिमवर अवलंबून, बाण की किंवा बॅकस्पेस की काम करणार नाही. त्याऐवजी, तुम्ही समान फंक्शन करण्यासाठी या आज्ञा वापरू शकता.

To	Hold down Ctrl key and press	Instead of
Delete a character	backspace	backspace
Move up a line	p	up arrow
Move down a line	n	down arrow
Move left one space	b	sleft arrow
Move right one space	f	right arrow
Move to the end of line	e	end

vi	एकाच डिरेक्टरीमध्ये एक किंवा अधिक फाइल्स उघडण्यासाठी कमांड लाइनवर टाईप केले (vi tomato.txt चालू डिरेक्टरीमध्ये "tomato.txt" नावाची फाइल उघडते) (vi पार्सर् सेज रोझमेरी "पार्सर्," "सेज" या तीन फाइल्स उघडते. आणि करंट डिरेक्टरी "रोझमेरी")
vi *	करंट डिरेक्टरी तील प्रत्येक फाइल उघडण्यासाठी कमांड लाइनवर टाईप केले
:q	फाइल बंद करते (बाहेर पडते) ज्यामध्ये कोणतेही बदल केले गेले नाहीत
:q!	कोणतेही बदल सेव्ह न करता सोडतो
:w	करंट फाइल डिस्कवर लिहिते (म्हणजे सेव्ह करते).
:wq	डिस्कवर बफर कंटेंट लिहिते (म्हणजे बदल सेव्ह करते) आणि सोडते
ZZ	same as :wq
i	टेक्स्ट इन्सर्ट मोड ऍक्टिव्ह करते, कर्सरच्या करंट स्थितीत त्वरित टेक्स्ट इन्सर्ट .
h	कर्सर एक कॅरक्टर डावीकडे हलवतो (2h कर्सर दोन कॅरक्टर डावीकडे हलवतो)
j	कर्सर एक ओळ खाली हलवते (3j कर्सर तीन ओळी खाली हलवते)
k	कर्सरला एक ओळ हलवते
l	वर कर्सर एक कॅरक्टर उजवीकडे हलवते
g	कर्सरला इच्छित ओळीवर हलवते; सुधारित पूर्णांकाच्या आधी नसल्यास कर्सरला टेक्स्ट च्या शेवटच्या ओळीवर हलवते (5G कर्सरला पाचव्या ओळीवर हलवते)
a	इन्सर्ट मोडवर स्विच करते आणि कर्सरच्या उजवीकडे टेक्स्ट टाकण्याची परवानगी देते.
x	कर्सरच्या खाली असलेले कॅरक्टर ताबडतोब हटवते (xxx कर्सरच्या खाली असलेले कॅरक्टर लगेच हटवते आणि नंतर उजवीकडील दोन कॅरक्टर हटवते)
X	कर्सरच्या डावीकडील सिंगल कॅरक्टर हटवते
D	कर्सरच्या खाली असलेल्या अक्षरापासून ओळीच्या शेवटपर्यंत करंट ओळीवरील टेक्स्ट काढून टाकते
dw	कर्सरच्या खाली असलेले कॅरक्टर ताबडतोब हटवते आणि त्याच शब्दात त्याच्या उजवीकडे उरलेले कॅरक्टर हटवते (2dw कर्सरच्या खाली असलेले कॅरक्टर लगेच हटवते, बाकीचे कॅरक्टर त्याच्या उजवीकडे एकाच शब्दात आणि पुढील सर्व शब्दात)
dd	कर्सर असलेली संपूर्ण ओळ हटवते, आणि कर्सर नंतर पुढच्या ओळीवर जातो (2dd करंट ओळीपासून सुरू होणाऱ्या सलग दोन ओळी हटवते)
cw	कर्सरच्या खाली आणि त्याच शब्दातील त्याच्या उजवीकडे कॅरक्टर हटवते आणि त्यांना पुनर्स्थित करण्यासाठी नवीन कॅरक्टर टाईप करण्यास अनुमती देते (2cw कर्सरच्या खाली आणि त्याच शब्दातील आणि पुढील शब्दातील त्याच्या उजवीकडे कॅरक्टर हटवते, आणि नंतर परवानगी देते रिप्लेसमेंट कॅरक्टर टाईप करायचे)
cc	करंट ओळ मिटवते आणि बदली टेक्स्ट टाईप करण्यास अनुमती देते (2cc करंट ओळ आणि पुढील ओळ मिटवते आणि दोन्ही ओळींसाठी बदली टेक्स्ट टाईप करण्यास अनुमती देते)
cb	करंट शब्दातील कर्सरच्या डावीकडील कॅरक्टर हटवते आणि बदली कॅरक्टर टाईप करण्यास अनुमती देते (3cb करंट शब्दातील कर्सरच्या डावीकडील कॅरक्टर हटवते आणि त्याच्या डावीकडील दोन शब्दांसह नंतर टेक्स्ट बदलण्याची परवानगी देते. टाईप करा)
R	कर्सरच्या खाली आणि उजवीकडे टेक्स्ट एका वेळी एक कॅरक्टर ओव्हरराईट करण्यास अनुमती देऊन टेक्स्ट इनपुट मोड ऍक्टिव्ह करते
xp	दोन समीप कॅरक्टर हस्तांतरित करते
deep	दोन समीप शब्द बदलते
ddp	दोन समीप रेषा हस्तांतरित करते



~	कर्सर इंटरनल कॅरक्टर केस बदलते
J	करंट रेषेला पुढील ओळीसह जोडते
u	बफर बदललेल्या सर्वात अलीकडील कमांडचे रिझल्ट उलट करते
U	करंट भेटीदरम्यान करंट ओळीत केलेले सर्व बदल पूर्ववत करते
:s/	नियुक्त केलेल्या स्ट्रिंगच्या पहिल्या प्रसंगासाठी टेक्स्ट सर्च ते (:s/cucumber "cucumber" या स्ट्रिंगच्या पहिल्या प्रसंगासाठी टेक्स्ट सर्च ते)
n	नियुक्त स्ट्रिंगच्या पुढील उदाहरणासाठी टेक्स्ट सर्च ते
:s/ //	नियुक्त केलेल्या स्ट्रिंगचा पहिला प्रसंग बदलतो (:s/cucumber/radish/ "cucumber" या स्ट्रिंगच्या पहिल्या उदाहरणाला "मुळा" ने बदलतो)
:%s/ //	नियुक्त केलेल्या स्ट्रिंगचा प्रत्येक प्रसंग बदलतो (:%s/cucumber/radish/ "cucumber" या स्ट्रिंगच्या प्रत्येक प्रसंगाला "मुळा" ने बदलतो)
:r	दुसऱ्या फाईलमधून सध्या उघडलेल्या फाईलमध्ये टेक्स्ट समाविष्ट करते (:r lettuce.txt "lettuce.txt" नावाच्या फाईलमधून सध्या उघडलेल्या फाईलमध्ये टेक्स्ट समाविष्ट करते)
:w>	सध्या उघडलेल्या फाईलमधील टेक्स्ट दुसऱ्या फाईलमध्ये जोडते (:w> cabbage सध्या उघडलेल्या फाईलमधील टेक्स्ट "cabbage" नावाच्या फाईलमध्ये जोडते).

## फायली आणि डिरेक्टरी मॅनेज करणे (Managing files and directories)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- फेरफार करणाऱ्या फाइल्स आणि डिरेक्टरी परिभाषित करा
- बेसिक फाइल कमांड परिभाषित करा
- इतर फाइल कमांड स्पष्ट करा
- लिनक्स OS मध्ये अतिरिक्त सबयुक्त कमांड्स परिभाषित करा.

फायली किंवा डिरेक्टरी हाताळणे: लिनक्स वापरणे इतर कोणत्याही संगणक ऑपरेटिंग सिस्टमपेक्षा वेगळे नाही. तुमची इनफार्मेशन मॅनेज करण्यासाठी आणि तुमची सिस्टिम कशी कार्य करते किंवा दिसते हे मॅनेज करण्यासाठी तुम्ही तुमच्या हार्ड ड्राइव्हवर फाइल्स तयार, हटवता आणि हलवता. ही कामे जलद आणि सहज कशी करायची हा विपार्ट तुम्हाला दाखवतो. तुमची हार्ड ड्राइव्ह तुमची इनफार्मेशन मॅनेज करण्यासाठी आणि तुमची सिस्टिम कशी कार्य करते किंवा दिसते हे मॅनेज करण्यासाठी. ही कामे जलद आणि सहज कशी करायची हा विपार्ट तुम्हाला दाखवतो.

जरी Linux साठी ग्राफिकल इंटरफेस, X विंडो सिस्टिम, फाइल्स कॉपी किंवा हटवण्यासाठी ड्रॅग आणि ड्रॉप किंवा मल्टिपल निवडी देऊ शकते, तरीही तुम्ही येथे शिकू शकाल अशा अनेक कमांड या ऑपरेटिंगचा आधार बनतात. तुम्ही कन्सोल मोडमध्ये लिनक्स वापरत नसले तरीही हे प्रोग्राम कसे कार्य करतात हे जाणून घेणे योग्य आहे.

फाइल्ससह कार्य करणे:या धड्यात आपण फाइल, touch, rm, cp, mv and rename इ. सारख्या कमांडचा वापर करून फायली ओळखणे, तयार करणे, काढणे, कॉपी करणे आणि हलवणे कसे शिकतो ...

सर्व फायली केस सेन्सिटिव्ह आहेत:लिनक्स (किंवा कोणत्याही युनिक्स) वरील फाइल केस सेन्सिटिव्ह असतात. याचा अर्थ असा की FILE1 हे

file1 पेक्षा वेगळे आहे आणि /etc/hosts हे /etc/Hosts पेक्षा वेगळे आहे (नंतरचे लिनक्स संगणकावर अस्तित्वात नाही).

**फाइल कमांड:** फाइल कमांड प्रत्येक फाइलसिस्टम ऑब्जेक्टचे वर्गीकरण करण्याचा प्रयत्न करते (म्हणजे, फाइल, डिरेक्टरी किंवा लिंक) जी त्याला आर्ग्युमेंट्स म्हणून प्रदान केली जाते (म्हणजे, इनपुट). अशा प्रकारे, हे सहसा काही स्पेसिफाइड ऑब्जेक्ट आहे की नाही याबद्दल त्वरित इनफार्मेशन प्रदान करू शकते, उदाहरणार्थ, GIF89a प्रतिमा फाइल, डिरेक्टरी, एक GNU टार संग्रहण, ASCII इंग्रजी टेक्स्ट, एक प्रतीकात्मक दुवा, एक HTML डॉक्युमेंट्स, एक रिक्त फाइल, bzip2 कॉम्प्रेस डेटा, एक ELF 32-बिट LSB एक्झिक्युटेबल, इ.

एक यशस्वी होईपर्यंत प्रत्येक ऑब्जेक्टची तीन टाइप च्या चाचण्यांद्वारे तपासणी करून फाइल हे पूर्ण करते. पहिली फाइल सिस्टम टेस्टिंग आहे, जी ऑब्जेक्टच्या इनोड (ज्यात फाइलबद्दल इनफार्मेशन असते) वरून इनफार्मेशन मिळविण्यासाठी स्टेट सिस्टम कॉल वापरते. सिस्टिम कॉल ही युनिक्स सारखी ऑपरेटिंग सिस्टिममध्ये कर्नल (म्हणजे ऑपरेटिंग सिस्टिमचा मुख्य पार्ट) द्वारे केलेल्या सेवेसाठी विनंती आहे.

दुसरी टेस्टिंग मॅजिक नंबर आहे की नाही हे तपासते, जी अनेक टाइप च्या फाइल्सच्या सुरुवातीला किंवा जवळ एम्बेड केलेली नंबर आहे जी फाइल फॉर्मॅट (म्हणजे फाइलचा टाइप) दर्शवते.

पहिल्या दोन चाचण्या फाईलचा टाइप निर्धारित करण्यात अयशस्वी झाल्यास, तो साधा टेक्स्ट आहे की नाही हे निर्धारित करण्यासाठी भाषा चाचण्या वापरल्या जातात (म्हणजे, पूर्णपणे मानवी वाचनीय (रिडेबल) कॅरक्टर नी बनलेले), आणि तसे असल्यास, कोणत्या टाइप चा साधा टेक्स्ट, जसे की एचटीएमएल (हायपरटेक्स्ट मार्कअप भाषा) किंवा सोर्स कोड (म्हणजे, एखाद्या प्रोग्रामची मूळ आवृत्ती एखाद्या मानवाने लिहिलेली). या परिस्थितीत, फाईलमध्ये वापरल्या जाणाऱ्या नैसर्गिक भाषा (उदा. इंग्रजी, तुर्की किंवा जपानी) देखील निर्धारित करण्याचा प्रयत्न केला जातो.

फाइलच्या सिंटॅक्सची एक सरलीकृत आवृत्ती आहे

### file[option] object\_name(s)

फाईलमध्ये अनेक पर्याय आहेत, परंतु त्यापैकी कोणत्याही शिवाय ती सर्वात जास्त वापरली जाते. उदाहरणार्थ, सध्याच्या डिरेक्ट्री असलेल्या file1 नावाच्या फाइलबद्दल इनफार्मेशन (म्हणजे, युजर सध्या काम करत असलेली डिरेक्ट्री ) फक्त खालील टाइप करून आणि RETURN की दाबून मिळवता येईल:

### file file 1

करंट डिरेक्ट्री तील सर्व फाईल्सच्या प्रकारांबद्दल इनफार्मेशन खालीलप्रमाणे त्या डिरेक्ट्री तील प्रत्येक ऑब्जेक्टचे प्रतिनिधित्व करण्यासाठी स्टार वाइल्डकार्ड वापरून मिळवता येते:

### file \*

त्याचप्रमाणे, दुसऱ्या डिरेक्ट्रीमधील सर्व फाईल्सची इनफार्मेशन ती डिरेक्ट्रीचा आर्ग्यूमेंट्स म्हणून वापरून आणि फॉरवर्ड स्लॅश आणि स्टार वाइल्डकार्डद्वारे त्वरित फॉलो करून मिळवता येते. उदाहरणार्थ, खालील / boot डिरेक्ट्री तील सर्व ऑब्जेक्ट्सचे वर्गीकरण करते:

### file /boot/\*

चौकोनी कंसातील वाइल्डकार्ड तारा वाइल्डकार्डसह एकत्रितपणे वापरल्या जाऊ शकतात ज्यांची नावे स्पेसिफाइड अक्षरांनी सुरू होतात किंवा अक्षरांच्या स्पेसिफिक रेन्ज सह असतात. उदाहरणार्थ, करंट डिरेक्ट्री खालील फक्त तेच ऑब्जेक्ट्स दाखवतील ज्यांची नावे a ते g अक्षरांनी सुरू होतात:

### file [a-g]\*

-k पर्याय फाइलला पहिल्या यशस्वी टेस्टिंग वर थांबू नये, तर पुढे चालू ठेवण्यास सांगतो; यामुळे काही फाइलसिस्टम ऑब्जेक्ट्सबद्दल अतिरिक्त इनफार्मेशन चा रिपोर्ट मिळू शकतो. -b (म्हणजे, संक्षिप्त) पर्याय फाईलला आउटपुट लाइन्समध्ये फाइलनावे अगोदर न जोडण्यास सांगतो, जे फाइल प्रकारांबद्दल आकडेवारी संकलित करताना सबयुक्त ठरू शकते. -v पर्याय इंस्टाल केलेल्या फाइलच्या आवृत्तीबद्दल इनफार्मेशन देतो.

## files आणि डिरेक्ट्री कमांड तयार करणे

### mkdir कमांड

mkdir कमांड नवीन डिरेक्ट्री तयार करण्यासाठी वापरली जाते.

काही ऑपरेटिंग सिस्टीममध्ये फोल्डर म्हणून संदर्भित असलेली डिरेक्ट्री,

यूजर ला इतर डिरेक्ट्री आणि फाइल्ससाठी कंटेनर म्हणून दिसते. तथापि, युनिक्स सारखी ऑपरेटिंग सिस्टीम स्ट्रीट डिरेक्ट्री फक्त एक स्पेशल टाइप ची फाईल आहे ज्यात फाइल नावांची लिस्ट आणि त्यांच्याशी संबंधित इनोड क्रमांक असतात. प्रत्येक inode क्रमांक एक inode संदर्भित करतो, जो inode टेबलमध्ये स्थित असतो (जे फाइल सिस्टमच्या आजूबाजूच्या मोक्याच्या ठिकाणी ठेवलेले असते) आणि ज्यामध्ये फाईलची सर्व इनफार्मेशन असते (उदा. आकार, परवानग्या आणि निर्मितीची date) शिवाय त्याचे नाव आणि वास्तविक फाइलमध्ये समाविष्ट असलेला डेटा.

mkdir चे खालील उदाहरण आहे

```
$ mkdir example
```

```
$ ls -l
```

```
total 4
```

```
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 उदाहरण
directory_name हे कोणत्याही डिरेक्ट्रीचे नाव आहे जी युजर mkdir तयार करण्यास सांगत आहे. एकाच वेळी कितीही डिरेक्ट्री तयार केल्या जाऊ शकतात.
```

अशा प्रकारे, उदाहरणार्थ, खालील कमांड सध्याच्या डिरेक्ट्रीमध्ये dir\_1, dir\_2 आणि dir\_3 या नावांसह तीन डिरेक्ट्री तयार करेल (म्हणजे, युजर सध्या काम करत असलेली डिरेक्ट्री )

```
mkdir dir_1 dir_2 dir_3
```

mkdir ला आर्ग्यूमेंट्स (म्हणजे इनपुट) म्हणून दिलेले डिरेक्ट्री चे नाव विद्यमान डिरेक्ट्री किंवा त्याच डिरेक्ट्रीमधील फाइल सारखेच असेल ज्यामध्ये युजर mkdir ला नवीन डिरेक्ट्री तयार करण्यास सांगत असेल तर mkdir चेतावणी संदेश देईल. mkdir म्हणून: डिरेक्ट्री 'dir\_1' तयार करू शकत नाही:

फाइल अस्तित्वात आहे आणि त्या नावाची फाइल तयार करणार नाही. तथापि, ते टच च्या सहाय्याने इतर कोणत्याही नावांसाठी डिरेक्ट्री तयार करणे सुरू ठेवेल.

सक्षम होण्यासाठी यूजर ला मूळ डिरेक्ट्री (म्हणजेच नवीन डिरेक्ट्री ज्या डिरेक्ट्री तयार करायची आहे) लिहिण्याची परवानगी (म्हणजे, फाइल किंवा डिरेक्ट्री तयार करण्यासाठी किंवा बदलण्यासाठी सिस्टमची परवानगी) असणे आवश्यक आहे. नवीन डिरेक्ट्री तयार करण्यासाठी.

mkdir द्वारे तयार केलेल्या डिरेक्ट्रीमध्ये ऑटोमैटिक दोन लपविलेल्या डिरेक्ट्री समाविष्ट केल्या जातात, एक नुकतीच तयार केलेली डिरेक्ट्री दर्शवते (आणि एका पॉइंट द्वारे दर्शविली जाते) आणि दुसरी त्याच्या मूळ डिरेक्ट्रीचे प्रतिनिधित्व करते (आणि सलग दोन बिंदूनी दर्शविली जाते). हे त्याच्या -a पर्यायासह ls (म्हणजे, लिस्ट ) कमांड वापरून पाहिले जाऊ शकते, जे ls ला सर्व डिरेक्ट्री आणि फाइल्स दाखवण्यास सांगते, (लपलेल्यासह) कोणत्याही डिरेक्ट्रीमध्ये आर्ग्यूमेंट्स म्हणून किंवा करंट डिरेक्ट्री . कोणतेही आर्ग्यूमेंट्स नसल्यास, म्हणजे,

ls -a

mkdir च्या -m पर्यायाचा वापर नवीन डिरेक्टरीच्या परवानग्या कंट्रोल करण्यासाठी केला जातो. नवीन डिरेक्टरीज बाय डीफॉल्ट रीड, राईट आणि एक्झिक्यूट (म्हणजे प्रोग्राम असल्यास प्रोग्राम म्हणून चालवल्या जातात) मालक (म्हणजे डीफॉल्टनुसार डिरेक्टरीचा निर्माता) आणि ग्रुपसाठी सक्षम केलेल्या परवानग्या आणि वाचन आणि कार्यान्वित केलेल्या परवानग्यांसह तयार केल्या जातात. इतर वापरकर्ते. अशा प्रकारे, उदाहरणार्थ, dir\_4 नावाची डिरेक्टरी तयार करण्यासाठी ज्यासाठी सर्व वापरकर्त्यांसाठी सर्व तीन टाइप च्या परवानग्या सक्षम केल्या होत्या, सिक्सेस 777 -m नंतर वापरला जाईल, उदाहरणार्थ:

#### mkdir -m 777 dir\_4

पहिला अंक मालकाचे प्रतिनिधित्व करतो, दुसरा गटाचे प्रतिनिधित्व करतो आणि तिसरा इतर वापरकर्त्यांचे प्रतिनिधित्व करतो. क्रमांक 7 सर्व तीन टाइप च्या परवानगीचे प्रतिनिधित्व करतो (म्हणजे, वाचा, लिहा आणि कार्यान्वित करा), 6 म्हणजे फक्त वाचणे आणि लिहा, 5 म्हणजे केवळ वाचणे आणि कार्यान्वित करणे, 4 फक्त वाचणे, 3 लिहा आणि चालवणे, 2 फक्त लिहा, 1 फक्त कार्यान्वित आहे आणि 0 कोणतीही परवानगी नाही.

अशा प्रकारे, उदाहरणार्थ, dir\_5 नावाची नवीन डिरेक्टरी तयार करण्यासाठी ज्यासाठी मालकाने वाचन आणि लेखन परवानगी दिली आहे, गटाला वाचण्याची परवानगी आहे आणि इतर वापरकर्त्यांना कोणतीही परवानगी नाही, खालील गोष्टी वापरल्या जातील:

#### mkdir -m 640 dir\_5

-p (म्हणजे पालक) पर्याय नवीन डिरेक्टरी साठी स्पेसिफाइड इंटरमीडिएट डिरेक्टरी तयार करतो जर त्या आधीपासून अस्तित्वात नसतील. उदाहरणार्थ, खालील डिरेक्टरी रचना तयार करण्यासाठी याचा वापर केला जाऊ शकतो:

#### mkdir -p food/fruit/citrus/oranges

आर्ग्युमेंट म्हणून पहिल्या डिरेक्टरीच्या नावासह du (म्हणजे डिस्क वापर) कमांड वापरून निर्देशिकांची ही सीरीज तयार केली गेली आहे याची कन्फर्म करणे खूप सोपे आहे. वरील उदाहरणाच्या बाबतीत असे होईल

#### du food

इतर पर्यायांमध्ये -v (म्हणजे, वर्बोज), जे प्रत्येक तयार केलेल्या डिरेक्टरी साठी संदेश परत करते, --help, जे mkdir बदल थोडक्यात इनफार्मेशन देते आणि --version, जे सध्या इंस्टाल mkdir प्रोग्रामचा आवृत्ती क्रमांक परत करते.

टच कमांड: टच कमांड प्रत्येक FILE च्या ऍक्सेस आणि फेरफार वेळा करंट सिस्टम वेळेवर अपडेट करते.

तुम्ही आधीच अस्तित्वात नसलेली FILE स्पेसिफाइड केल्यास, टच च्या सहाय्याने त्या नावाची रिकामी फाइल तयार होते.

जर FILE आर्ग्युमेंट्स डॅश असेल तर (" - ") स्पेशल हाताळले जाते आणि स्टॅण्डर्ड आउटपुटशी संबंधित फाइलच्या वेळा बदलण्यास स्पर्श करण्यास कारणीभूत ठरते.

```
$ touch file1 file2 file3
```

```
$ls -l
```

```
total 4
```

```
drwxr-xr-x 2 raghu raghu 4096 2012-07-06 14:09 example
```

```
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
```

```
-rw-r--r-- 1 raghu raghu 0 2012-07-06 14:20 file1
```

```
-rw-r--r--1 raghu raghu 0 2012-07-06 14:20 file1
```

टचचे वैशिष्ट्य म्हणजे, cp (ज्याचा वापर फाइल्स आणि डिरेक्टरी कॉपी करण्यासाठी केला जातो) आणि mv (ज्याचा वापर फाइल्स आणि डिरेक्टरीज हलवण्यासाठी किंवा पुनर्नामित करण्यासाठी केला जातो) यांसारख्या काही कमांड्सच्या उलट, ते ऑटोमॅटिक ओव्हरराईट होत नाही (म्हणजे, पुसून टाकते. ची कंटेंट ) समान नावाच्या विद्यमान फायली. त्याऐवजी, ते अशा फायलींसाठी फक्त शेवटच्या एक्सेस वेळा करंट वेळेत बदलते.

यूजर ला फायलींसाठी टाइमस्टॅम्प बदलण्याची परवानगी देण्यासाठी अनेक स्पर्श पर्याय स्पेशल तः डिझाइन केलेले आहेत. उदाहरणार्थ, -a पर्याय केवळ एक्सेस वेळ बदलतो, तर -m पर्याय केवळ बदल वेळ बदलतो. या दोन्ही पर्यायांचा एकत्र वापर केल्याने एक्सेस आणि बदल दोन्ही वेळ करंट वेळेत बदलते, उदाहरणार्थ:

#### touch -am file3

-r (म्हणजे रेफरन्स ) पर्याय थेट स्पेस आणि नंतर फाइलच्या नावाने टचला करंट वेळेऐवजी फाइलचे टाइम स्टॅम्प वापरण्यास सांगते. उदाहरणार्थ, फाइल 5 साठी फाइल 4 च्या वेळा वापरण्यास खालील गोष्टी सांगतील:

#### touch -r file4 file5

-B पर्याय स्पेसिफाइड सेकंदांच्या संख्येच्या मागे जाऊन टाइमस्टॅम्पमध्ये बदल करतो आणि -F पर्याय सेकंदांच्या स्पेसिफाइड संख्येच्या पुढे जाऊन वेळ सुधारतो. उदाहरणार्थ, खालील कमांड फाइल 6 पेक्षा 30 सेकंद जुनी करेल.

#### touch -r file 6 -B 30 file7

-d आणि -t पर्याय यूजर ला स्पेसिफिक अंतिम एक्सेस वेळ जोडण्याची परवानगी देतात. आधीच्या पाठोपाठ date, महिना, वर्ष, मिनिट:सेकंड फॉर्मॅटमध्ये स्ट्रिंग (म्हणजे कॅरक्टर चा क्रम) येतो आणि नंतरचा [[CC] YY]MMDDhhmm[.ss] फॉर्मॅट वापरतो. उदाहरणार्थ, 1 मे 2005, 1 मे 2005 10:22 सकाळी 10:22 पर्यंत फाइल 8 ची शेवटची एक्सेस वेळ बदलण्यासाठी 10:22 सिंगल अवतरणांमध्ये बंद केली जाईल आणि खालीलप्रमाणे वापरली जाईल, म्हणजे:

#### touch -d '1 May 2005 10:22' file8

आंशिक date-वेळ स्ट्रिंग वापरल्या जाऊ शकतात. उदाहरणार्थ, खालील फाइल 9 साठी दर्शविल्याप्रमाणे, फक्त date प्रदान करणे आवश्यक आहे (ज्या बाबतीत वेळ ऑटोमॅटिक पणे 0:00 वर सेट केली जाते):

## touch -d '14May' file9

खाली दर्शविल्याप्रमाणे, फक्त वेळ प्रदान करून, ऑटोमॅटिक पणे date करंट तारखेमध्ये बदलते:

## touch -d '14:24' file9

फाइल्ससाठी शेवटची फेरफार date पाहण्याचा सर्वात सामान्यपणे वापरला जाणारा पाथ म्हणजे ls कमांडचा -l पर्यायासह वापर करणे. उदाहरणार्थ, file10 नावाच्या फाईलच्या बाबतीत हे असेल

## ls -l file10

स्टेट कमांड वापरून कोणत्याही फाईल किंवा डिरेक्टरीचे संपूर्ण टाइमस्टॅम्प पाहिले जाऊ शकतात. उदाहरणार्थ, file11 नावाच्या फाईलसाठी खालील टाइमस्टॅम्प दर्शविले:

## state file 11

--help पर्याय पर्यायांची बेसिक लिस्ट दाखवतो, आणि --version पर्याय सध्या इंस्टाल टच प्रोग्रामची आवृत्ती परत करतो.

आज्ञा कॉपी करा, हलवा आणि काढा

कॉपी कमांड:cp कमांड फाइल्स आणि डिरेक्टरी कॉपी करण्यासाठी वापरली जाते. प्रती मूळपासून स्वतंत्र होतात (म्हणजे, एकामध्ये नंतरच्या बदलामुळे दुसऱ्यावर रिझल्ट होणार नाही).

## cp चे बेसिक वाक्यरचना आहे

### \$cp सोर्स डेस्टिनेशन

सुरक्षेची खबरदारी म्हणून, डीफॉल्ट cp फक्त फाइल्स कॉपी करते डिरेक्टरी नाही. फाईलच्या कॉपीला नियुक्त केलेल्या समान नावाची फाईल (किंवा डिरेक्टरी च्या कॉपीला नियुक्त केलेल्या समान नावाची डिरेक्टरी ) आधीपासून अस्तित्वात असल्यास, ती ओवरराइडिंग केली जाईल (म्हणजे, त्यातील कंटेंट गमावली जाईल) . तथापि, कॉपीसाठी मालक, ग्रुप आणि परवानग्या त्या फाईलच्या नावाप्रमाणेच बनतात ज्याने ते बदलले आहे. सोर्स फाइलची शेवटची एक्सेस वेळ आणि नवीन फाइलची शेवटची फेरबदल वेळ कॉपी केली गेली त्या वेळेवर सेट केली जाते.

जेव्हा एखादी प्रत फाइल किंवा डिरेक्टरीची बनवली जाते, तेव्हा कॉपीला मूळ नावापेक्षा वेगळे नाव असणे आवश्यक आहे जर ती मूळ डिरेक्टरी ठेवायची असेल. तथापि, प्रत वेगळ्या डिरेक्टरी बनविल्यास त्याचे नाव समान असू शकते. अशाप्रकारे, उदाहरणार्थ, सध्याच्या डिरेक्टरी तील फाइल (म्हणजे, युजर सध्या काम करत असलेली डिरेक्टरी ) file1 नावाची फाइल त्याच नावाने दुसऱ्या डिरेक्टरी कॉपी केली जाऊ शकते, जसे की /home/john/ मध्ये, खालीलप्रमाणे:

## cp file1 /home/john/file1

डिरेक्टरीच्या नावापुढे त्यांची नावे लिस्ट बद्ध करून कितीही फाईल्स एकाच वेळी दुसऱ्या डिरेक्टरी कॉपी केल्या जाऊ शकतात. cp ही एक बुद्धिमान कमांड आहे आणि जेव्हा फक्त अंतिम आर्ग्युमेंट्स (म्हणजे इनपुट डेटाचा तुकडा) एक डिरेक्टरी असेल तेव्हा हे करणे माहित आहे. डिरेक्टरीमध्ये कॉपी केलेल्या फायलींना मूळ नावांसारखीच नावे असतील. अशा प्रकारे,

उदाहरणार्थ, खालील फाइल 2, फाइल3 आणि फाइल4 नावाच्या फाइल्स dir1 नावाच्या डिरेक्टरीमध्ये कॉपी करेल:

## cp file2 file3 file4 dir1

-r (म्हणजे, रिकर्सिव्ह) पर्याय, जो मोठ्या केस R ने देखील लिहिला जाऊ शकतो, त्यांच्या सर्व कंटेंट सह निर्देशिकांना कॉपी करण्याची परवानगी देतो. (डिरेक्टरीज डिफॉल्टनुसार कॉपी केल्या जात नाहीत जेणेकरून वापरकर्त्यांना चुकीने विद्यमान डिरेक्टरी ओव्हरराइट करणे अधिक कठीण व्हावे जेणेकरून प्रत तयार केलेल्या कॉपीला नियुक्त केलेल्या नावाप्रमाणेच नाव असेल आणि ज्यात गंभीर डिरेक्टरी स्ट्रक्चर किंवा महत्त्वपूर्ण डेटा असू शकतो.) अशा प्रकारे, यासाठी

उदाहरणार्थ, खालील कमांड dir2 नावाच्या विद्यमान डिरेक्टरी ची प्रत तयार करेल, ज्यामध्ये सर्व कंटेंट समाविष्ट आहे (उदा. फाइल्स, सबडिरेक्टरी , त्यांच्या सबडिरेक्टरी इ.), dir3 म्हणतात:

## cp -r dir2 dir3

-i (म्हणजे, परस्परसंवादी) पर्याय यूजर ला प्रतला नियुक्त केलेले कोणतेही नाव दुसऱ्या फाईलद्वारे आधीपासूनच वापरात असल्याच्या घटनेत सूचित करतो आणि त्यामुळे ती फाईल ओवरराइडिंग केली जाईल. प्रॉम्प्टला प्रतिसाद म्हणून y अक्षर (लोअर केस किंवा अप्पर केस) एंटर केल्याने कमांड चालू राहते; इतर कोणतेही उत्तर कमांडला फाईल ओव्हरराइट करण्यापासून प्रतिबंधित करते. अशा प्रकारे, उदाहरणार्थ, dir4 नावाच्या डिरेक्टरीची प्रत बनवायची असेल आणि त्याला dir5 म्हणायचे असेल आणि dir4 नावाची डिरेक्टरी आधीपासून अस्तित्वात असेल तर, नंतरच्या डिरेक्टरीमध्ये समान नाव असलेल्या फाइल्स बदलण्यापूर्वी खालील यूजर ला सूचित करेल:

## cp -ri dir4 dir5

-a पर्याय नवीन डिरेक्टरीमध्ये मूळ डिरेक्टरी आणि त्यातील कंटेंट ची जास्तीत जास्त रचना आणि प्रॉपर्टीज सेव्ह करतो आणि त्यामुळे संग्रहण तयार करण्यासाठी सबयुक्त आहे. हे -r पर्यायासारखे आहे ज्यामध्ये ते डिरेक्टरी आवर्तीपणे कॉपी करते; तथापि, ते कधीही प्रतिकामक दुव्यांचे अनुसरण करत नाही. हे पर्यायांच्या -rdp संयोजनाच्या समतुल्य आहे.

स्टार वाइल्डकार्ड वापरून डिरेक्टरीमधील सर्व फाइल्स दुसऱ्या डिरेक्टरीमध्ये कॉपी केल्या जाऊ शकतात. तारा कॅरक्टर हे कोणतेही सिंगल कॅरक्टर किंवा कॅरक्टर चे कोणतेही संयोजन दर्शवते. अशा प्रकारे, उदाहरणार्थ, dir6 नावाच्या डिरेक्टरीमधील सर्व फाइल्स dir7 नावाच्या दुसऱ्या विद्यमान डिरेक्टरीमध्ये कॉपी केल्या जातील:

## cp dir6/\* dir7

cp चा वापर फाइल्स आणि डिरेक्टरी निवडण्यासाठी स्टार वाइल्डकार्ड किंवा इतर पॅटर्न जुळणार्या कॅरक्टर सह देखील केला जाऊ शकतो. उदाहरणार्थ, सध्याच्या डिरेक्टरीमधील सर्व फाइल्स ज्यांचे फाईलनाम एक्सपान्शन .html आहे त्या dir8 नावाच्या दुसऱ्या विद्यमान डिरेक्टरीमध्ये कॉपी करण्यासाठी, खालील गोष्टी वापरल्या जातील:

## cp \*.html dir8

या प्रकरणात, स्टार वाइल्डकार्ड कोणत्याही गोष्टीचे प्रतिनिधित्व करते ज्याचे नाव .html एक्सटेन्शन ने संपते.

cp साठी इतर पर्यायांपैकी -b, जे प्रत्येक डेस्टिनेशन फाइलच्या बॅकअप प्रती बनवते, -f (म्हणजे, फोर्स), जे डेस्टिनेशन फाइल्स काढून टाकते ज्या उघडल्या जाऊ शकत नाहीत आणि पुन्हा प्रयत्न करतात, -s, जे कॉपी करण्याऐवजी प्रतिकात्मक दुवे बनवते, -u (म्हणजे अपडेट), जे फक्त जर सोर्स फाइल डेस्टिनेशन फाइलपेक्षा नवीन असेल किंवा डेस्टिनेशन फाइल गहाळ असेल तरच कॉपी करते, -v (म्हणजे, व्हर्बोज), जे काय चालले आहे याबद्दल थोडक्यात कमेंट देते आणि -x, जे cp ला समान फाइल सिस्टमवर राहण्यास सांगते.

Move कमांड:mv कमांड फाइल्स आणि डिरेक्ट्रीजचे नाव बदलण्यासाठी आणि हलवण्यासाठी वापरली जाते. त्याची सामान्य वाक्यरचना आहे:

## \$ mv सोर्स डेस्टिनेशन

आर्ग्यूमेंट्स फाइल्स आणि निर्देशिकांची नावे आहेत. जर दोन फाइलची नावे आर्ग्यूमेंट्स म्हणून दिली असतील, तर mv पहिल्याचे दुसरे नाव बदलते. जर वितर्काची लिस्ट दिली असेल आणि क्रमातील अंतिम आर्ग्यूमेंट्स विद्यमान डिरेक्टरी चे नाव असेल, तर mv इतर सर्व आयटम त्या डिरेक्ट्री हलवते. अंतिम आर्ग्यूमेंट्स विद्यमान डिरेक्टरी नसल्यास आणि दोनपेक्षा जास्त आर्ग्यूमेंट्स प्रदान केले असल्यास, एक त्रुटी संदेश परत केला जातो.

जर डेस्टिनेशन फाइल सोर्स फाइल सारख्या डिरेक्ट्री स्थित असेल, तर सोर्स फाइल फक्त पुनर्नामित केली जाऊ शकते. जर दोन्ही वेगवेगळ्या डिरेक्टरीमध्ये असतील, तर सोर्स फाइल डेस्टिनेशन युक्तिवादामध्ये नाव दिलेल्या डिरेक्ट्री हलवली जाते, ज्यामध्ये ती त्याचे मूळ नाव ठेवू शकते किंवा नवीन नाव नियुक्त केले जाऊ शकते. जर टारगेट एक डिरेक्टरी असेल, तर सोर्स फाइल किंवा डिरेक्टरी त्या डिरेक्ट्री हलविली जाते आणि तिचे मूळ नाव राखून ठेवते.

अशाप्रकारे, उदाहरणार्थ, खालील फाइल 1 नावाच्या फाइलचे नाव बदलून file2 ठेवेल, ती करंट डिरेक्ट्री ठेवली जाईल (म्हणजे, युजर सध्या काम करत असलेली डिरेक्टरी):

## mv file1 file2

खालील फाइल 3 नावाची फाइल, तिचे नाव न बदलता, सध्याच्या डिरेक्टरीमधून dir1 नावाच्या करंट डिरेक्टरीच्या विद्यमान सबडिरेक्टरीमध्ये हलवेल:

## mv file3 dir1/file3

mv चा वापर एकाच वेळी कितीही फाइल्स आणि डिरेक्टरी हलवण्यासाठी करता येतो. उदाहरणार्थ, खालील कमांड सर्व फाइल्स आणि डिरेक्ट्रीजसह हलवते

त्या डिरेक्टरीची कंटेंट, सध्याच्या डिरेक्ट्रीपासून /home/alice/new/ डिरेक्टरीमध्ये:

## mv \* /home/alice/new/

asterisk हे वाइल्डकार्ड कॅरक्टर आहे जे कोणत्याही स्ट्रिंगचे प्रतिनिधित्व करते (म्हणजे, कॅरक्टर चा क्रम). अशा प्रकारे, वरील उदाहरणामध्ये ते सध्याच्या डिरेक्टरी तील प्रत्येक फाइल आणि डिरेक्टरी चे नाव दर्शवते.

mv फाइल किंवा डिरेक्ट्रीच्या पदानुक्रमावर (म्हणजे, root डिरेक्ट्रीच्या जवळ) खाली हलवणे तितकेच सोपे करते. उदाहरणार्थ, खालील फाइल 4 नावाची फाइल, जी सध्या यूजर च्या होम डिरेक्ट्रीच्या सब-डिरेक्टरी dir/ dir/ मध्ये स्थित आहे, यूजर च्या होम डिरेक्टरीमधील शीर्ष लेयर वर हलवेल:

## mv dir/dir/file4 ~

root डिरेक्टरी ही अशी डिरेक्टरी आहे ज्यामध्ये युनिक्स सारख्या ऑपरेटिंग सिस्टमवरील इतर सर्व डिरेक्टरी असतात आणि जी डिरेक्टरीच्या पदानुक्रमाच्या शीर्षस्थानी असते. यूजर ची होम डिरेक्टरी ही अशी डिरेक्टरी असते ज्यामध्ये युजर सिस्टममध्ये लॉग इन केल्यानंतर डीफॉल्टनुसार स्वतःला सर्च तो आणि ज्याला टिल्ड (वेळी क्षैतिज रेखापिक्चर) द्वारे दर्शविले जाऊ शकते.

डीफॉल्टनुसार, एमव्ही डिस्प्ले स्क्रीनवर कोणतीही कन्फर्म प्रदान करत नाही जर त्याची क्रिया समस्यांशिवाय पूर्ण झाली. तो युनिक्स तत्त्वज्ञानाच्या मौनाच्या नियमाशी सुसंगत आहे.

अशाप्रकारे युनिक्स सारख्या ऑपरेटिंग सिस्टीमसाठी नवीन वापरकर्त्यांनी नेहमी -i पर्याय वापरणे शहाणपणाचे आहे, जे mv ला परस्परसंवादी बनवते ज्या परिस्थितीत डेस्टिनेशन डिरेक्ट्री समान नावाच्या फाइल्स आणि/ किंवा डिरेक्टरी आधीपासूनच अस्तित्वात आहेत. उदाहरणार्थ, वरील आज्ञा खालीलप्रमाणे परस्परसंवादी केली जाईल:

## mv -i \* /home/alice/new/

mv च्या इतर काही पर्यायांपैकी -b, जे त्यास प्रत्येक फाइलची बॅकअप प्रत बनवण्यास सांगते जी अन्यथा ओवरराइडिंग किंवा काढून टाकली जाईल आणि -v, जे त्यास बेव्हरबोस करण्यास सांगते आणि प्रत्येक फाइल हलवण्यापूर्वी त्याचे नाव प्रदर्शित करते. mv बद्दल तपशीलवार इनफार्मेशन (सर्व पर्यायांसह) त्याच्या --help पर्यायाचा वापर करून मिळवता येते आणि सध्याच्या आवृत्तीबद्दल इनफार्मेशन त्याच्या --version पर्यायाचा वापर करून मिळवता येते.

## काढा किंवा डिलेट करा (Remove or Delete)

### \$ rmdir

'rmdir' कमांड कोणत्याही रिकाम्या डिरेक्टरी काढून टाकते, परंतु त्यात फाइल असल्यास डिरेक्टरी हटवू शकत नाही. 'rmdir' कमांड वापरण्यासाठी, तुम्ही प्रथम सबस्थित असलेल्या सर्व फाइल्स काढून टाकल्या पाहिजेत

तुम्ही काढू इच्छित असलेली डिरेक्टरी (आणि शक्यतो डिरेक्टरी असल्यास).

फायली आणि डिरेक्टरी रिमूव्ह: rm (म्हणजे, काढून टाका) कमांड लिनक्स आणि इतर युनिक्स सारख्या ऑपरेटिंग सिस्टमवरील फाइल्स आणि डिरेक्टरी हटवण्यासाठी वापरली जाते.

## rm साठी सामान्य वाक्यरचना आहे:

### rm [options] [-r directories] filenames

चौरस कंसातील आयटम ऐच्छिक आहेत. फक्त एक किंवा अधिक फायलीच्या नावांसह वापरल्यास, rm यूजर कडून कन्फर्म न घेता त्या सर्व फायली हटवते. अशाप्रकारे, खालील उदाहरणात, rm हे तिन्ही कॅन्डिडेट्स (म्हणजे, युजर सध्या काम करत असलेली डिरेक्टरी) मध्ये स्थित आहेत असे गृहीत धरून file1, file2 आणि file3 नावाच्या फाइल्स त्वरित हटवेल:

### rm file1 file2 file3

फाइल अस्तित्वात नसल्यास किंवा यूजर ला ती हटवण्याची योग्य परवानगी नसल्यास त्रुटी संदेश परत केले जातात. राइट-संरक्षित फायली यूजर ला कन्फर्म करणासाठी (होयसाठी y आणि नाहीसाठी n सह) काढण्यापूर्वी सूचित करतात. राईट प्रोटेक्टेड डिरेक्टरीमध्ये असलेल्या फाइल्स कधीही काढल्या जाऊ शकत नाहीत, जरी त्या फाइल्स राइट प्रोटेक्टेड नसल्या तरीही.

-f (म्हणजे, बल) पर्याय rm ला सर्व स्पेसिफाइड फायली काढून टाकण्यास सांगतो, मग ते लेखन-संरक्षित असो वा नसो, यूजर ला प्रॉम्प्ट न करता. तो एरर मेसेज दाखवत नाही किंवा स्पेसिफाइड फाइल अस्तित्वात नसल्यास एरर स्टेटस रिटर्न करत नाही. तथापि, लेखन-संरक्षित डिरेक्टरी तील फाइल्स काढून टाकण्याचा प्रयत्न केल्यास, हा पर्याय त्रुटी संदेश दडपणार नाही.

-i (म्हणजे, परस्परसंवादी) पर्याय rm ला प्रत्येक फाइल आणि डिरेक्टरी काढून टाकण्यापूर्वी यूजर ला कन्फर्म करणासाठी सूचित करण्यास सांगतो. -f आणि -i दोन्ही पर्याय स्पेसिफाइड केले असल्यास, स्पेसिफाइड केलेला शेवटचा पर्याय प्रभावित होतो.

सुरक्षितता उपाय म्हणून, rm डिफॉल्टनुसार डिरेक्टरी हटवत नाही. डिरेक्टरी हटवण्यासाठी, -r पर्याय वापरणे आवश्यक आहे, जे -R पर्यायासारखेच आहे. हा पर्याय आर्ग्यूमेंट्स लिस्ट मधील डिरेक्टरी आणि त्यांची कॅन्डेट वारंवार काढून टाकतो; म्हणजेच, स्पेसिफाइड डिरेक्टरी प्रथम कोणत्याही सबडिरेक्टरी (त्यांच्या सबडिरेक्टरी आणि फाइल्स इ. सह) आणि फाइल्समधून रिक्त केल्या जातील आणि नंतर काढल्या जातील. -f पर्याय वापरला जात नाही तोपर्यंत यूजर ला डिरेक्टरीमधील कोणत्याही लेखन-संरक्षित फाइल्स काढून टाकण्यासाठी सामान्यपणे सूचित केले जाते.

जर rm द्वारे समोर आलेली फाइल प्रतीकात्मक दुवा असेल, तर ती लिंक काढून टाकली जाईल, परंतु ती लिंक ज्या फाइल किंवा डिरेक्टरी शी संबंधित असेल त्यावर रिझल्ट होणार नाही. जोपर्यंत यूजर कडे ती लिंक आहे त्या डिरेक्टरी साठी लिहिण्याची परवानगी असेल तोपर्यंत यूजर ला प्रतीकात्मक दुवा हटवण्यासाठी लिहिण्याची परवानगी आवश्यक नसते.

rm कमांड -- (दोन सलग डॅश) पॅरामीटरला परिशीलक म्हणून सपोर्ट देते जे पर्यायांचा शेवट सूचित करते. फाइलचे नाव किंवा

डिरेक्टरी डॅश किंवा हायफनने सुरू होते. उदाहरणार्थ, खालील -dir1 नावाची डिरेक्टरी काढून टाकते:

### rm -r --- -dir1

इतर पर्यायांमध्ये -v (म्हणजे, व्हर्बोज), जे काय घडत आहे याबद्दल अतिरिक्त इनफार्मेशन प्रदान करते, --help, जे rm बद्दल बेसिक डॉक्युमेंट्स प्रदान करते आणि -- आवृत्ती, जी सध्या वापरात असलेल्या rm ची आवृत्ती सांगते. rm च्या विविध आवृत्त्यांमध्ये काही फरक आहेत, त्यामुळे स्पेसिफिक सिस्टिम साठी कागदपत्रे वाचणे नेहमीच शहाणपणाचे असते.

rmdir कमांड rm पेक्षा वेगळी आहे कारण ती फक्त रिकामी डिरेक्टरी काढण्यासाठी वापरली जाते

### rmdir कमांड

rmdir कमांड लिनक्समधील रिकाम्या डिरेक्टरी काढून टाकण्यासाठी वापरली जाते rmdir साठी सिंटॅक्स आहे

### rmdir [option] Directory\_names

कोणत्याही पर्यायांशिवाय वापरल्यास, rm अशा डिरेक्टरींना लेखन परवानगी आहे की नाही याची पर्वा न करता आर्ग्यूमेंट्स (म्हणजे इनपुट) म्हणून पुरवठा केलेल्या रिकाम्या डिरेक्टरी हटवेल. अशा प्रकारे, उदाहरणार्थ, खालील कमांड सध्याच्या डिरेक्टरीमध्ये असलेल्या dir1 आणि dir2 नावाच्या दोन रिकाम्या डिरेक्टरी काढून टाकेल (म्हणजे, युजर सध्या काम करत असलेली डिरेक्टरी):

### rmdir dir1 dir2

फक्त रिकाम्या डिरेक्टरी काढण्याची क्षमता ही अंगभूत सुरक्षा आहे जी डेटाचे अपघाती नुकसान टाळण्यास मदत करते. हे महत्वाचे आहे कारण एकदा हटवल्यानंतर, युनिक्स सारखी ऑपरेटिंग सिस्टिम 1 वर हटवलेला डेटा पुनर्प्राप्त करणे अत्यंत कठीण किंवा अशक्य आहे.

-p (म्हणजे पालक) पर्याय rmdir ला स्पेसिफाइड डिरेक्टरी च्या मूळ डिरेक्टरी काढून टाकण्यास सांगतो जर प्रत्येक क्रमिक पॅरेंट डिरेक्टरी रिकामी होईल आणि प्रत्येक पॅरेंट डिरेक्टरीला लेखन परवानगी असेल तर. अशाप्रकारे, उदाहरणार्थ, dir5 रिकामे असल्यास dir5, dir4 आणि dir3 खालील काढून टाकतील, dir4 मध्ये फक्त dir5 आणि dir3 मध्ये फक्त dir4 असेल (ज्यामध्ये dir5 असेल):

### rmdir -p dir3/dir4/dir5

हे mkdir कमांडच्या -p पर्यायासह सममिती प्रदान करते, ज्याचा वापर डिरेक्टरी तयार करण्यासाठी केला जातो. अशा प्रकारे, नेस्टेड डिरेक्टरीचा वरील संच खालील गोष्टींसह सहजपणे तयार केला जाऊ शकतो:

### mkdir -p dir3/dir4/dir5

rm कमांडच्या उलट, ज्याचा वापर फाइल्स आणि डिरेक्टरी दोन्ही हटवण्यासाठी केला जातो, rmdir साठी -r पर्याय नाही. किमान Linux वर स्टॅण्डर्ड असलेल्या GNU आवृत्तीवर. तो पर्याय rm ला डिरेक्टरीच्या सर्वात खालच्या लेयर वरील सबडिरेक्टरीजपासून सुरुवात करून, त्यातील सर्व टेक्स्ट हटवून डिरेक्टरी हटवण्याची परवानगी देतो. अशाप्रकारे, जर यूजर ला संपूर्ण डिरेक्टरी स्ट्रक्चर काढून टाकायचे असेल तर, प्रत्येक डिरेक्टरी, त्याची सबडिरेक्टरीज इत्यादीची कॅन्डेट प्रथम काढून टाकण्याचा प्रयत्न करण्याऐवजी त्याच्या -r पर्यायासह rm वापरणे सर्वात कार्यक्षम आहे.

rmmdir ने rm सह सामायिक केलेले तीन पर्याय आहेत -v (म्हणजे, व्हर्बोज), जे काय होत आहे याबद्दल अतिरिक्त इनफार्मेशन प्रदान करते, --help, जे rmmdir बद्दल बेसिक डॉक्युमेंट्स प्रदान करते आणि --version, जे rmmdir ची आवृत्ती सांगते जी सध्या आहे वापर rmmdir च्या विविध आवृत्त्यांमध्ये काही फरक आहेत, त्यामुळे स्पेसिफिक सिस्टिम साठी कागदपत्रे वाचणे नेहमीच शहाणपणाचे असते.

### cat कमांडसह फाइल्सची लिस्ट आणि संयोजन

cat (concatenate file) कमांड तुमच्या स्क्रीनवर फाइल्सची कंटेंट पाठवण्यासाठी वापरली जाते. या आदेशाचा वापर फाइल्समधील टेक्स्ट इतर फाइल्समध्ये पाठवण्यासाठी देखील केला जाऊ शकतो. Hour 6 मध्ये स्टॅंडर्ड इनपुट, स्टॅंडर्ड आउटपुट आणि रीडायरेक्शन यासारख्या संज्ञा समाविष्ट आहेत आणि हा विपार्ट तुम्हाला या कमांडसाठी काही बेसिक यूजेस दाखवतो.

जरी लहान फाइल्स वाचण्यासाठी cat सबयोगी असू शकते, परंतु ती सहसा फायली एकत्र करणे, तयार करणे, ओवरराइडिंग करणे किंवा जोडण्यासाठी वापरली जाते. एक लहान फाइल पाहण्यासाठी cat वापरण्यासाठी, आपण प्रविष्ट करू शकता

#### \$ cat test.txt

ही टेक्स्ट फाइल cat कमांडने तयार केली होती.

cat जगातील सर्वात सोपा टेक्स्ट एडिटर असू शकते.

आपण हे पुस्तक वाचल्यास, आपण cat कसे वापरावे ते शिकाल.

या फाइलमधील टेक्स्ट ची ही शेवटची ओळ आहे.

कॅट कमांडमध्ये अनेक पर्याय आहेत. तुम्हाला तुमची फाइल ओळ क्रमांकांसह पहायची असल्यास, कदाचित स्पेसिफिक वाक्यांश लक्षात ठेवण्यासाठी, तुम्ही -n पर्याय वापरू शकता:

#### \$ cat -n test.txt

- ही टेक्स्ट फाइल cat कमांडद्वारे तयार करण्यात आली होती.
- cat जगातील सर्वात सोपा टेक्स्ट एडिटर असू शकते.
- तुम्ही हे पुस्तक वाचल्यास, तुम्ही cat कसे वापरावे ते शिकाल.
- या फाइलमधील टेक्स्ट ची ही शेवटची ओळ आहे.

आणि एकाच वेळी अनेक फाइल्स पाहण्यासाठी cat वापरा, कारण cat वाइल्डकार्ड स्वीकारते, उदाहरणार्थ:

#### \$ cat -n test\*

- ही टेक्स्ट फाइल cat कमांडद्वारे तयार करण्यात आली होती.
- cat जगातील सर्वात सोपा टेक्स्ट एडिटर असू शकते.
- तुम्ही हे पुस्तक वाचल्यास, तुम्ही cat कसे वापरावे ते शिकाल.
- या फाइलमधील टेक्स्ट ची ही शेवटची ओळ आहे.
- test2.txt ची ही पहिली ओळ आहे.

- ही फाइल देखील मांजरीने तयार केली होती.
- test2.txt ची ही शेवटची ओळ आहे.

जसे आपण पाहू शकता, मांजरीने त्याच्या आउटपुटमध्ये दुसरी फाइल देखील समाविष्ट केली आहे, आणि प्रत्येक फाइल नव्हे तर आउटपुटच्या प्रत्येक ओळीला क्रमांक दिलेला आहे. लक्षात ठेवा की यासह दोन्ही फाइल्स देखील पहा

#### \$ cat test.txt test2.txt

आउटपुट वाइल्डकार्ड वापरल्याप्रमाणेच असेल. परंतु अनेक फाइल्स पाहणे म्हणजे cat वापरण्याचा एकच पाथ आहे. फाइल्स एकत्र करण्यासाठी तुम्ही रीडायरेक्शन ऑपरेटर > सह cat कमांड देखील वापरू शकता. उदाहरणार्थ, तुम्हाला test3.txt नावाच्या तिसऱ्या फाइलमध्ये test.txt आणि test2.txt एकत्र करायचे असल्यास, तुम्ही वापरू शकता

#### \$ cat test\* > test3.txt

सह निकाल तपासा

#### # ls -l test\*

या प्रकरणात, तुम्ही test.txt ची कंटेंट test2.txt मध्ये जायची आहे की test2.txt ची कंटेंट test.txt मध्ये जायची आहे हे युजर प्रथम ठरवेल. नंतर, >> पुनर्निर्देशन ऑपरेटरसह cat वापरून, तुम्ही टाइप करू शकता

#### \$ cat test.txt >> test2.txt

हे test2.txt च्या शेवटी test.txt ची कंटेंट जोडते. रिझल्ट तपासण्यासाठी, पुन्हा cat वापरा:

#### \$ cat test2.txt

test2.txt ची ही पहिली ओळ आहे.

ही फाइलही cat नेच तयार केली होती.

test2.txt ची ही शेवटची ओळ आहे.

ही टेक्स्ट फाइल cat कमांडने तयार केली होती.

cat जगातील सर्वात सोपा टेक्स्ट एडिटर असू शकते.

आपण हे पुस्तक वाचल्यास, आपण cat कसे वापरावे ते शिकाल.

या फाइलमधील टेक्स्ट ची ही शेवटची ओळ आहे.

जर तुम्ही कमांड एंटर केली असेल तर लक्षात ठेवा.

#### \$ cat -n test.txt >> test2.txt

test2.txt फाइल अशी दिसेल

#### \$ cat test2.txt

test2.txt ची ही पहिली ओळ आहे.

ही फाइलही cat नेच तयार केली होती.

test2.txt ची ही शेवटची ओळ आहे.

- ही टेक्स्ट फाइल cat कमांडद्वारे तयार करण्यात आली होती.
- cat जगातील सर्वात सोपा टेक्स्ट एडिटर असू शकते.
- तुम्ही हे पुस्तक वाचल्यास, तुम्ही cat कसे वापरावे ते शिकाल.
- या फाईलमधील टेक्स्ट ची ही शेवटची ओळ आहे.

शेवटी, तुम्हाला वर्ड प्रोसेसर किंवा टेक्स्ट एडिटर न चालवता एक लहान टेक्स्ट फाइल तयार करायची असल्यास तुम्ही वापरू शकता अशी युक्ती येथे आहे. कारण cat कमांड स्टँडर्ड इनपुट वाचू शकते (याबद्दल अधिक तास 6 मध्ये), तुम्ही cat कमांडला फाइल तयार करण्यास आणि तुमच्या कीस्ट्रोकसह भरा.

**कसे ते येथे आहे:**

**\$ cat > myfile.txt**

आता, काही टेक्स्ट प्रविष्ट करा:

**\$ cat > myfile.txt**

हा कॅट वर्ड प्रोसेसर आहे.

हा फाईलचा शेवट आहे.

त्यानंतर, तुमचे टायपिंग पूर्ण झाल्यावर, फाइल बंद करण्यासाठी Ctrl+D दाबा. हे कार्य करते की नाही हे पाहण्यासाठी, प्रयत्न करा

**\$ ls -l myfile.txt**

-rw-rw-r-- 1 bball bball 61 Nov 12 18:26 myfile.txt

**\$ cat myfile.txt**

हा कॅट वर्ड प्रोसेसर आहे.

हा फाईलचा शेवट आहे.

यूजर ला हे देखील माहित असले पाहिजे की cat कमांड कोणत्याही फाईलची कंटेंट प्रिंटेड करेल, केवळ टेक्स्ट फायलीच नाही. जरी एक किंवा अनेक लहान फाईल्स पाहण्यासाठी cat सबयुक्त असू शकते,

सिस्टम date वेळ प्रदर्शित आणि सेट करण्यासाठी date कमांड उदाहरणे date कमांड अनेक फॉर्मॅटमध्ये date प्रदर्शित करण्यासाठी सबयुक्त आहे. हे तुम्हाला सिस्टमची date आणि वेळ सेट करण्यास देखील अनुमती देते.

येथे व्यावहारिक उदाहरणांसह date कमांड कशी वापरायची याची काही उदाहरणे आहेत.

कोणत्याही पर्यायांशिवाय date कमांड कार्यान्वित केल्यावर, ते खाली दर्शविल्याप्रमाणे करंट date आणि वेळ प्रदर्शित करेल.

**\$ date**

Mon 20 May 22:02:24 PDT 2013

## 1 date पर्याय वापरून स्ट्रिंग व्हॅल्यूमधून date प्रदर्शित करा

जर तुमच्याकडे स्ट्रिंगमध्ये स्थिर date किंवा वेळ व्हॅल्यू असेल, तर तुम्ही खाली दाखवल्याप्रमाणे इनपुट स्ट्रिंगला डेट फॉर्मॅटमध्ये रूपांतरित करण्यासाठी -d किंवा -date पर्याय वापरू शकता.

कृपया लक्षात घ्या की हे करंट date आणि वेळ व्हॅल्यू वापरत नाही. त्याऐवजी तुम्ही स्ट्रिंग म्हणून पास केलेले date आणि वेळ व्हॅल्यू वापरते.

खालील उदाहरणे इनपुट date फक्त स्ट्रिंग घेतात आणि आउटपुट date स्वरूपात दाखवतात. तुम्ही वेळ स्पेसिफाइड न केल्यास, ते वेळेसाठी 00:00:00 वापरते.

**\$ date --date="12/2/2014"**

Tue 2 Dec 00:00:00 PST 2014

**\$ date --date="2 Feb 2014"**

Sun 2 Feb 00:00:00 PST 2014

**\$ date --date="Feb 2 2014"**

Sun 2 Feb 00:00:00 PST 2014

खालील उदाहरण इनपुट date आणि वेळ स्ट्रिंग घेते आणि date स्वरूपात आउटपुट प्रदर्शित करते.

**\$ date --date="Feb 2 2014 13:12:10"**

Sun 2 Feb 13:12:10 PST 2014

## 2 -file पर्याय वापरून फाईलमधील date नमुने वाचा

हे आम्ही वर चर्चा केलेल्या -d किंवा -date पर्यायांसारखे आहे. परंतु, तुम्ही ते अनेक तारखेसाठी करू शकता. तुमच्याकडे विविध स्थिर तारखा असलेली फाइल असल्यास, तुम्ही खाली दाखवल्याप्रमाणे -f किंवा -file पर्याय वापरू शकता.

या उदाहरणात आपण पाहू शकतो की datefile मध्ये 2 date strings आहेत. डेटफाइलची प्रत्येक ओळ date कमांडद्वारे पार्स केली जाते आणि प्रत्येक ओळीसाठी date आउटपुट केली जाते.

**\$ cat datefile**

9 Sept 1986

२३ Aug 1987

**\$ date --file=datefile**

Tue 9 Sep 00:00:00 PDT 1986

Sun 23 Aug 00:00:00 PDT 1987

## 3 -date पर्याय वापरून संबंधित date मिळवा

रिलेटिव्ह व्हॅल्यू वापरून भविष्यातील date मिळविण्यासाठी तुम्ही date कमांड देखील वापरू शकता. उदाहरणार्थ, पुढील उदाहरणांना पुढील सोमवारची date मिळते.



**\$ date --date="next mon"**

Mon 27 May 00:00:00 PDT 2013

जर string=@ date कमांडला दिले असेल, तर date कमांड युगापासून (1970-01-01 UTC) सेकंदाला तारखेत रूपांतरित करा.

हे 1970-01-01 UTC च्या युगापासून 5 सेकंद गेलेली date प्रदर्शित करते:

**\$ date --date=@5**

Wed 31 Dec 16:00:05 PST 1969

हे 1970-01-01 UTC च्या युगापासून 10 सेकंद गेलेली date प्रदर्शित करते:

**\$ Date --Date=@10**

Wed 31 Dec 16:00:10 PST 1969

हे 1970-01-01 UTC च्या युगापासून 1 मिनिट (म्हणजे 60 सेकंद) निघून गेलेली date प्रदर्शित करते:

**\$ Date --date=@60**

Wed 31 Dec 16:01:00 PST 1969

4 मागील Date प्रदर्शित करा

-date कमांड वापरून तुम्ही मागील date प्रदर्शित करू शकता. खाली काही शक्यता दाखवल्या आहेत.

**\$ date --date='3 seconds ago'**

Mon May 20 21:59:20 PDT 2013

**\$ date --date="1 Day ago"**

Sun 19 मे 21:59:36 PDT 2013

**\$ date --date="yesterday"**

Sun 19 मे 22:00:26 PDT 2013

**\$ date --date="1 month ago"**

शनि 20 एप्रिल 21:59:58 PDT 2013

**\$ date --date="1 year ago"**

Sun 20 May 22:00:09 PDT 2012

5 -set पर्याय वापरून date आणि वेळ सेट करा

तुम्ही खाली दाखवल्याप्रमाणे -s किंवा -set पर्याय वापरून तुमच्या सिस्टमची date आणि वेळ सेट करू शकता..

या उदाहरणात, सुरुवातीला ते 20:09:31 म्हणून वेळ प्रदर्शित करते. त्यानंतर आम्ही ती 21:00:00 मध्ये बदलण्यासाठी date कमांड वापरली.

**\$ date**

Sun 20 May 20:09:31 PDT 2013

**\$ date -s "Sun May २० 21:00:00 PDT 2013"**

Sun 20 May 21:00:00 PDT 2013

**\$ date**

Sun 20 मे 21:00:05 PDT 2013

**6 -u पर्याय वापरून सार्वत्रिक वेळ प्रदर्शित करा**

तुम्ही खाली दाखवल्याप्रमाणे -u, किंवा -utc, किंवा - युनिव्हर्सल पर्याय वापरून UTC फॉर्मॅटमध्ये date प्रदर्शित करू शकता.

**\$ date**

Mon 20 May 22:07:53 PDT 2013

**\$date -u**

Tue 21 May 05:07:55 UTC 2013

**7 -r पर्याय वापरून अंतिम बदल वेळ प्रदर्शित करा**

या उदाहरणात, करंट वेळ 20:25:48 आहे

**\$ date**

Sun 20 May 20:25:48 PDT 2013

टच कमांड वापरून datefile चा टाइमस्टॅम्प बदलला आहे. हे वरील तारखेच्या काही सेकंदांनंतर केले गेले

कमांडचे आउटपुट.

**\$ touch datefile**

वरील टच कमांड नंतरची करंट वेळ 20:26:12 आहे

**\$ date**

Sun 20 May 20:26:12 PDT 2013

शेवटी, खाली दर्शविल्याप्रमाणे फाईलचा शेवटचा सुधारित टाइमस्टॅम्प प्रदर्शित करण्यासाठी date कमांड -r पर्याय वापरा. या उदाहरणात, ते डेटफाइलची शेवटची सुधारित वेळ 20:25:57 दर्शवते. हे कुठेतरी 20:25:48 आणि 20:26:12 च्या दरम्यान आहे (जे जेव्हा आपण टाइमस्टॅम्पमध्ये बदल करण्यासाठी वरील टच कमांड कार्यान्वित करतो).

**\$ date -r datefile**

Sun 20 May 20:25:57 PDT 2013

**8 विविध date कमांड फॉर्मॅट**

तुम्ही खालील वाक्यरचना वापरून date कमांड प्रदर्शित करण्यासाठी फॉर्मॅटिंग पर्याय वापरू शकता:

**\$ date +%<format-option>**

कमांड	डिस्क्रिप्शन
• apropos whatis	स्ट्रिंगशी संबंधित आज्ञा दाखवा. थ्रेडसेफ देखील पहा
• man -t ascii   ps2pdf - > ascii.pdf	मॅन्युअल पेजची पीडीएफ बनवा
कोणती आज्ञा	आदेशाचे संपूर्ण पाथ नाव दर्शवा
time कमांड	कमांडला किती वेळ लागतो ते पहा
• time cat	स्टॉपवॉच सुरू करा. थांबण्यासाठी Ctrl-d. हे देखील पहा
<b>dir नेव्हिगेशन</b>	
• cd -	मागील डिरेक्टरी वर जा
• cd	\$HOME डिरेक्टरी वर जा
(cd dir && कमांड)	dir वर जा, कमांड कार्यान्वित करा आणि चालू dir वर परत या
• pushd.	स्टॅकवर करंट dir ठेवा जेणेकरून आपण हे करू शकता popd त्याच्याकडे परत
<b>फाइल सर्च</b>	
• alias l='ls -l --color=auto'	quick dir लिस्ट .   देखील पहा
• ls -lrt	तारखेनुसार फायलींची यादी करा. नवीनतम देखील पहा आणि_mm_yyyy शोधा
• ls /usr/bin   pr -T9 -W\$COLUMNS	टर्मिनलच्या रुंदीपर्यंत 9 स्तंभांमध्ये प्रिंट करा
find -name '*.ch]'   xargs grep -E 'expr'	या dir मध्ये आणि खाली 'expr' शोधा. रेपो देखील पहा
find -type f -print0   xargs -r0 grep -F 'उदाहरण'	या dir मध्ये आणि खाली 'example' साठी सर्व नियमित फाईल्स शोधा
find -maxdepth 1 -type f   xargs grep - F'उदाहरण'	या dir मध्ये 'example' साठी सर्व नियमित फाईल्स शोधा
find -maxdepth 1 -type d   dir वाचताना; doecho \$dir; echo cmd2; पूर्ण	प्रत्येक आयटमवर प्रक्रिया करा मल्टिपल कमांड (वेळात पळवाट)
• find -type f ! -perm -444	सर्वानी वाचता येत नसलेल्या फायली शोधा (वेबसाइटसाठी सबयुक्त)
• find -type d ! -perm -111	सर्वांद्वारे एक्सेस करण्यायोग्य नसलेल्या dirs शोधा (वेबसाइटसाठी सबयुक्त)
• find -r file[^\]*\.txt'	नावांसाठी कॅशड इंडेक्स शोधा. हे पुन्हा ग्लोब *फाइल*.txt सारखे आहे
• रेफरेन्स पहा (look reference)	सबसर्गासाठी झटपट शोधा (क्रमबद्ध) डिक्शनरी
• grep---color reference /usr/share/dict/words	च्या इवेंट्स हायलाइट करा मध्ये नियमित अभिव्यक्ती डिक्शनरी
<b>संग्रहण आणि कम्प्रेसन (archives and compression)</b>	
gzip -c file	फाइल एन्क्रिप्ट करा
gzip file.gzip	फाइल डिक्लिप्ट करा
tar -c dir/   bzip2 > dir.tar.bz2	dir/ चे कॉम्प्रेस संग्रहण करा
bzip2 -dc dir.tar.bz2   tar -x	संग्रहण काढा (tar.gz फायलींसाठी bzip2 ऐवजी gzip वापरा)

tar -c dir/   gzip   gpg -c   ssh user@remote 'ddof=dir.tar.gz.gpg'	रिमोटमशीनवर dir/ चे कूटबद्ध संग्रहण करा
dir/ -name '*.txt'   serch xargs cp -a -- target-	directory=dir_txt/ --parents dir/ आणि खालील सबसंचाची प्रत तयार करा
(tar -c /dir/to/copy )   ( cd /where /to/ && tar -x-p)	कॉपी (परवानगीसह) कॉपी/dir/where/to/dir वर
(cd /dir/to/copy && tar -c.)   ( cd /where /to /&& tar -x -p	कॉपी/दीरची कंटेंट /कुठे/ते/वर कॉपी करा (परवानग्यांसह)
(tar -c /dir/to/copy )   ssh -C user@remote 'cd	
/where/to/&& tar -x -p'	कॉपी (परवानग्यांसह) कॉपी/ dir toremote:/where/to/ dir
dd bs=1M if=/dev/sda   gzip   ssh user@remote'dd of=sda.gz'	रिमोट मशीनवर हार्डडिस्कचा बॅकअप घ्या
<b>rsync (नेटवर्क कार्यक्षम फाइल कॉपीअर: टेस्टिंग साठी --dry-run पर्याय वापरा)</b>	
rsync -P rsync://rsync.server.com/path/to/filefile	फक्त फरक मिळवा. त्रासदायक डाउनलोडसाठी अनेक वेळा करा
rsync --bwlimit=1000 tofile tofile	दर मर्यादितसह लोकल कॉपी करा. हे I/O साठी छान आहे
rsync -az -e ssh --delete ~/public_html/ remote.com:'~/public_html'	मिरर वेब साइट (कंप्रेशन आणि एन्क्रिप्शन वापरून)
rsync -auz -e ssh remote:/dir/&& rsync -auz -e ssh.रिमोट:/dir/	रिमोटसह करंट डिरेक्टरी सिंक्रोनाइझ करा
<b>ssh (सुरक्षित शेल)</b>	
ssh \$USER@\$HOST कमांड	\$HOST वर \$USER म्हणून कमांड चालवा (डिफॉल्ट कमांड=शेल)
• ssh -f -Y \$USER@\$HOSTNAME xeyes	\$HOSTNAME वर GUI कमांड म्हणून चालवा \$USER
scp -p -r \$USER@\$HOST: फाइल dir/	\$HOST वर \$USER च्या होमडिरेक्टरीमध्ये परवानगीसह कॉपी करा
scp -c arcfour \$USER@\$LANHOST: bigfile	लोकल LAN साठी जलद क्रिप्टो वापरा. हे GigE संतृप्त करू शकते
ssh -g -L 8080:localhost:80 root@\$HOST	कडे कनेक्शन फॉरवर्ड करा \$HOSTNAME:8080 ते \$HOST:80
ssh -R 1434:imap:143 root@\$HOST	\$HOST:1434 वरून imap:143 मध्ये कनेक्शन फॉरवर्ड करा
ssh-copy-id \$USER@\$HOST	पासवर्ड-लेस लॉग इन करण्यासाठी \$USER@\$HOST साठी सार्वजनिक की इंस्टाल करा
<b>नेटवर्किंग (note ifconfig, route, mii-tool, nslookup कमांड अप्रचलित आहेत)</b>	
ethtool eth0	इथरनेट इंटरफेस eth0 ची स्थिती दर्शवा
ethtool -- बदला eth0 autoneg बंद गती 100duplex full	इथरनेट इंटरफेस गती व्यक्तिचलितपणे सेट करा
iw dev wlan0 लिंक	वायरलेस इंटरफेस 0 ची लिंक स्थिती दर्शवा
iw dev wlan0 set bitrates legacy-2.4 1	वायरलेस इंटरफेस गती स्वहस्ते सेट करा
• iw dev wlan0 scan	रेन्ज तील वायरलेस नेटवर्कची यादी करा
• ip link show	नेटवर्क इंटरफेस लिस्ट बद्ध करा
ip link set dev eth0 name wan	इंटरफेस eth0 चे नाव बदलून wan करा
ip lonk dev eth0 सेट करा	इंटरफेस eth0 वर (किंवा खाली) आणा
• ip addr शोshow	इंटरफेससाठी एड्रेस लिस्ट बद्ध करा

• ip addr 1.2.3.4/24 brd + dev eth0 ऑड करा	ऑड करा (किंवा डेल) ip आणि मुखवटा(255.255.255.0)
ip path show	राउटिंग टेबलची यादी करा
• ip root 1.2.3.254 द्वारे डीफॉल्ट ऑड करा	डीफॉल्ट गेटवे 1.2.3.254 वर सेट करा
• ss -tupl	सिस्टमवर इंटरनेट सर्विस ची यादी करा
• ss -tup	सिस्टीमवर/वरून ऍक्टिव्ह कनेक्शन लिस्ट बद्ध करा
• pixelbeat.org होस्ट करा	नाव किंवा उलट साठी DNS ip एड्रेस शोधा
• hostname -i	लोकल आयपी एड्रेस शोधा (होस्ट 'होस्टनाव' समतुल्य)
whois pixelbeat.org	होस्टनाव किंवा ipaddress साठी whois इनफार्मेशन पहा
<b>विंडोज नेटवर्किंग (नोट सांबा हे पॅकेज आहे जे हे सर्व विंडो स्पेसिफिक नेटवर्किंग सपोर्ट प्रदान करते)</b>	
• smbtree	विंडोज मशीन शोधा. findsmb देखील पहा
nmblookup -A 1.2.3.4	विंडो शोधा (नेटबायोस) ip शी संबंधित नाव एड्रेस
smbclient -L windows_box	विंडोज मशीन किंवा सांबा सर्व्हरवर शेअर्सची यादी करा
mount -t smbfs -o fmask=666,अतिथी // windows_box/share /mnt/share	विंडोज शेअर माउंट करा
इको 'संदेश'   smbclient -M windows_box	विंडोज मशीनवर पॉपअप पाठवा (XP sp2 मध्ये बायडिफॉल्ट बंद)
<b>टेक्स्ट हाताळणी(टीप sed stdin आणि stdout वापरते. नवीन आवृत्त्या -i पर्यायासह इनप्लेस संपादनास सपोर्ट देतात)</b>	
sed 's/string1/string2/g'	string1 ला string2 ने बदला
sed 's/(.*)1\12/g'	anystring1 to anystring2 मध्ये बदल करा
sed '/^ *#/d; /^ *\$/d'	कमेंट आणि रिक्त ओळी काढा
sed ':a; \\\\$/N; s\\\n//; ta'	अनुगामी रेषा एकत्र करा \
sed 's/[ \t]*\$//'	ओळीमधून मागची जागा काढा
sed 's/(["\$\\])\1/g'	एस्केप शेल मेटाकॅरेक्टर्स डबल कोट्समध्ये ऍक्टिव्ह आहेत
• seq 10   sed "s/^ / /; s/ *\{7,\}\1/"	नंबर उजवीकडे एलाइनमेंट करा
• seq 10   sed p   paste-	एक कॉलम डुप्लिकेट करा
sed -n '1000{p;q}'	1000 वी ओळ प्रिंटेड करा
sed -n '10,20p;20q'	10 ते 20 ओळी प्रिंटेड करा
sed -n 's/.*<title>\(.*\)</title>.*\1/ip;T;q'	HTML वेब पृष्ठावरून टाईटल काढा
sed -i 42d ~/.ssh/known_hosts	एक स्पेसिफिक ओळ डिलेट करा
sort -t. -k1,1n -k2,2n -k3,3n -k4,4n	IPV4 आयपी एड्रेस क्रमवारी (सॉर्टिंग) लावा
• echo test   tr '[:lower:]' '[:upper:]'	केस रूपांतरण
• tr -dc '[:print:]' < /dev/urandom	प्रिंट करण्यायोग्य नसलेले कॅरक्टर फिल्टर करा
• tr -s '[:blank:]' '\t' </proc/diskstats   कट - f4	रिक्त जागांद्वारे विभक्त फील्ड कट करा

• history   wc -l	ओळी मोजा
• seq 10   paste -s -d "	एकाशी जोडणे आणि लाइन आयटम वेगळे करा
<b>ऑपरेशन्स सेट करा (लक्षात ठेवा तुम्ही वेगासाठी LANG=C निर्यात करू शकता. तसेच या फाइलमध्ये डुप्लिकेट रेषा गृहीत धरत नाहीत)</b>	
sort file1 file2   uniq	क्रमवारी (सॉर्टिंग) न लावलेल्या फायलींचे संघ
sort file1 file2   uniq -d	क्रमवारी (सॉर्टिंग) न लावलेल्या फायलींचे छेदनपॉइंट
sort file1 file1 file2   uniq -u	क्रमवारी (सॉर्टिंग) न लावलेल्या फायलींचा फरक
sort file1 file2   uniq -u	क्रमवारी (सॉर्टिंग) न लावलेल्या फायलींचा सममितीय फरक
join -t\0' -a1 -a2 file1 file2	क्रमवारी (सॉर्टिंग) केलेल्या फायलींचे संघटन
join -t\0' file1 file2	क्रमवारी (सॉर्टिंग) केलेल्या फायलींचा छेदनपॉइंट
join -t\0' -v2 file1 file2	क्रमवारी (सॉर्टिंग) लावलेल्या फाइल्समधील फरक
join -t\0' -v1 -v2 file1 file2	क्रमवारी (सॉर्टिंग) लावलेल्या फायलींचा सममितीय फरक
<b>math</b>	
• echo '(1 + sqrt(5))/2'   bc -l	
• seq -f '4/%g' 1 2 99999   पेस्ट -sd-+   bc -l	द्रुत गणित ( $\phi$ गणना करा). बीसी देखील पहा
• echo 'pad=20; min=64; (100*10^6)/((pad+min)*8)'   bc	युनिक्स मेथड्स ने $\phi$ ची गणना करा
• echo 'pad=20; min=64; print (100E6)/((pad+min)*8)'   python	अधिक जटिल (int) उदा. हे maxFastE पॅकेट दर दर्शविते
• echo 'pad=20; plot [64:1518] (100*10**6)/((pad+x)*8)'   gnuplot -persist	पायथन वैज्ञानिक नोटेशन हाताळते
• echo 'obase=16; ibase=10; 64206'   bc	प्लॉट फास्टई पॅकेट दर वि पॅकेट आकार
• echo \$(0x2dec)	बेस रूपांतरण (दशांश ते हेक्साडेसिमल)
• units -t '100m/9.58s' 'miles/hour'	बेस रूपांतरण (हेक्स ते डिसेंबर) ((शेल अर्थमेटिक एक्सपांशन ))
• units -t '500GB' 'GiB'	युनिट रूपांतरण (मेट्रिक ते इम्पीरियल)
• units -t '1 googol'	युनिट रूपांतरण (SI ते IEC सबसर्ग)
• seq 100   paste -s -d+   bc	व्याख्या सर्व नंबर चा एक कॉलम ऑड करा. Addand Funcpy देखील पहा
<b>कॅलेंडर</b>	
• cal -3	एक कॅलेंडर प्रदर्शित करा
• cal ९ १७५२	स्पेसिफिक महिन्याच्या वर्षासाठी कॅलेंडर प्रदर्शित करा
• date -d fri	आज शुक्रवारी कोणती date आहे. दिवस देखील पहा
• [ \$(date -d '12:00 today +1 day' +%d) = '01' ]    exit	महिन्याचा शेवटचा दिवस असल्याशिवाय स्क्रिप्टमधून बाहेर पडा
• date --date='25 Dec' +%A	या वर्षी ख्रिसमस कोणत्या दिवशी येतो
• date --date='@2147483647'	युग (1970-01-01 UTC) पासून आजपर्यंतचे सेकंद रूपांतरित करा
• TZ='America/Los_Angeles' date	यूएसच्या पश्चिम किनाऱ्यावर किती वाजले आहेत (TZ सर्व प्यासाठी निवडा)

• date --date='TZ="America/Los_Angeles" 09:00 next Fri'	अमेरिकेच्या पश्चिम किनाऱ्यावर पुढील शुक्रवारी सकाळी ९ ची लोकल वेळ किती आहे
<b>लोकल</b>	
• printf "%d\n" १२३४	लोकेलसाठी योग्य हजारो गटांसह अंक प्रिंटेड करा
• BLOCK_SIZE=\`1 ls -l	ls मध्ये लोकेल हजारो ग्रुपिंग वापरा. सी देखील एल
• echo "I live in `locale territory`"	लोकल डेटाबेसमधून इनफार्मेशन काढा
• LANG=en_IE.utf8 locale int_prefix	स्पेसिफिक देशासाठी लोकल इनफार्मेशन पहा. सीसीओड देखील पहा
• locale -kc \$(locale   sed -n 's/(LC_\\{4,\\})=.*\^1/p')   less	लोकल डेटाबेसमध्ये सबलब्ध फील्डची यादी करा
<b>रीकोड (नापसलेले iconv, dos2unix, unix2dos)</b>	
• recode -l   less	सबलब्ध रूपांतरणे दाखवा (प्रत्येक ओळीचे सबनाव)
recode windows-1252.. file_to_change.txt	Windows "ansi" ते लोकल कॅरक्टर सेट (ऑटोमॅटिक CRLF रूपांतरण)
recode utf-8/CRLF.. file_to_change.txt	Windows utf8 ते लोकल कॅरक्टर सेट
recode iso-8859-15..utf8 file_to_change.txt	लॅटिन9 (पश्चिम युरोप) ते utf8
recode ../b64 < file.txt > file.b64	बेस 64 एन्कोड
recode /qp.. < file.qp > file.txt	उद्दत प्रिंट योग्य डीकोड
recode ../HTML < file.txt > file.html	HTML वर टेक्स्ट
• recode -lf windows-1252   grep euro	कॅरक्टर ची टेबल पहा
• echo -n 0x80   recode latin-9/x1..dump	लॅटिन-9 charmap मध्ये कोड काय दर्शवतो ते दर्शवा
• echo -n 0x20AC   recode ucs 2/x2..latin-9/x	लॅटिन-9 एन्कोडिंग दाखवा
• echo -n 0x20AC   recode ucs-2/x2..utf 8/x	utf-8 एन्कोडिंग दाखवा
<b>सीडी</b>	
gzip < /dev/cdrom > cdrom.iso.gz	डेटा cdrom ची प्रत सेव्ह करा
mkisofs -V LABEL -r dir   gzip > cdrom.iso.gz	dir च्या कंटेंट वरून cdrom प्रतिमा तयार करा
mount -o loop cdrom.iso /mnt/dir	cdrom प्रतिमा /mnt/dir वर माउंट करा (फक्त वाचण्यासाठी)
wodim dev=/dev/cdrom blank=fast	CDRW क्लियर करा
gzip -dc cdrom.iso.gz   wodim -tao	Cdrom प्रतिमा बर्न करा (वापर करा --
dev=/dev/cdrom -v -data -	dev कन्फर्म करण्यासाठी prcap)
cdparanoia -B	CD पासून wav फाइल्सवर चालू dir ऑडिओ ट्रॅक रिप करा
wodim -v dev=/dev/cdrom -audio -pad *.wav	currentdir मधील सर्व wavs मधून ऑडिओ सीडी बनवा (cdrdao देखील पहा)
oggenc --tracknum=\$track track.cdda.wav -otrack.ogg	wav फाइलमधून ogg फाइल बनवा

डिस्क जागा	
• ls -lSr	आकारानुसार फाइल्स दाखवा, सर्वात मोठे शेवटचे
• du -s *   sort -k1,1rn   head	करंट dir मध्ये शीर्ष डिस्क वापरकर्ते दर्शवा. डुटॉप देखील पहा
• you -hs /home/*   sort -k1,1h	डिस्क वापराचे स्पष्टीकरण सोपे करून पाथ क्रमवारी (सॉर्टिंग) लावा
• df -h	माउंट केलेल्या फाइलसिस्टमवर मोकळी जागा दाखवा
• df -i	माउंट केलेल्या फाइलसिस्टमवर विनाव्हॅल्यु इनोड दर्शवा
• fdisk -l	डिस्क विभाजनांचे आकार आणि टाइप दर्शवा (root म्हणून चालवा)
• rpm -q -a --qf '%10{SIZE}\t{NAME}\n'   sort -k1,1n	rpm distros वर सर्व पॅकेजेस इंस्टाल आकारानुसार (बाइट्स) लिस्ट बद्ध करा
• dpkg-query -W -f='\${InstalledSize;10}\t\${Package}\n'   sort -k1,1n	इंस्टाल करून सर्व पॅकेजेस लिस्ट बद्ध करा deb distros वर आकार(KBytes).
• dd bs=1 seek=2TB if=/dev/null of=ext3.test	एक मोठी टेस्टिंग फाइल तयार करा (जागा न घेता). ट्रंकेट देखील पहा
• > file	फाईलचा डेटा ट्रंक करा किंवा एक तयार करा रिक्त फाइल
निरीक्षण/डीबगिंग	
• tail -f /var/log/messages	लॉग फाइलमधील संदेशांचे निरीक्षण करा
• strace -c ls >/dev/null	कमांडद्वारे केलेले सारांश/प्रोफाइल सिस्टम कॉल
• strace -f -e उघडा ls >/dev/null	कमांडद्वारे केलेल्या सिस्टम कॉलची यादी करा
• strace -f -e trace=write -e write=1,2 ls >/dev/null	stdout आणिstderr वर काय लिहिले आहे याचे निरीक्षण करा
• ltrace -f -e getenv ls >/dev/null	कमांडद्वारे केलेल्या लायब्ररी कॉलची यादी करा
• lsof -p \$\$	प्रक्रिया आयडी उघडलेले पाथ लिस्ट बद्ध करा
• lsof ~	पॅथोपेन स्पेसिफाइड केलेल्या प्रक्रियांची यादी करा
• tcpdump not port 22	ssh वगळता नेटवर्क रहदारी दर्शवा. tcpdump_not_me देखील पहा
• ps -e -o pid, args --forest	पदानुक्रमात प्रक्रियांची यादी करा
• ps -e -o pcpu,cpu,nice,state,cputime,args --sort pcpu   sed '/^ 0.0 /d'	% cpu वापरानुसार प्रक्रियांची यादी करा
• ps -e -orss=,args=   क्रमवारी (सॉर्टिंग) -b - k1,1n   pr -TW\$COLUMNS	मेम (KB) वापरानुसार प्रक्रियांची यादी करा. ps_mem.py देखील पहा
• ps -C firefox - bin - L - o pid , वेळ , pcpu , राज्य	एका स्पेसिफिक प्रक्रियेसाठी सर्व थ्रेड्सची यादी करा
• ps -p 1,\$\$ -o etime=	स्पेसिफिक प्रक्रिया आयडीसाठी संपलेली भिंत वेळ लिस्ट बद्ध करा
• watch -n.1 pstree -Uacp \$\$	बदलणारी प्रक्रिया सबट्री दाखवा
• last reboot	सिस्टम रीबूट इतिहास दर्शवा
• free -m	(उर्वरित) RAM चे प्रमाण दर्शवा (-एमबी मध्ये प्रदर्शित करते)
• watch -n.1 'cat/proc/interrupts'	सतत बदलणारा डेटा पहा
• udevadm monitor	कॉन्फिगर नियमांना मदत करण्यासाठी udev घटनांचे निरीक्षण करा

सिस्टम इनफार्मेशन	
• <code>uname -a</code>	कर्नल आवृत्ती दर्शवा आणि सिस्टम आर्किटेक्चर
• <code>head -n1 /etc/issue</code>	डिस्ट्रीब्युशन चे नाव आणि आवृत्ती दर्शवा
• <code>cat/proc/partitions</code>	सिस्टमवर नोंदणीकृत सर्व पार्टीशन दाखवा
• <code>grep MemTotal /proc/meminfo</code>	सिस्टमद्वारे एकूण RAM दर्शवा
• <code>grep "मॉडेल नाव" /proc/cpuinfo</code>	CPU(ची) इनफार्मेशन दाखवा
• <code>lspci -tv</code>	PCI इनफार्मेशन दाखवा
• <code>lsusb -tv</code>	USB इनफार्मेशन दाखवा
• <code>mount   column -t</code>	सिस्टमवर माउंट केलेल्या फाइल सिस्टमची यादी करा (आणि आउटपुट एलाइनमेंट करा)
• <code>grep -F capacity: /proc/acpi/battery/BAT0/info</code>	लॅपटॉप बॅटरीमधील सेलची स्थिती दर्शवा
# <code>dmidecode -q   less</code>	SMBIOS/DMI इनफार्मेशन प्रदर्शित करा
# <code>smartctl -A /dev/sda   grep Power_On_Hours</code>	ही डिस्क (सिस्टम) एकूण किती काळ चालू आहे
# <code>hdparm -i /dev/sda</code>	डिस्क sda बदल इनफार्मेशन दाखवा
# <code>hdparm -tT /dev/sda</code>	डिस्क sda वर वाचन गती टेस्टिंग करा
# <code>badblocks -s /dev/sda</code>	डिस्क sda वर न वाचता येणार्या ब्लॉक्ससाठी टेस्टिंग
परस्परसंवादी (interactive)	
• <code>readline</code>	bash, python, bc, gnuplot, ... द्वारे वापरलेले लाइन एडिटर
• <code>screen</code>	विलग करण्याची क्षमता असलेले आभासी टर्मिनल, ...
• <code>mc</code>	शक्तिशाली फाइल व्यवस्थापक जो rpm, tar, ftp, ssh, ... ब्राउझ करू शकतो.
• <code>gnuplot</code>	परस्परसंवादी/स्क्रिप्टेबल ग्राफिंग
• <code>links</code>	अंतर्जाल सर्च क
• <code>xdg-open.</code>	नोंदणीकृत डेस्कटॉप अनुप्रयोगासह फाइल किंवा url उघडा



## एमएस वर्ड 2010 थिअरी (MS WORD 2010 Theory)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एमएस ऑफिस काय आहे ते सांगा
- एमएस वर्ड काय आहे ते थोडक्यात सांगा आणि सुरुवातीच्या पायऱ्या
- MS Word चे विविध स्क्रीन ब्लॉक्स स्पष्ट करा
- डॉक्युमेंट्स तयार करणे, सेव्ह करणे, प्रिंटेड करणे यासाठी प्रक्रिया स्पष्ट करा.

मायक्रोसॉफ्ट ऑफिस हे मायक्रोसॉफ्ट कॉर्पोरेशनने सादर केलेले ऑप्लिकेशन सॉफ्टवेअर पॅकेज आहे. एमएस ऑफिसमध्ये खालील लोकप्रिय पॅकेजेस आहेत:

- मायक्रोसॉफ्ट वर्ड
- मायक्रोसॉफ्ट एक्सेल
- Microsoft PowerPoint
- मायक्रोसॉफ्ट ऍक्सेस
- Microsoft Outlook

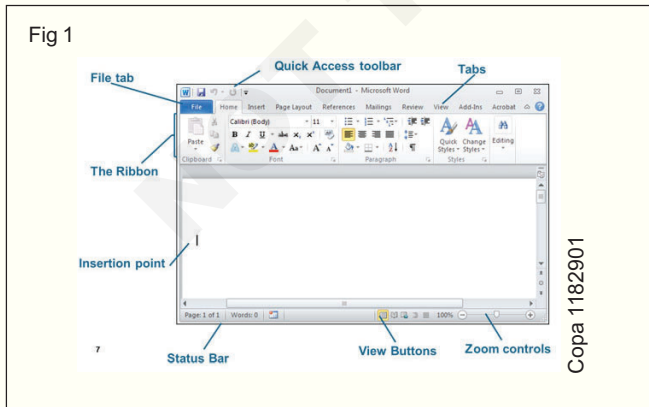
### वर्ड 2010

हे एक वर्ड प्रोसेसर पॅकेज आहे जे डॉक्युमेंट्स तयार आणि एडिट करण्यास मदत करते. हा आजकालचा सर्वात प्रसिद्ध वर्ड प्रोसेसर आहे. डॉक्युमेंट्स , मेमो, पत्रे, रिपोर्ट , इनफार्मेशन पत्रके, व्यवसाय डॉक्युमेंट्स आणि अगदी इंटरनेट वेब पेजेज तयार करण्यासाठी आणि फॉर्मॅटिंग करण्यासाठी साधनांचा सर्वसमावेशक संच प्रदान करून ते व्यावसायिक दिसणारे डॉक्युमेंट्स बनवते.

### वर्ड

Start > (windows Logo) All programs > Microsoft office > Microsoft Word वर क्लिक करा. डीफॉल्ट सेटिंग्ज पेज लेआउटसह डीफॉल्ट रिक्त डॉक्युमेंट्स सह एमएस वर्ड उघडतो.

### डीफॉल्ट पेज लेआउट (पिक्चर 1)



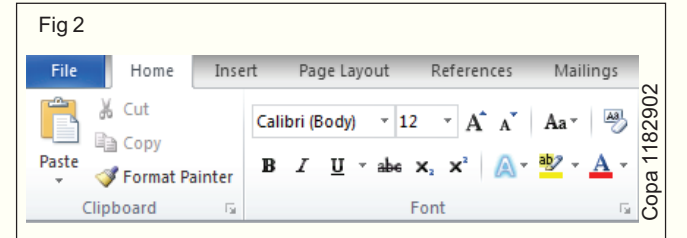
मागील आवृत्तीच्या विपरीत, MS Office 2010 मध्ये सर्व एलिमेंटसाठी वैशिष्ट्यांचा सामान्य संच आहे. हे समजणे सोपे करते आणि वर्ड, एक्सेल, पॉवरपॉईंट, ऍक्सेस इ.सह एमएस ऑफिसच्या सर्व पॅकेजेसवर सामान्य

सबयोगिता सबलब्ध राहते. इतर पॅकेजच्या वळवलेल्या वापरकर्त्यांना परवडेल म्हणून अनेक वैशिष्ट्ये पुन्हा डिझाइन केली आहेत. Word 2010 मधील या सुधारणा आणि सबयुक्तता खाली डिस्क्रिप्शन केल्या आहेत.

Word च्या रिबन टॅबमध्ये आठ प्रमुख पार्ट आहेत उदा. फाइल, मुख्यपृष्ठ, इन्सर्ट , पृष्ठ सिन्टेक्स , रेफरन्स , मेलिंग, रिव्यू आणि पहा.

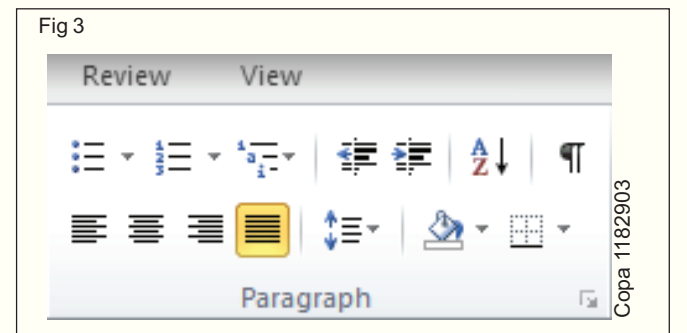
रिबनचा फाइल टॅब आवश्यकतेनुसार नवीन फाइल, रिक्त किंवा टेम्पलेट पृष्ठ तयार करण्यासाठी सबयुक्त आहे. Ctrl + N नेहमी ऑफिस पॅकेजमध्ये नवीन रिक्त डॉक्युमेंट्स तयार करा. असे तयार केलेले डॉक्युमेंट्स डीफॉल्ट शब्द डॉक्युमेंट्स म्हणून सेव्ह केले जाऊ शकतात किंवा सुसंगत फॉर्मेट दिले जाऊ शकतात. शिवाय, डॉक्युमेंट्स मध्ये एक्सेस करण्याची परवानगी निश्चित केली जाऊ शकते, नेटवर्कवर डॉक्युमेंट्स सामायिक करू शकता, अगदी मागील आणि एक्सटर्नल आवृत्त्यांशी सुसंगत डॉक्युमेंट्स मॅनेज करू शकता. प्रिंट पर्याय इंस्टाल प्रिंटर वैशिष्ट्यांनुसार डॉक्युमेंट्स हार्ड कॉपी किंवा पीडीएफ फॉर्मेट मिळवू शकतो. नुकत्याच उघडलेल्या फायली देखील पाहिल्या जाऊ शकतात जेणेकरून ते पुन्हा कार्य करणे सोपे होईल. शब्दावर मदत त्याच टॅबमध्ये आहे.

### होम टॅब (पिक्चर 2)



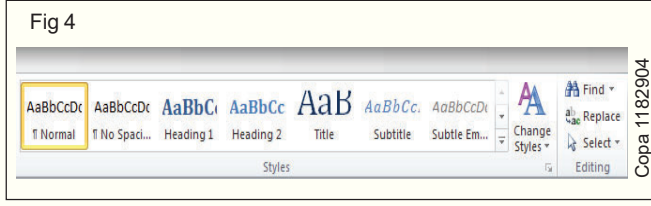
होम टॅबमध्ये कट, कॉपी, पेस्ट आणि पेस्ट स्पेशलसह क्लिपबोर्ड पर्याय आहेत. फॉन्ट ब्लॉकमध्ये फॉन्ट, आकार, सुपरस्क्रिप्ट, सबस्क्रिप्ट, बोल्ड, इटालिक, अधोरेखित, स्ट्राइकथ्रू, फॉन्ट रंग इत्यादी सेट करण्याचे सर्व पर्याय आहेत.

### पैराग्राफ ग्रुप (पिक्चर 3)



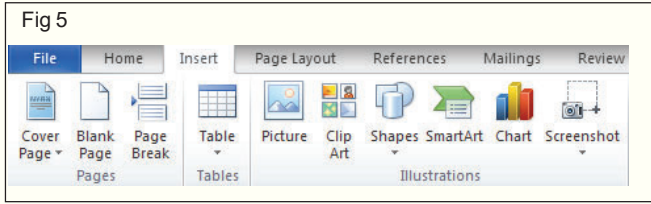
पैराग्राफ ग्रुप डावीकडे, उजवीकडे, मध्यभागी, जस्टिफाय आणि इंडेंटेशन, पॅरा आणि लाइन स्पेसिंगसह फॉर्मॅटिंग प्याराग्राफ तयार करतो. रेखा आणि पेज ब्रेक पेज नंबर पर्याय आणि फॉर्मॅटिंग तयार करण्यास अनुमती देतात.

#### स्टाइल आणि एडिटिंग ग्रुप (पिक्चर 4)



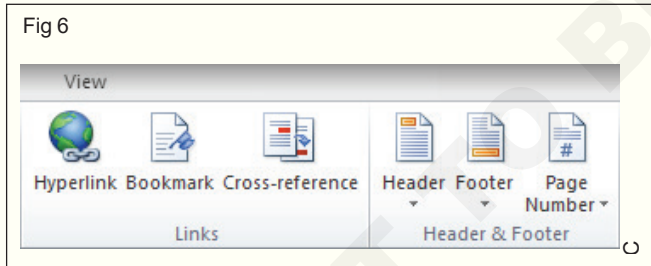
स्टाइल्स ग्रुप हेडिंग स्टाइल्स, पॅराग्राफ स्टाइल्स, सबटायटल्स इ. सारख्या प्रीफॉर्मेट केलेल्या टेक्स्ट शैलींना परवानगी देतो. कस्टम स्टाइल देखील बदललेल्या वैशिष्ट्यांसह स्टोर केल्या जाऊ शकतात. टेक्स्ट तील स्पेसिफाइड एरिया मध्ये स्पेसिफिक एरिया निवडण्यासाठी, सर्च करण्यासाठी आणि/किंवा पर्याय बदलण्यासाठी एडिटिंग ग्रुप वापरला जातो.

इन्सर्ट टॅबमध्ये 7 ग्रुप आहेत. पेजेज, टेबल्स आणि चित्रे ब्लॉक (पिक्चर 5)



पेज समूह कव्हर पेजेज, रिक्त पेजेज समाविष्ट करणे आणि पेज ब्रेक करतो. टेबल्स ग्रुप टेक्स्ट डॉक्युमेंट्स त तयार स्पेसिफाइड रो-कॉलम सेट किंवा कस्टमाइज टेबल फॉर्मॅटसह टेबल घालण्यास मदत करतो. इलस्ट्रेशन ग्रुप एक्सटर्नल स्रोतातील चित्रे, क्लिपआर्ट, आकार, स्मार्ट आर्ट, डेटाचे टेबल्स, अगदी स्क्रीनशॉट टेक्स्ट डॉक्युमेंट्स त समाविष्ट करतो.

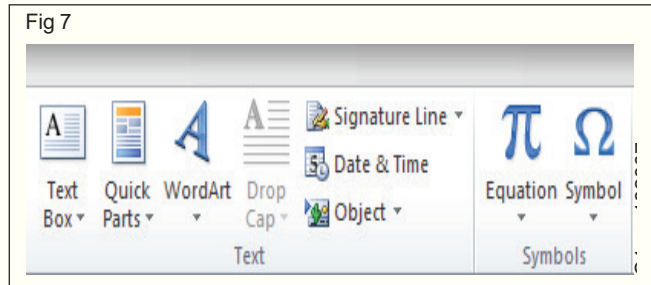
#### लिंक्स आणि हेडर/फुटर ग्रुप (पिक्चर 6)



लिंक्स ग्रुप डॉक्युमेंट्स तील टेक्स्ट, बुकमार्क आणि क्रॉस संदर्भावर हायपरलिंक तयार करतो.

हेडर आणि फुटनोट ग्रुप प्रत्येक पेजवर दिसणारे हेडर, फुटनोट आणि प्लेसमेंट एरिया म्हणून प्रदर्शित करण्यासाठी पृष्ठ क्रमांक समाविष्ट करतो.

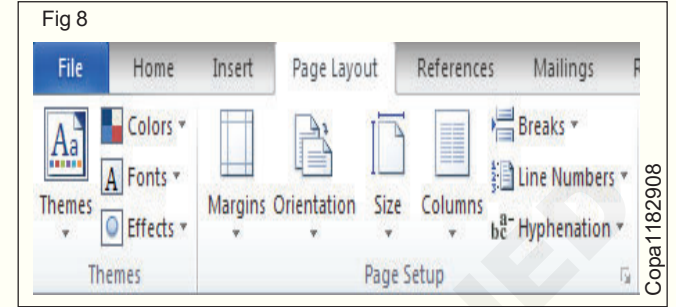
#### टेक्स्ट आणि सिम्बॉल्स ब्लॉक (पिक्चर 7)



टेक्स्ट ग्रुप टेक्स्ट बॉक्स, ब्रोशर तयार करण्याची शक्यता असलेले द्रुत पार्ट, वर्डआर्ट म्हणून डिझाइन केलेला टेक्स्ट, पैराग्राफ स्टाइल, सिगनीचर लाइन, date आणि वेळ आणि ऑब्जेक्ट इन्सर्ट पर्याय तयार करण्यास परवानगी देतो.

सिम्बॉल्स ग्रुप समीकरणांची सिम्बॉल्स समाविष्ट करतो जसे की गणिताची समीकरणे किंवा सिम्बॉल्स जसे की चलन, गणित सिम्बॉल्स इ.

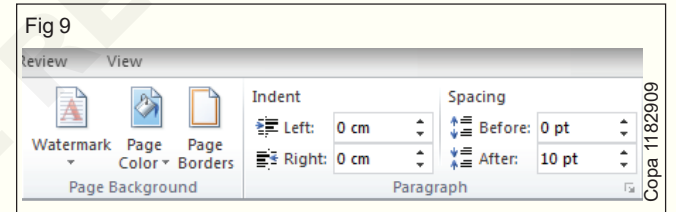
#### थीम आणि पेज सेटअप ग्रुप (पिक्चर 8)



पेज लेआउट टॅबमध्ये पाच प्रमुख ग्रुप आहेत. थीम ब्लॉक डॉक्युमेंट्स वर थीम वापरून पूर्वनिर्धारित टेम्पलेट सेटअप तयार करते. अगदी कस्टमाइज नवीन थीम तयार केल्या जाऊ शकतात आणि भविष्यातील वापरासाठी सेव्ह केल्या जाऊ शकतात.

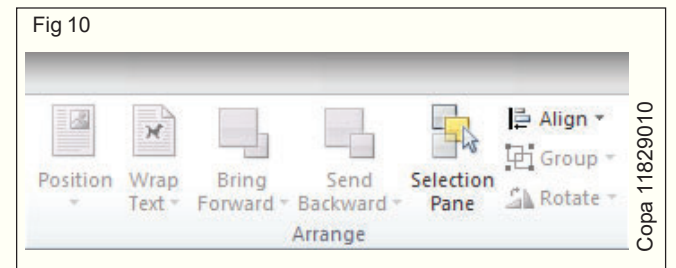
पेज लेआउट समुहात समासाच्या आसपास वैशिष्ट्ये आहेत, पेज अभिमुखता म्हणजे व्हर्टिकल किंवा हॉरिजॉन्टल, कागदाचा आकार, प्रदर्शित करण्यासाठी कॉलम, खंड, रेखा क्रमांक आणि हायफनेशन.

#### पेज बॅकग्राउंड आणि पैराग्राफ ग्रुप (पिक्चर 9)



पृष्ठ बॅकग्राउंड ग्रुप वॉटरमार्क, पेज बॅकग्राउंड रंग आणि पेज बॉर्डर तयार करतो. पॅराग्राफ ब्लॉकमध्ये होम -> पॅराग्राफ ग्रुपमध्ये आधीच चर्चा केलेले पर्याय स्पेसिफाइड केले आहेत.

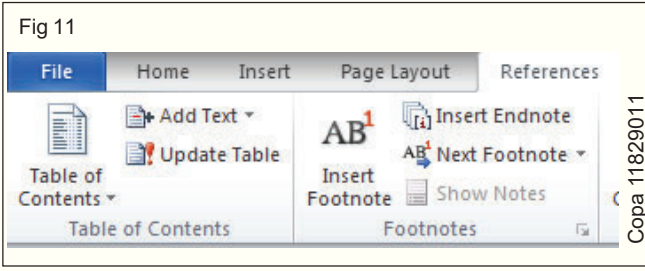
#### ग्रुप अरेंज करा (पिक्चर 10)



अरेंज ग्रुप ऑब्जेक्ट्सची स्थिती, टेक्स्ट एलिमेंट, टेक्स्ट रॅपिंग, ऑब्जेक्ट्सची सिन्टेक्स, ऑब्जेक्ट्सचे एलाइनमेंट, ग्रुप बद्ध आणि परिवर्तने तयार करतो.

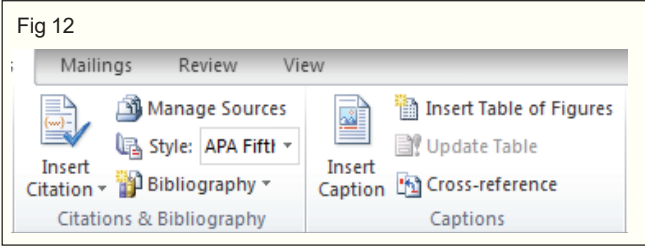
रेफरेन्स टॅबमध्ये सहा ब्लॉक्स आहेत.

#### TOC आणि फुटनोट ग्रुप (पिक्चर 11)



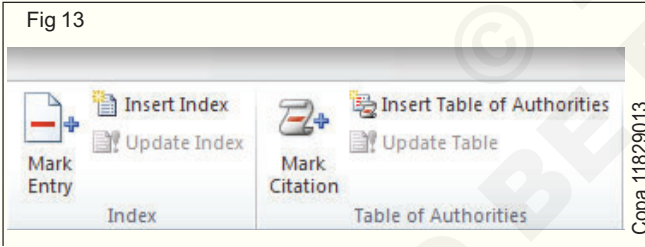
कंटेंट टेबल स्पेसिफिक प्रकाशन डॉक्युमेंट्स ची TOC ऑटोमॅटिक पणे आणि सुधारणांनंतर अपडेट करण्यायोग्य बनवते. फुटनोट टॅब पृष्ठाच्या फुटनोट तयार करते, स्पष्टीकरणात्मक वाक्ये किंवा रेफरेन्स समाविष्ट करते. एंडनोट डॉक्युमेंट्स च्या शेवटी टीप बनवते.

### प्रशस्तिपत्रे, रेफरेन्स ग्रंथ आणि कॅप्शन्स ग्रुप (पिक्चर 12)



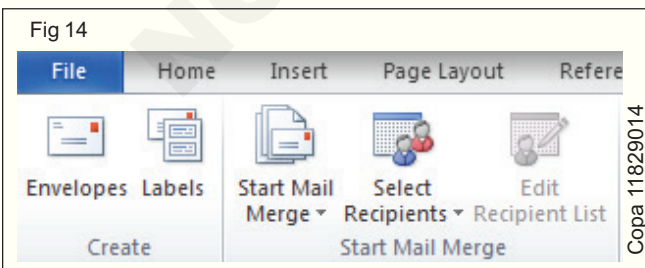
प्रशस्तिपत्रे आणि ग्रंथलिस्ट हे प्रकाशनाशी संबंधित लेखकांचे साइटेशन, रेफरेन्स टेबल आणि क्रेडिट्स म्हणून एक्सटर्नल सोर्स बनवते. कॅप्शन्स टॅब टेक्स्ट च्या आत एक कॅप्शन, आकृत्यांची टेबल, क्रॉस रेफरेन्स समाविष्ट करतो.

### ऑथरिटी गटाची इंडेक्स आणि टेबल (पिक्चर 13)



निर्देशांक मुख्य विषयावर किंवा सबविषयावर व्हॅल्यू प्रविष्ट करतो आणि त्यानुसार निर्देशांक तयार करतो. प्राधिकरणाच्या टेबल मध्ये साइटेशन चिन्ह आहे ज्यामध्ये डॉक्युमेंट्स त नमूद केलेल्या प्रकरणे, कायदे आणि इतर प्राधिकरणांची यादी समाविष्ट आहे.

### क्रिएट आणि मेल मर्ज करा (पिक्चर 14)

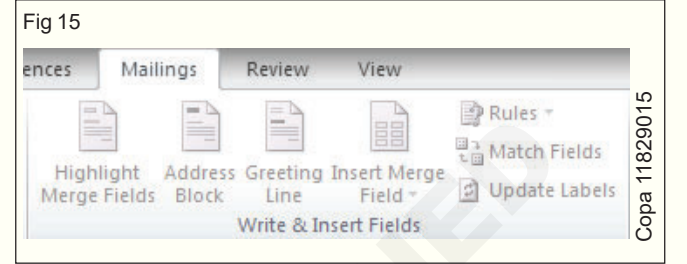


मेलिंग टॅब सामान्य मेलिंग वैशिष्ट्यांचा वापर करून पत्र किंवा ऑर्डरचे मेलिंग तयार करण्यासाठी वापरले जाते. क्रिएट ब्लॉक पूर्वनिर्धारित फॉर्मेट तसेच कस्टम फॉर्मेट वापरून लिफाफे प्रिंट करते. लेबल देखील प्रिंटेड

केले जाऊ शकतात ज्यावर वितरण एड्रेस नमूद केला आहे.

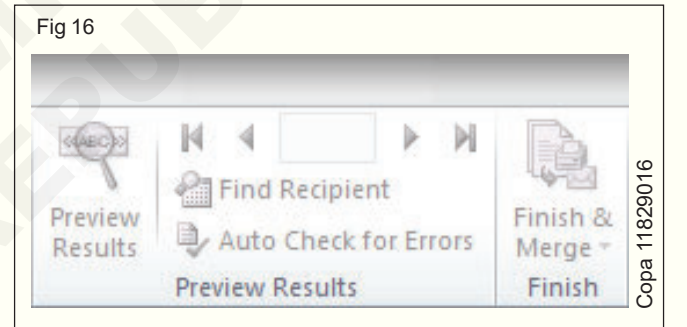
पत्र, ईमेलचे मेल मर्ज तयार करण्यासाठी मेल मर्ज ब्लॉकचा वापर केला जातो. जर एखादे सामान्य पत्र तयार केले असेल आणि अनेक प्राप्तकर्त्यांना पाठवायचे असेल तर, हा पर्याय एड्रेस ब्लॉक तयार करण्यास परवानगी देतो जिथे प्राप्तकर्त्यांची इनफार्मेशन दिसली पाहिजे. अगदी इलेक्ट्रिक प्राप्तकर्ते देखील लिस्ट बद्ध केले जाऊ शकतात तसेच नवीन प्राप्तकर्त्यांची यादी तयार केली जाऊ शकते.

### लिहा आणि इन्सर्ट फील्ड ग्रुप (पिक्चर 15)



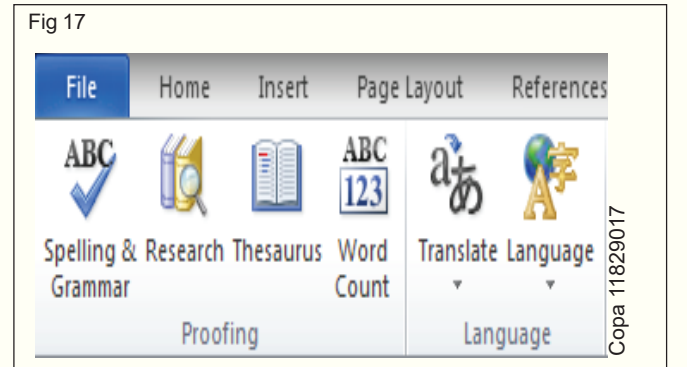
पत्र लिहिताना, त्यात अनेक पार्ट असतात, जे या मेल मर्ज मुळे अॅड्रेस ब्लॉक, ग्रीटिंग लाइन, कंटेंटचे विलीन केलेले फील्ड, लेबल्स इत्यादी फील्ड तयार होतात.

### प्रीव्यू रिजल्ट्स आणि फिनिष ग्रुप (पिक्चर 16)



मेल मर्ज यशस्वीरित्या पूर्ण झाल्यावर, त्याचे प्रीव्यू केले जाऊ शकते आणि प्रत्येक प्राप्तकर्त्यासाठी त्रुटी तपासल्या जाऊ शकतात आणि त्यानुसार एडिट करा. प्रमाणीकरणानंतर मेल मर्ज पूर्ण होते.

### प्रूफिंग आणि लॅंग्वेज ग्रुप (पिक्चर 17)



येथे रिव्यू टॅब विविध डॉक्युमेंट्स क्षेत्रांचे रिव्यू करण्यासाठी वापरले जाते. प्रूफिंगमुळे डॉक्युमेंट्स चे स्पेलिंग आणि व्याकरण तपासणे सोपे होते. येथे अपघाती चुका दुरुस्त केल्या जाऊ शकतात. रिसर्च न केवळ डॉक्युमेंट्स च्या आतच नव्हे तर संदर्भित डॉक्युमेंट्स देखील सर्च ते. थिसॉर्स इंग्रजी

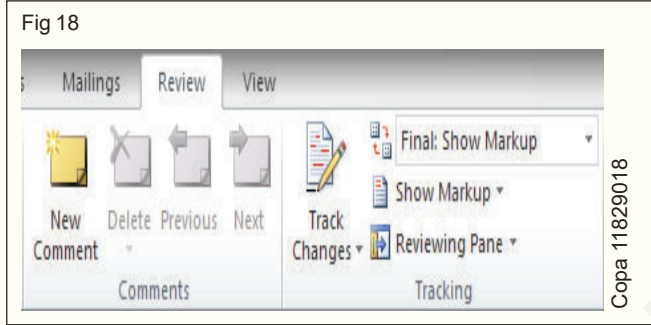
यूके किंवा इंग्रजी यूएसए इत्यादी प्रदेशाच्या स्टॅण्डर्ड भाषा सुसंगततेचा वापर करून डॉक्युमेंट्स ची पूर्णता ओळखतो. शब्द नंबर डॉक्युमेंट्स चे विश्लेषण एकूण शब्द, एकूण कॅरक्टर, रिक्त स्थानांसह आणि न करता, क्र. प्यारोग्राफ, ओळी इ. येथे अशा विश्लेषणासाठी टेक्स्ट ब्लॉक देखील समाविष्ट केले जाऊ शकतात.

भाषा ब्लॉक इंस्टाल केलेल्या इतर भाषांमध्ये पृष्ठाचे भाषांतर करण्यास आणि प्रूफिंग भाषा बदलण्यास मदत करते. यासाठी मायक्रोसॉफ्ट ऑनलाइनकडून थेट अनुवादक सर्विस आवश्यक आहे.

### कमेंट आणि ट्रॅकिंग ग्रुप (पिक्चर 18)

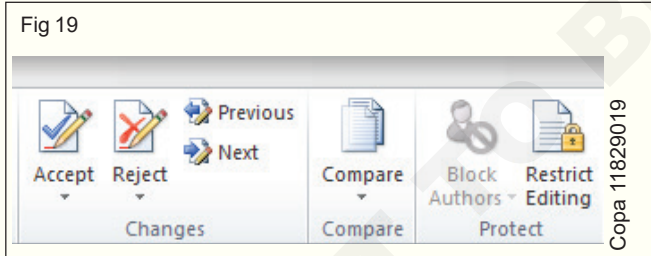
कमेंट ग्रुप स्पेसिफिक प्यारोग्राफ किंवा टेक्स्ट ब्लॉकसाठी कमेंट समाविष्ट करते.

ट्रॅकिंग ग्रुप संरक्षित मोडमध्ये इतर लेखकांद्वारे डॉक्युमेंट्स वर केलेले बदल सर्च तो. उदाहरणार्थ, जर एखादा डॉक्युमेंट्स युजर X द्वारे तयार केला गेला असेल आणि युजर ने Y द्वारे एडिट केला असेल तर, ट्रॅक चिन्हांसह स्वतंत्रपणे ट्रॅक केला जाईल.



### बदल, कम्प्यारीजन करा आणि ग्रुप संरक्षित करा (पिक्चर 19)

त्यामुळे कागदपत्रांमध्ये केलेले बदल स्वीकारले किंवा नाकारले जाऊ शकतात. कम्प्यारीजन वापरून समानतेसाठी डॉक्युमेंट्स पेक्षा अधिक कम्प्यारीजन केली जाऊ शकते. तसेच डॉक्युमेंट्स इतर लेखकांद्वारे एडिट करण्यापासून संरक्षित केले जाऊ शकते.



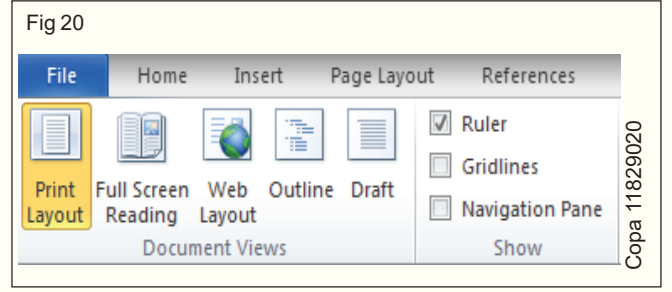
## फाइल तयार करणे, सेव्ह करणे आणि इतर पर्याय (Creating a file, save and other options)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- नवीन डॉक्युमेंट्स कसा तयार करायचा, सेव्ह आणि प्रिंट कसा करायचा ते स्पष्ट करा
- टेक्स्ट आणि डॉक्युमेंट्स स्टाइल एडिट, फॉर्मॅटिंग कसे करावे
- डेटासह शब्द डॉक्युमेंट्स च्या आत टेबल चा संक्षिप्त वापर
- डॉक्युमेंट्स त स्टाइल कशी तयार करावी आणि भविष्यातील वापरासाठी सेव्ह कसे करावे हे स्पष्ट करा
- Word 2010 ची काही अद्वितीय वैशिष्ट्ये स्पष्ट करा
- मेल मर्ज प्रक्रिया थोडक्यात सांगा

फाइल इनफार्मेशन व्ह (पिक्चर 1): Word 2010 नवीन डॉक्युमेंट्स नेहमीप्रमाणे Ctrl + N किंवा फाइल मेनू नवीन पर्यायाद्वारे तयार केला जाऊ

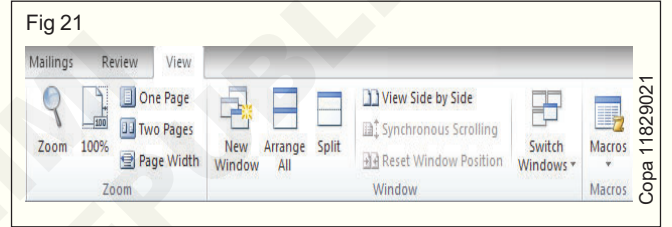
### डॉक्युमेंट्स व्ह आणि शो ग्रुप (पिक्चर 20)



व्ह टॅब वर्ड डॉक्युमेंट प्रदर्शित करण्याचा पाथ दाखवतो. डॉक्युमेंट्स दृश्यांमध्ये प्रिंट लेआउट आहे, शब्दाचे सामान्य व्ह, पूर्ण स्क्रीन वाचन, टॅब आणि रिबन्स अव्ह होण्यासाठी आणि वाचण्यास सोपे, वेब लेआउट, जे html सुसंगत दृश्याचे प्रीव्यू करते, एक्सटर्नल रेखा, डॉक्युमेंट्स ची मूळ आवृत्ती केवळ टेक्स्ट त पाहते टेक्स्ट एडिटिंग मोडसाठी मोड आणि मसुदा मोड.

नियम, ग्रिडलाइन आणि नेव्हिगेशन पॅन युजर च्या पसंतीनुसार दर्शविले किंवा लपवले जाऊ शकतात.

### झूम / विंडो / मॅक्रो ग्रुप (पिक्चर 21)



तसेच पृष्ठाचे व्ह पूर्ण पेजवर झूम केले जाऊ शकते, दोन पेजचे व्ह, 100% डॉक्युमेंट्स आणि कस्टम व्ह. डॉक्युमेंट्स चे एडिटिंग सोपे करण्यासाठी विंडोचे दोन पार्ट केले जाऊ शकतात, डॉक्युमेंट्स कापण्यासाठी पेस्ट करण्यासाठी नवीन विंडो तयार केली जाऊ शकते आणि सर्व ओपन वर्ड डॉक्युमेंट्स पाहण्याची व्यवस्था केली जाऊ शकते. डॉक्युमेंट्स चे फॉर्मॅटिंग करताना, कमांड च्या पुनरावृत्तीसाठी मॅक्रो पर्याय वापरला जाऊ शकतो. हे कमांड्सचा संच स्टोर करण्यासाठी रेकॉर्ड पर्याय वापरते आणि एका क्लिकवर डॉक्युमेंट्स च्या इतर भागावर किंवा दुसऱ्या डॉक्युमेंट्स वर पुन्हा ते पुनरावृत्ती करते.

**Word मध्ये सेव्ह पर्याय (पिक्चर 2):** कोणत्याही सुसंगत स्वरूपातील किंवा ऑफिसच्या जुन्या आवृत्त्यांमध्ये, 2003 किंवा पूर्वीच्या आवृत्त्यांमध्ये सेव्ह म्हणून पर्याय वापरून ते सेव्ह केले जाऊ शकते. वर्ड सॉफ्टवेअरची मुख्य सबयुक्तता म्हणजे वर्ड प्रोसेसिंग डॉक्युमेंट्स तयार करणे. हे प्रकाशन, पत्र, ब्रोशर इ. सारखे कोणतेही फॉर्मॅट असू शकते. टेक्स्ट आधारित प्रेसेंटेशन डिझाइन करण्यासाठी शब्द सर्व टाइप च्या फॉर्मॅट नास सपोर्ट देतो. तसेच हे उद्योग मानकांनुसार प्रमुख स्वीकृत स्वरूपातील आउटपुट फाइल्सना सपोर्ट देते.

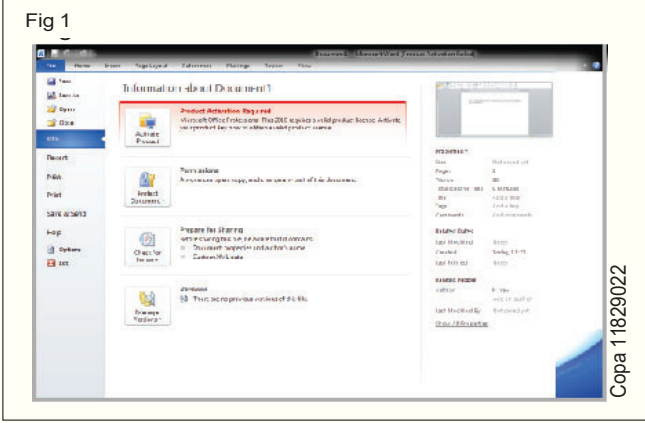


Fig 1  
Copa 11829022

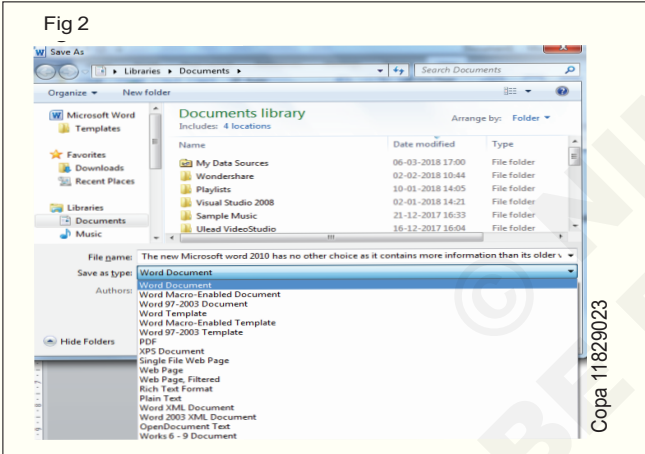


Fig 2  
Copa 11829023

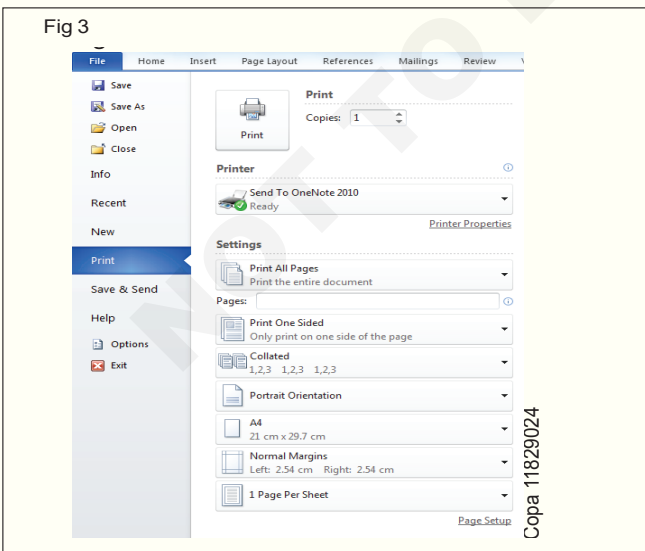


Fig 3  
Copa 11829024

**Word प्रिंट पर्याय (पिक्चर 3):** सेव्ह केलेले डॉक्युमेंट्स फाइल मेनू -> प्रिंट पर्याय वापरून प्रिंटेड केले जाऊ शकतात आणि इंस्टाल प्रिंटर सपोर्ट डॉक्युमेंट्स प्रिंटेड करणे सोपे करते. Adobe Acrobat Professional

इन्स्टॉल केले असल्यास, समान डॉक्युमेंट्स शेअरिंगच्या उद्देशाने PDF फाइल म्हणून स्टोर केले जाऊ शकते. प्रिंटिंग आणि स्टोरेज व्यतिरिक्त, तयार केलेली फाइल वेब पृष्ठ किंवा टेम्पलेट म्हणून वेबवर प्रकाशित केली जाऊ शकते जी भविष्यातील प्रकाशनांसाठी समान स्वरूपात वापरली जाऊ शकते.

वर्ड वापरताना लक्षात घ्यायच्या बेसिक संकल्पना म्हणजे टेक्स्ट प्रॉपर्टीज . यात फॉन्ट टाइप , फॉन्ट आकार, टेक्स्ट रंग आणि टेक्स्ट ची नेहमीची सजावट समाविष्ट आहे. प्याराग्राफ स्टाइल तयार करणे देखील लक्षात ठेवले पाहिजे. प्याराग्राफ सरिखनामध्ये डावीकडे, उजवीकडे, मध्यभागी आणि न्याय्य सेटिंग्ज आहेत. टेक्स्ट एलिमेंटमध्ये ऑर्डर केलेली लिस्ट , अक्रमित लिस्ट , सबविपार्ट लिस्ट यासारख्या आयटम असू शकतात. ते होम टॅबच्या पॅराग्राफ फॉर्मॅटिंग ब्लॉकमध्ये आढळतात. कोट्स तयार करण्यासाठी टेक्स्ट इंडेंट करणे देखील त्याच टॅबमध्ये आहे. पानाच्या आकारानुसार, ओळीतील अंतर आणि प्याराग्राफ अंतर अडजस्ट केले जाऊ शकते, जसे की परिच्छेदाच्या आधी आणि नंतरची जागा, रेषेची उंची, इ. स्टाइल आवश्यक असलेल्या ठिकाणी ठेवण्यासाठी तयार म्हणून वापरण्यासाठी पूर्वनिर्धारित केली जाऊ शकते. स्टॅण्डर्ड टेम्पलेट सबलब्ध आहेत परंतु ते कस्टम स्टाइल देखील तयार करण्यास अनुमती देते.

विविध सबयुक्ततांसाठी टेबल्स घातल्या जाऊ शकतात. सामान्य टेबल मध्ये समायोज्य रुंदी आणि कॉलम असतात त्यामुळे ते थेट Insert > Table पर्यायाने घालता येते.

#### टेबल इन्सर्ट विझार्ड पर्याय (पिक्चर 4)

कस्टम आकाराचे टेबल आवश्यक असल्यास, ते इन्सर्ट टेबल / ड्रॉ टेबल पर्याय वापरून तयार केले जाऊ शकते. इन्सर्ट टेबल कॉलम, रुंदी आणि इतर गुणधर्मांसाठी कस्टम व्हॅल्यू ना अनुमती देते. ड्रॉ टेबल पेन ड्रॉइंग टूल वापरून कस्टम टेबल बनवते ज्याद्वारे सबलब्ध टेक्स्ट कंटेंट नुसार नवीन टेबल काढता येते.

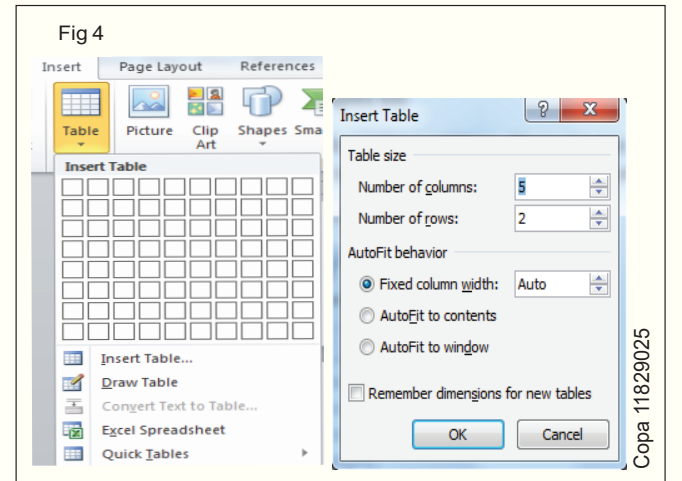
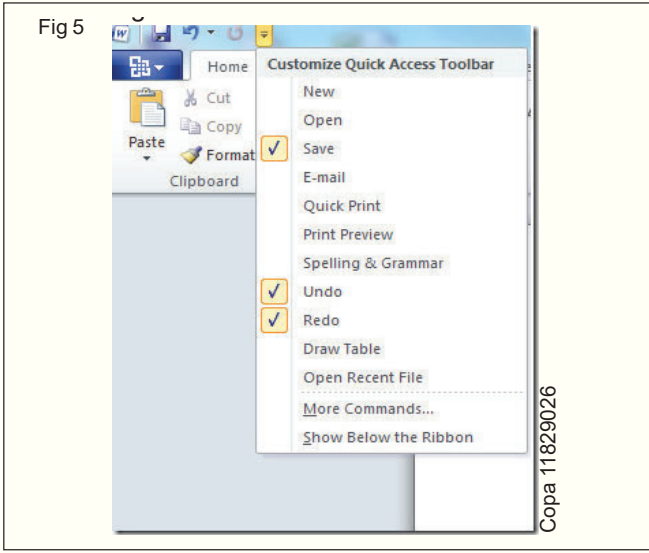


Fig 4  
Copa 11829025

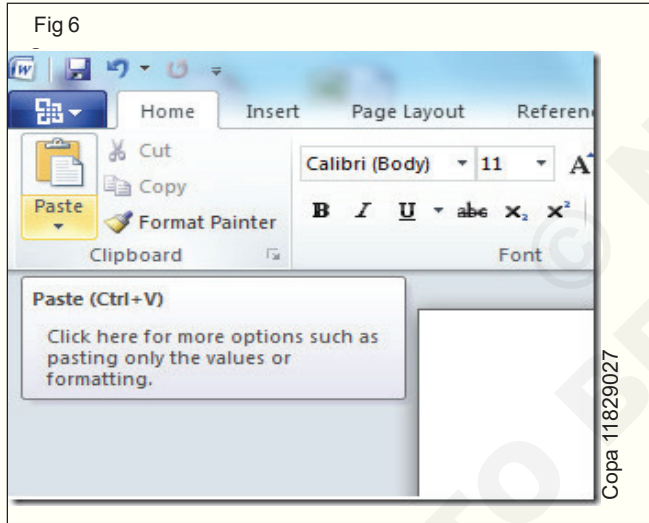
#### कस्टम करण्यायोग्य क्लिक एक्सेस टूल बार

Word 2010 चा Quick Access Toolbar सर्व सामान्यपणे वापरलेले पर्याय दाखवतो. हे ऍप्लिकेशन विंडोच्या वरच्या डाव्या बाजूला, ऑफिस बटणाजवळ स्थित आहे. डीफॉल्टनुसार ते खालील तीन पर्याय प्रदर्शित करते, सेव्ह करा, पूर्ववत करा आणि पुन्हा करा, परंतु ते कस्टम करण्यायोग्य आहे आणि तुम्ही त्यात सहजपणे अधिक पर्याय जोडू शकता. (पिक्चर 5)



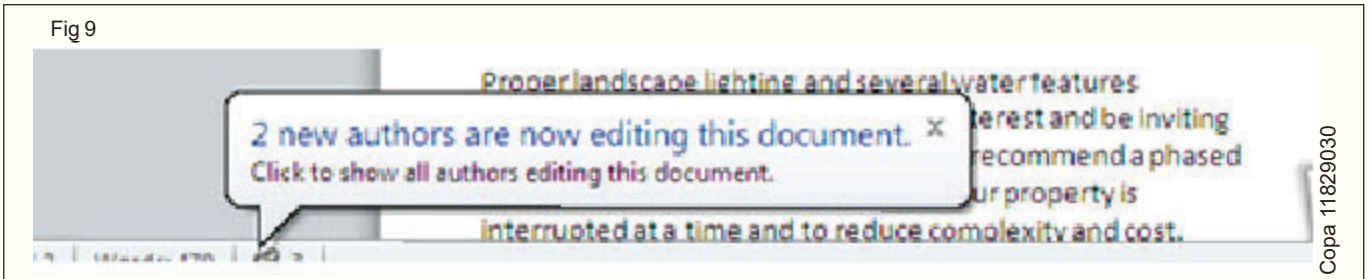
### पेस्ट प्रीव्यू

बहुतेक वापरकर्त्यांसोबत असे घडते की त्यांच्या डॉक्युमेंट्स त काहीतरी कॉपी आणि पेस्ट केल्यानंतर, त्यांना काही बदल पूर्ववत करणे आवश्यक आहे. Word 2010 ने वापरकर्त्यांसाठी हे सोपे केले आहे, आता तुम्ही पेस्ट प्रीव्यू पर्याय वापरून ही अनावश्यक स्टेप दूर करू शकता. हे वापरकर्त्यांना केवळ व्हॅल्यू किंवा फॉर्मॅटिंग पेस्ट करण्यास अनुमती देते. (पिक्चर 6)



### नॅव्हिगेशन पॅन

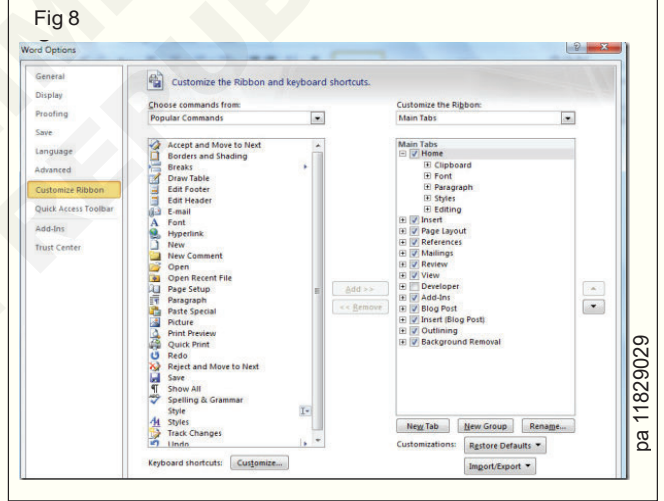
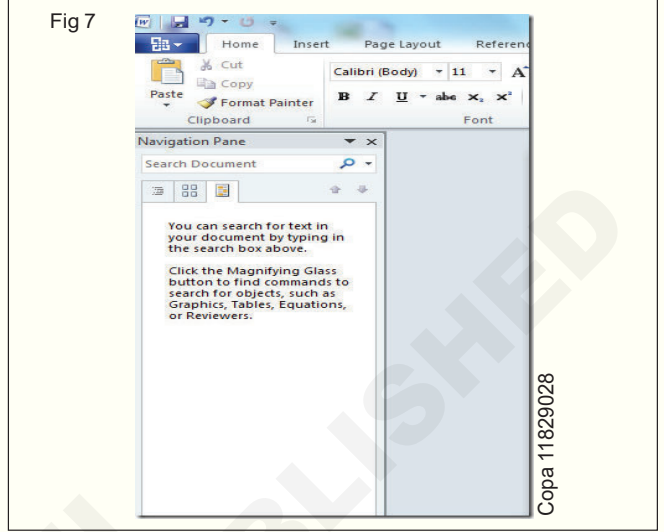
मायक्रोसॉफ्ट ऑफिसच्या मागील आवृत्त्यांमध्ये, डॉक्युमेंट्स तील कोणताही शब्द किंवा वाक्यांश सर्चण्यासाठी एखाद्याला Ctrl+F हॉटकी वापरावी लागते. Word 2010 ने या पर्यायामध्ये एक नवीन जादू जोडली आहे, Ctrl+F आता डॉक्युमेंटच्या डाव्या बाजूला दिसणारा नॅव्हिगेशन पॅन समन्स करतो. तुम्हाला सबलब्ध तीन दृश्ये त्यांच्या संबंधित टॅबवर क्लिक करून, हेडिंग व्ह, थंबनेल पेज व्ह आणि सर्च रिझल्ट व्हवर दिसतील. (पिक्चर 7)



### कस्टम करण्यायोग्य रिबन बटण

वरवर पाहता वर्ड 2010 मधील रिबन बटण वर्ड 2007 मधील एकासारखे दिसते. परंतु त्यात एक मोठी भर आहे, तुम्ही 2010 चे रिबन बटण कस्टमाइज करू शकता. रिबन बटण कस्टमाइज करण्यासाठी खालील पर्यायावर नेव्हिगेट करा Office बटण > शब्द पर्याय > रिबन कस्टमाइज करा. (पिक्चर 8)

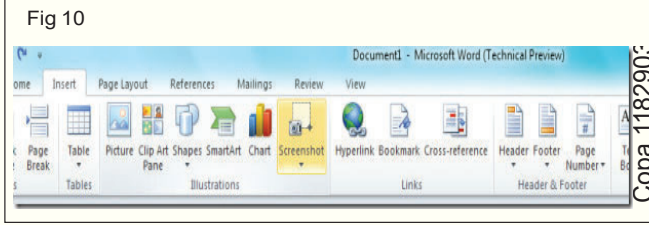
### सुधारित कोलायब्रेशन (Collaborations)



मायक्रोसॉफ्ट वर्ड 2010 मध्ये सह-लेखन नावाचे नवीन वैशिष्ट्य आहे. हे एकाच वेळी एकापेक्षा जास्त लेखकांना डॉक्युमेंट्स एडिट करण्यास अनुमती देते. Word 2010 तुम्हाला सांगतो की किती लेखक डॉक्युमेंट्स एडिट करत आहेत आणि त्यांचे बदल देखील पाहिले जाऊ शकतात. (पिक्चर 9)

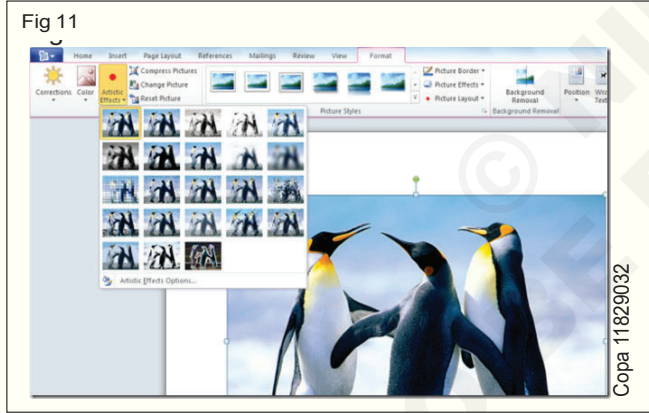
## स्क्रीन कॅप्चर टूल

Word 2010 मध्ये स्क्रीन कॅप्चरिंग नावाचे वैशिष्ट्य समाविष्ट आहे, आता वर्डमध्ये वापरण्यासाठी स्क्रीनशॉट कॅप्चर करण्यासाठी तृतीय पक्ष किंवा अतिरिक्त साधन वापरण्याची आवश्यकता नाही, स्क्रीनचे कोणतेही एरिया कॅप्चर करण्यासाठी फक्त Word 2010 च्या अंगभूत टूलचा वापर करा. खालील पर्याय Insert > Screenshot वर नेव्हिगेट करून स्क्रीनशॉट घेतला जाऊ शकतो. (पिक्चर 10)



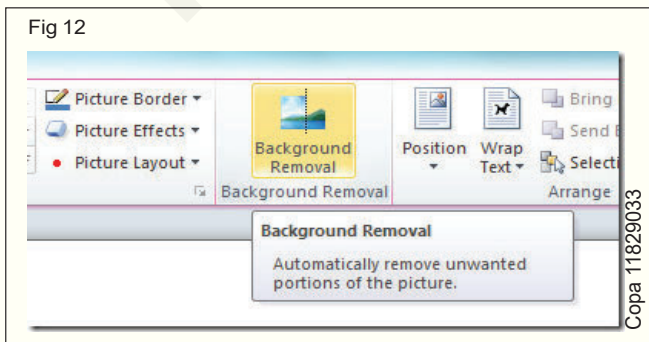
## अर्टिस्टिक इफेक्ट

Word 2010 मध्ये वापरकर्ते आता चित्रांवर अनेक सॅझी कलात्मक इफेक्ट लागू करू शकतात. तुमच्या डॉक्युमेंट्स त कलात्मक इफेक्ट जोडण्यासाठी, खालील पर्यायावर नेव्हिगेट करा Insert > Illustrations > Picture. नंतर तुम्हाला हवे असलेले पिक्चर ब्राउझ करा आणि निवडा, एकदा पिक्चर तुमच्या डॉक्युमेंट्स त जोडले गेले की, पिक्चर टूल्स रेफरन्स टॅब प्रदर्शित होईल आणि तुम्हाला येथे नवीन आर्टिस्टिक इफेक्ट्स ड्रॉप डाउन बटण दिसेल. (पिक्चर 11)



## बॅकग्राउंड काढण्याचा पर्याय

Office 2010 मध्ये Background Removal नावाचा एक अप्रतिम पर्याय आहे, जो कोणत्याही प्रतिमेची बॅकग्राउंड काढून टाकतो. होय, बॅकग्राउंड काढण्यासाठी तुम्हाला फोटोशॉपची आवश्यकता नाही. प्रथम Insert > Picture या पर्यायातून तुमच्या Word डॉक्युमेंटमध्ये पिक्चर इन्सर्ट. नंतर बॅकग्राउंड काढण्याचे साधन शोधा आणि बॅकग्राउंड पासून ओपन व्हा. (पिक्चर 12)



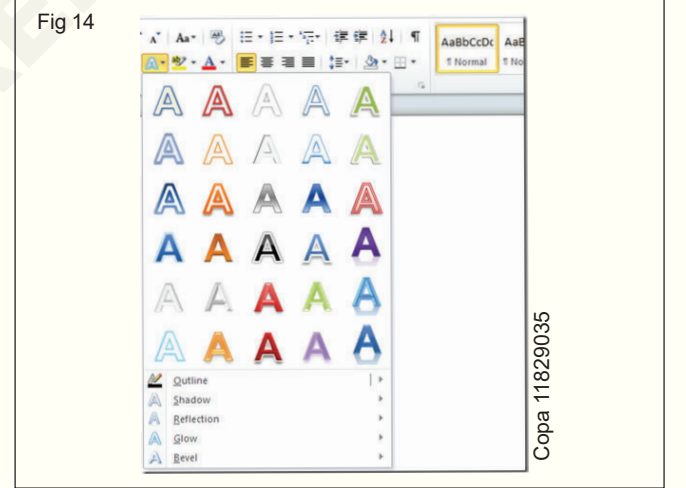
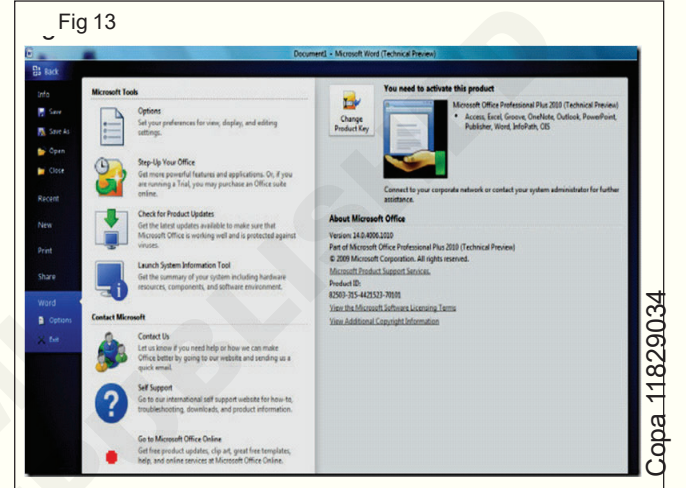
## ऑफिस बॅकस्टेज

ऑफिस बॅकस्टेज ही एक नवीन संकल्पना आहे, ती साध्या जुन्या ऑफिस बटणाचे वर्धित रूप आहे आणि युजर -अनुकूल मेनू प्रदान करते. हे वापरकर्त्यांना डॉक्युमेंट्स, प्रेजेंटेशन किंवा स्प्रेडशीट्स मोठ्या लेयर वर मॅनेज करण्यात मदत करते. (पिक्चर 13)

मनोरंजक वैशिष्ट्य, बरोबर? याबद्दल अधिक वाचा येथे.

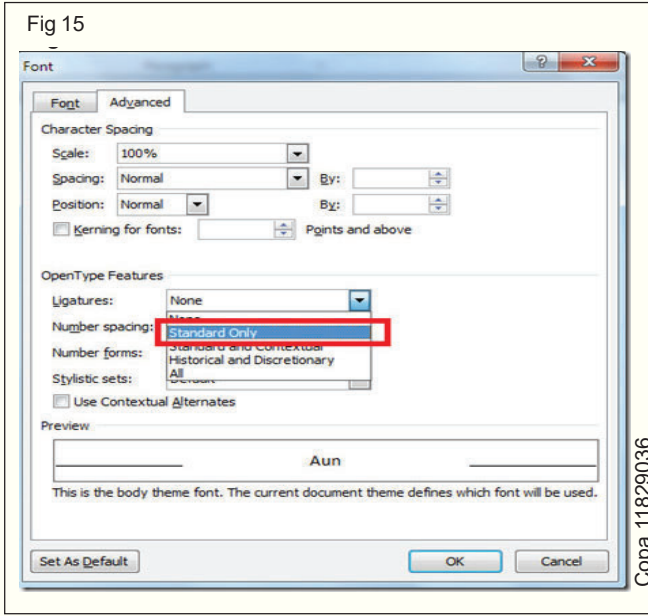
WordArt मध्ये नवीन कला इफेक्ट

इतर वैशिष्ट्यांप्रमाणेच, WordArt नवीन रंगीत कला प्रभावांसह अपडेट केले गेले आहे. टेक्स्ट निवडा, नंतर Word Art वर क्लिक करा आणि सर्व सबलब्ध पर्यायांची लिस्ट प्रदर्शित होईल. (पिक्चर 14)



## Ligatures

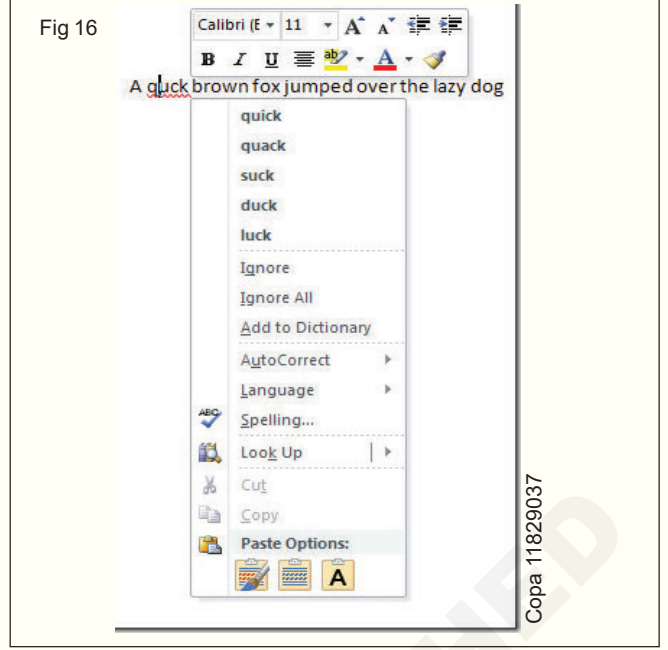
तुम्ही Ligatures बद्दल ऐकले असेल. ते फॉन्ट फॅन्सी दिसायला लावतात आणि ते अक्षरे वेगळे ठेवण्यासाठी देखील वापरले जातात ज्यामुळे तुम्हाला फॉन्ट नियमित असल्याप्रमाणे टेक्स्ट सर्च ता येतो. हे खरे आहे की सर्व फॉन्ट लिगॅचरला सपोर्ट देत नाहीत, परंतु फॉन्टची मोठी विविधता त्यांना सपोर्ट देते. तुम्ही त्यांना फॉन्ट प्राधान्ये > एडवॉन्स मधून सक्षम करू शकता, त्यानंतर लिगॅचर ड्रॉप डाउन बॉक्समध्ये फक्त स्टॅण्डर्ड पर्याय निवडा. (पिक्चर 15)



Copa 11829036

### सुधारित स्पेल चेक

Word 2010 ने त्याच्या शब्दलेखन तपासकामध्ये काही नवीन वैशिष्ट्ये जोडली आहेत, आता ते चूक ओळखेल आणि वाक्य बदलण्याची सूचना करेल.



Copa 11829037

## Word 2010 मधील शॉर्टकट की (Shortcut keys in Word 2010)

उद्दिष्टे: धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- MS Word मधील शॉर्टकट की जाणून घ्या.

CTRL+SHIFT+A	निवडलेल्या मजकुराचे कॅपिटल अक्षरांमध्ये किंवा त्याउलट रूपांतर करते
CTRL+SHIFT+F	फॉन्ट डायलॉग बॉक्स दाखवतो.
CTRL+SHIFT+G	वर्ड काउंट डायलॉग बॉक्स दाखवतो.
CTRL+SHIFT+S	Apply Styles टास्क पेन दाखवते.
ALT+R	रिव्यू टॅब प्रदर्शित करते
ALT+CTRL+1	हेडर 1 लागू करा, त्याचप्रमाणे ALT + CTRL + 2 हे टाईटल 2 लागू करेल
CTRL+SHIFT+L	बुलेट लागू होते
CTRL+SHIFT+F5	बुकमार्क करा
CTRL + B	बोल्ड टेक्स्ट
CTRL + I	इटालिक टेक्स्ट
CTRL + U	टेक्स्ट अधोरेखित करा
CTRL+PAGEDOWN browse	पुढे
CTRL+E	मध्यभागी प्याराग्राफ नेव्हिगेट करा



CTRL+SHIFT+ENTER	कॉलम खंडित
CTRL+SHIFT+C	फॉर्मेट कॉपी करा
ALT+SHIFT+F7	डिक्शनरी
ALT+CTRL+S	डॉक्युमेंट्स स्लिट करते
CTRL+SHIFT+D	दुहेरी अधोरेखित करा
CTRL+END	डॉक्युमेंट्स चा शेवट
END	ओळीचा शेवट
CTRL+SHIFT+P	फॉन्ट आकार निवडा
SHIFT+F5 or ALT+CTRL+Z	मागील स्थितीकडे परत जा
CTRL+SHIFT+.	फॉन्ट वाढवा
CTRL+] ]	फॉन्ट एक पॉइंट वाढवा
ALT+SHIFT+R	हेडर फुटनोट लिंक
CTRL+K	हायपरलिंक
CTRL+M	इंडेंटेशन
CTRL+J	परिच्छेदाचे सपोर्ट करते
ALT+F8	मॅक्रो इन्सर्ट
ALT+SHIFT+K	मेल मर्ज तपासा
F10	मोड मेनू
ALT+F7	पुढच्या चुकीच्या स्पेलिंगकडे जाते
CTRL+H	बदला
CTRL+P	छापा
CTRL+SHIFT+F12	प्रिंट देखील लॉच करते
ALT+SHIFT+BACKSPACE	पुन्हा करा
F12	म्हणून सेव्ह करा
CTRL+SHIFT+K	स्मॉल कॅप्स
CTRL+SHIFT+S	स्टाइल
SHIFT+F7	कोश
ALT+SHIFT+T	वेळ फील्ड
CTRL+SHIFT+M	अनोळखी

## COPA - डॉक्युमेंट्स फॉर्मेट करा

## टेक्स्ट आणि प्याराग्राफ इन्सर्ट , फॉर्मेटिंग करा, डॉक्युमेंट्स विपार्ट तयार करा आणि कॉन्फिगर करा (Insert, format text and paragraphs, Create and configure document sections)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

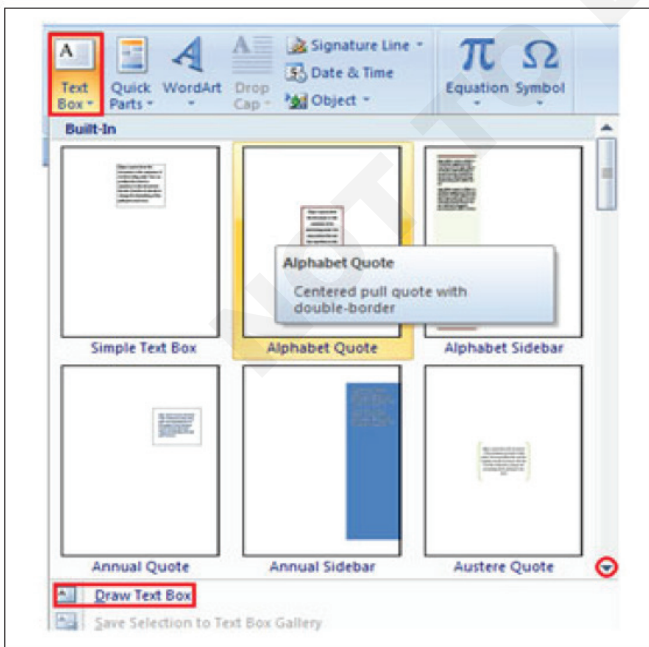
- MS Word मध्ये टेक्स्ट इन्सर्ट
- टेक्स्ट फॉर्मेट करा
- सेक्शन ब्रेक अड करा.

## MS Word मध्ये टेक्स्ट इन्सर्ट

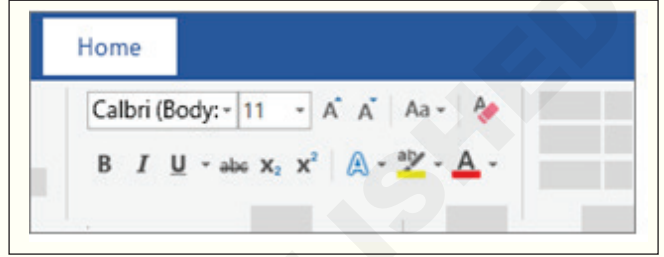
रिबनच्या खाली असलेल्या टेक्स्ट क्षेत्रात तुम्हाला ब्लिंकिंग कर्सर किंवा इन्सर्टेशन पॉइंट दिसेल. आता, जसे तुम्ही टाइप करणे सुरू कराल, तसे शब्द स्क्रीनवर टेक्स्ट भागात दिसतील. इन्सर्शन पॉइंटचे पोजिशन बदलण्यासाठी स्पेसबार, एंटर किंवा टॅब दाबा

टेक्स्ट घालण्यासाठी किंवा Word मध्ये नवीन डॉक्युमेंट्स तयार करण्यासाठी बेसिक पायऱ्या खाली लिस्ट बद्ध आहेत;

- स्टार्ट मेनूवर जा आणि Microsoft Word चिन्ह शोधा
- मायक्रोसॉफ्ट वर्ड उघडण्यासाठी आयकॉनवर क्लिक करा
- तुम्हाला रिबनच्या खाली टेक्स्ट क्षेत्रात एक ब्लिंकिंग कर्सर किंवा इन्सर्टेशन पॉइंट दिसेल
- आता तुम्ही टाइप करणे सुरू करताच, शब्द स्क्रीनवर टेक्स्ट एरियामध्ये दिसतील
- इन्सर्शन पॉइंटचे पोजिशन बदलण्यासाठी स्पेसबार, एंटर किंवा टॅब दाबा. कळा




## टेक्स्ट फॉर्मेटिंग करा



- 1 तुम्हाला फॉर्मेट करायचा असलेला टेक्स्ट निवडा.  
एकच शब्द निवडण्यासाठी, त्यावर डबल-क्लिक करा. टेक्स्ट ची ओळ निवडण्यासाठी, त्याच्या डावीकडे क्लिक करा.
- 2 फॉन्ट, फॉन्ट आकार, फॉन्ट रंग बदलण्यासाठी पर्याय निवडा किंवा टेक्स्ट ठळक, इटालिक किंवा अधोरेखित करा.

## फॉर्मेटिंग कॉपी करा

- 1 तुम्हाला कॉपी करायच्या असलेल्या फॉर्मेटिंगसह टेक्स्ट निवडा.
- 2 फॉर्मेट पेंटर वर क्लिक करा,  आणि नंतर तुम्हाला फॉर्मेटिंग कॉपी करायचे असलेला टेक्स्ट निवडा.

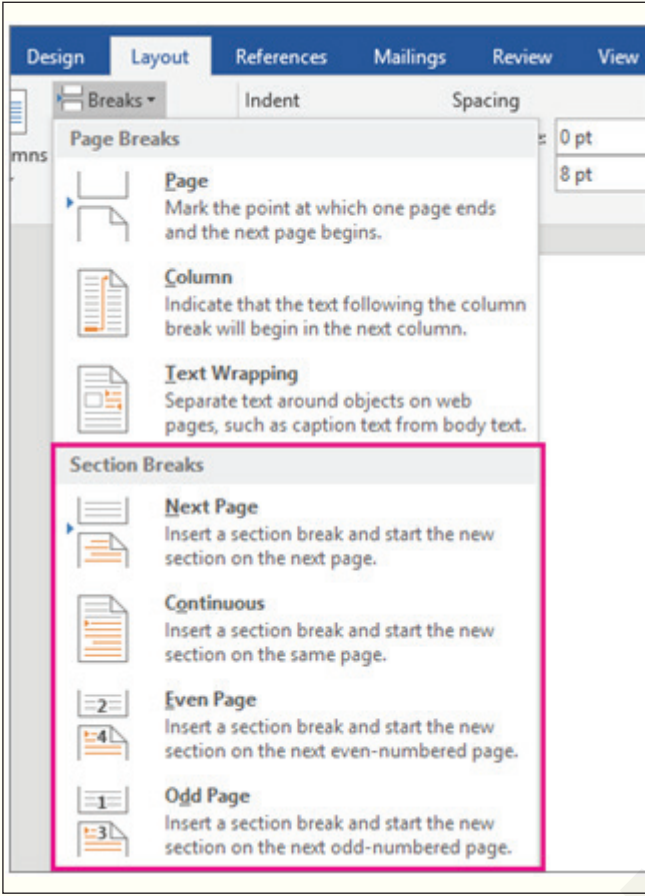
**टीप:** तुम्हाला एकापेक्षा जास्त ठिकाणी फॉर्मेटिंग कॉपी करायचे असल्यास फॉर्मेट पेंटरवर डबल-क्लिक करा.

## सेक्शन ब्रेक अड करा

- 1 तुम्हाला नवीन विपार्ट कोठे सुरू करायचा आहे ते निवडा.
- 2 लेआउट > ब्रेक्स वर जा.
- 3 तुम्हाला हवा असलेला विपार्ट खंड निवडा: पुढील पृष्ठ विपार्ट खंड पुढील पृष्ठावरील नवीन विपार्ट सुरू करतो. सतत विपार्ट खंडित केल्याने त्याच पृष्ठावरील नवीन विपार्ट सुरू होतो

## सेक्शन ब्रेक अड करा

- 1 तुम्हाला नवीन विपार्ट कोठे सुरू करायचा आहे ते निवडा.
- 2 लेआउट > ब्रेक्स वर जा.



3 तुम्हाला हवा असलेला विपार्ट खंड निवडा

∴ पुढील पृष्ठ -विपार्ट खंड पुढील पृष्ठावरील नवीन विपार्ट सुरू करतो.



- कैन्टीनुअस - सेक्शन ब्रेकमुळे त्याच पानावर नवीन विपार्ट सुरू होतो. नवीन पृष्ठ सुरू न करता स्तंभांची नंबर बदलण्यासाठी या टाइप चा विपार्ट खंडित केला जातो.



- इवन पेज - विपार्ट ब्रेक पुढील इवन -संख्येच्या पेजवर एक नवीन विपार्ट सुरू करतो.



- विषम पान -विपार्ट खंड पुढील विषम-संख्येच्या पेजवर एक नवीन विपार्ट सुरू करतो.



## टेबल तयार करा, बदला (Create, modify tables)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

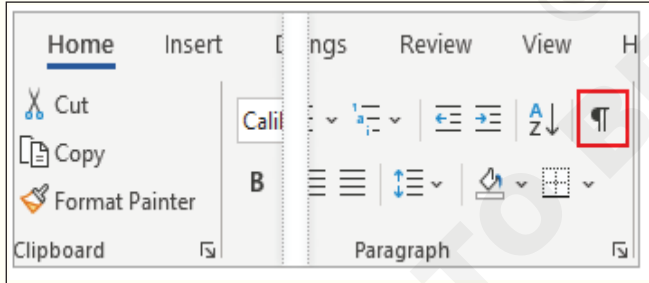
- टेबल तयार करा
- टेक्स्ट टेबलमध्ये रूपांतरित करा
- टेबल स्प्लिट करा.

मोठ्या टेबलसाठी, किंवा टेबल कस्टमाइज करण्यासाठी, इन्सर्ट > टेबल > टेबल इन्सर्ट निवडा. टिपा: जर तुमच्याकडे आधीच टॅबद्वारे टेक्स्ट विभक्त केला असेल, तर तुम्ही ते पटकन टेबलमध्ये रूपांतरित करू शकता. इन्सर्ट > टेबल निवडा आणि नंतर टेक्स्ट ते टेबलमध्ये रूपांतरित करा निवडा.

### टेबल तयार करा

डेटा संच, स्पेशल त: नंबर त्मक डेटा, अनेकदा टेक्स्ट च्या परिच्छेदापेक्षा टेबलमध्ये अधिक स्पष्टपणे आणि कार्यक्षमतेने सादर केले जाऊ शकतात. सारण्या मोठ्या प्रमाणात डेटा, किंवा जटिल डेटा अशा फॉर्मॅट नात सादर करतात जे वाचण्यास आणि समजण्यास सोपे असलेल्या रो आणि स्तंभांमध्ये संरचित करून, ज्यामध्ये डेटाचा उद्देश किंवा अर्थ स्पष्ट करण्यासाठी हेडर समाविष्ट असतात.

टेक्स्ट टेबलमध्ये किंवा टेबलमध्ये टेक्स्ट त रूपांतरित करण्यासाठी, होम टॅबवरील प्यारोग्राफ दर्शवा/लपवा या चिन्हावर क्लिक करून स्टार्ट करा जेणेकरून तुम्ही तुमच्या डॉक्युमेंट्स त टेक्स्ट कसा विभक्त केला आहे ते पाहू शकता.

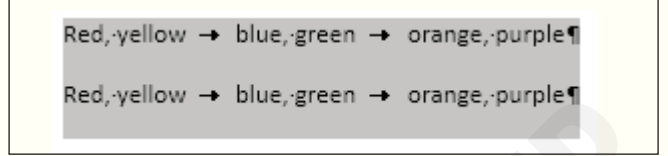


### टेक्स्ट टेबलमध्ये रूपांतरित करा

- 1 विभाजक कॅरक्टर इन्सर्ट -जसे की स्वल्पविराम किंवा टॅब हे टेबल स्तंभांमध्ये टेक्स्ट कुठे स्प्लिट करायचा हे सूचित करतात.

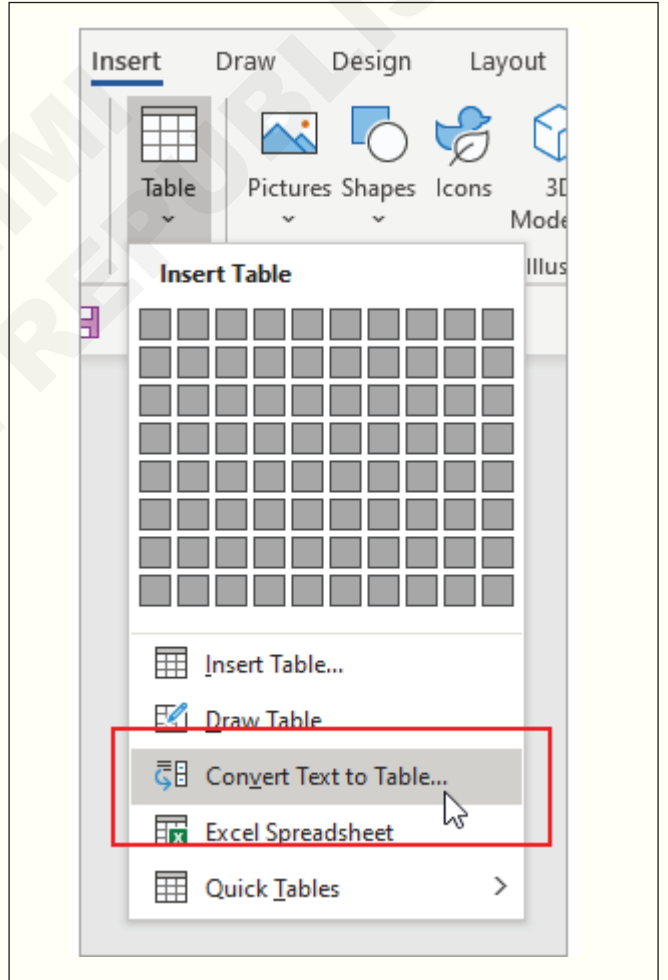
टीप: तुमच्या टेक्स्ट मध्ये स्वल्पविराम असल्यास, टॅब वापरा तुमच्या विभाजक कॅरक्टर साठी.

- 2 तुम्हाला नवीन टेबल रो कुठे सुरू करायची आहे हे सूचित करण्यासाठी प्यारोग्राफ चिन्हांचा वापर करा.  
या उदाहरणात, टॅब आणि प्यारोग्राफ चिन्ह असतील 3 कॉलम आणि 2 रो असलेले टेबल तयार करा:
- 3 तुम्हाला रूपांतरित करायचा असलेला टेक्स्ट निवडा आणि नंतर इन्सर्ट > टेबल > टेक्स्ट ते टेबलवर रूपांतरित करा वर क्लिक करा.

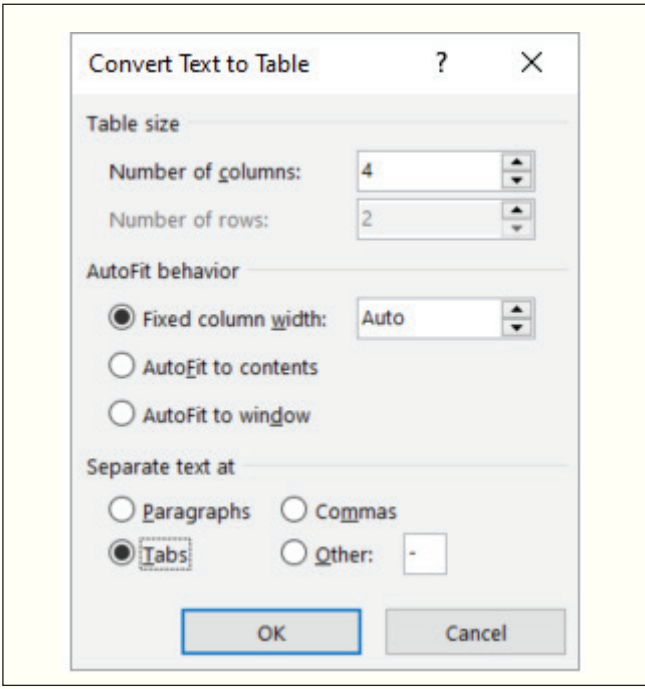


- 4 टेक्स्ट ते टेबलमध्ये रूपांतरित करा बॉक्समध्ये, तुम्हाला हवे असलेले पर्याय निवडा.

टेबल आकाराखाली, नंबर तुम्हाला हव्या असलेल्या कॉलम आणि रो च्या संख्येची जुळत असल्याची खात्री करा.



- 1 ऑटोफिट वर्तन इंटरनल , तुमचे टेबल कसे दिसावे ते निवडा. शब्द ऑटोमेटिक टेबल स्तंभांसाठी रुंदी निवडतो. तुम्हाला वेगळ्या कॉलम ची रुंदी हवी असल्यास, यापैकी एक पर्याय निवडा:
- 2 येथे सेपरेट टेक्स्ट इंटरनल , तुम्ही टेक्स्ट त वापरलेले सेपरेटर कॅरक्टर निवडा.



- 3 ओके क्लिक करा. टेबलमध्ये रूपांतरित केलेला टेक्स्ट असे काहीतरी दिसला पाहिजे:

Red, yellow	blue, green	orange, purple
Red, yellow	blue, green	orange, purple

हे करण्यासाठी	हा पर्याय निवडा
<p>सर्वांसाठी रुंदी स्पेसिफाइड करा</p> <p>प्रत्येक कॉलम तील टेक्स्ट च्या रुंदीमध्ये बसण्यासाठी स्तंभांचा आकार बदला</p> <p>सबलब्ध जागेची रुंदी बदलल्यास टेबलचा आकार ऑटोमैटिक बदला (उदाहरणार्थ, वेब लेआउट किंवा लँडस्केप अभिमुखता) मध्ये कॉलम निश्चित कॉलम</p>	<p>रुंदीचा बॉक्स, एक व्हॅल्यू टाइप करा किंवा निवडा.</p> <p>कंटेंट साठी ऑटोफिट</p> <p>विंडोमध्ये ऑटोफिट</p>

### एक टेबल स्प्लिट करा

- 1 तुमचा कर्सर तुमच्या दुसऱ्या टेबल च्या पहिल्या रो च्या रो वर ठेवा. उदाहरण टेबल मध्ये, ते तिसऱ्या रांगेत आहे. ...
- 2 लेआउट टॅबवर, मर्ज ग्रुपमध्ये, स्प्लिट टेबलवर क्लिक करा. टेबल दोन टेबलांमध्ये स्प्लिट होते.

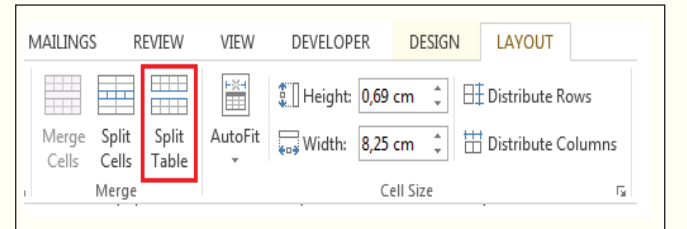
एकदा का तुमच्याकडे वर्डमध्ये टेबल असेल, तुम्ही ते टेबल दोन किंवा अधिक टेबलांमध्ये स्प्लिट करण्याचा निर्णय घेऊ शकता. अशा प्रकारे, तुम्ही लहान टेबल्स तयार करू शकता किंवा दोन टेबलांमध्ये टेक्स्ट जोडू शकता.

- 1 तुमचा कर्सर तुमच्या दुसऱ्या टेबल च्या पहिल्या रो च्या रो वर ठेवा. उदाहरण टेबल मध्ये, ते तिसऱ्या रांगेत आहे.

1	A
2	B
3	C
4	D

जेव्हा तुम्ही टेबलच्या आत क्लिक करता, तेव्हा रिबनवर दोन नवीन टेबल टूल्स टॅब दिसतात: डिझाइन आणि लेआउट. जेव्हा तुम्ही टेबलमध्ये असता तेव्हाच ही साधने दिसतात, उदाहरणार्थ, जेव्हा तुम्ही सेलमध्ये कंटेंट जोडता.

- 2 वर लेआउट टॅब, मध्ये विलीन ग्रुप , स्प्लिट टेबल क्लिक करा.



टेबल दोन टेबलांमध्ये स्प्लिट होते.

जोपर्यंत अनेक रो आहेत तोपर्यंत तुम्ही टेबल आणखी स्प्लिट करू शकता.

1	A
2	B

3	C
4	D

## रेफरेन्स एलिमेंट आणि सारण्या तयार करा आणि मॅनेज करा (Create and manage reference elements and tables)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- रेफरेन्स एलिमेंट मॅनेज करा
- टेबल कंटेंट कस्टमाइज करा.

शब्दाची स्वतःची, बेसिक रेफरेन्स सिस्टिम रिबनवरील 'रेफरेन्स' टॅबद्वारे सबलब्ध आहे. हे तुम्हाला डॉक्युमेंट्स त रेफरेन्स जोडू देते आणि नंतर टेक्स्ट च्या शेवटी एक ग्रंथलिस्ट तयार करू देते. रेफरेन्स मुख्य लिस्ट मध्ये स्टोर केले जातात, ज्याचा वापर पुढील डॉक्युमेंट्स मध्ये रेफरेन्स जोडण्यासाठी केला जाऊ शकतो.

टीप: ही सिस्टिम EndNote, Mendeley किंवा Zotero सारखी शक्तिशाली नाही.

### पहिली स्टेप

- 1 तुमचा डॉक्युमेंट्स तयार करा.
- 2 टेक्स्ट त साइटेशन घालण्यासाठी रिबनवरील 'रेफरेन्स' टॅबवर जा आणि 'साइटेशन इन्सर्ट' आणि 'नवीन सोर्स ऑड करा' वर क्लिक करा.
- 3 'सोर्स चा टाइप' निवडा आणि बॉक्स भरा.
- 4 तुमच्या डॉक्युमेंट्स त सर्व उद्धरणे ऑड करा.
- 5 तुम्ही पूर्ण केल्यावर, तुमच्या डॉक्युमेंट्स च्या शेवटी जा आणि 'बिब्लिओग्राफी' पर्यायावर क्लिक करा. प्रीफॉर्मेट केलेल्या पर्यायांपैकी एक निवडा किंवा स्वतःचे फॉर्मॅटिंग करण्यासाठी फक्त ग्रंथलिस्ट इन्सर्ट .
- 6 डीफॉल्ट 'APA' स्टाइल मधून तुमच्या संदर्भाची स्टाइल बदलण्यासाठी 'स्टाइल' लिस्ट वर क्लिक करा आणि दुसरी निवडा.

लक्षात ठेवा की Word मध्ये सबलब्ध स्टाइल कदाचित तुमच्या असाइनमेंटसाठी आवश्यक असलेल्या शैलीशी जुळत नाहीत. ते काळजीपूर्वक तपासा आणि जर ते जुळत नसतील तर तुम्ही रेफरेन्स ग्रंथ एडिट करण्यायोग्य टेक्स्ट मध्ये रूपांतरित करू शकता. हे करण्याच्या सूचनांसाठी खालील पाथ दर्शक पहा.

वर्डमधील रेफरेन्स तुमच्या संगणकाच्या हार्ड डिस्कवर साठवले जातात. तुम्हाला दुसऱ्या संगणकावर त्यांचावर काम करायचे असल्यास, फाइल सर्च प्यासाठी या स्टेप्स चे अनुसरण करा:

- स्टार्ट मेनू सर्च बॉक्समध्ये % APPDATA% टाइप करा
- फोल्डर क्लिक करा रोमिंग > मायक्रोसॉफ्ट > ग्रंथलिस्ट (Bibliography)
- रेफरेन्स XML फाइलमध्ये सेव्ह केले जातात सोर्स

### शब्दातील रेफरेन्स

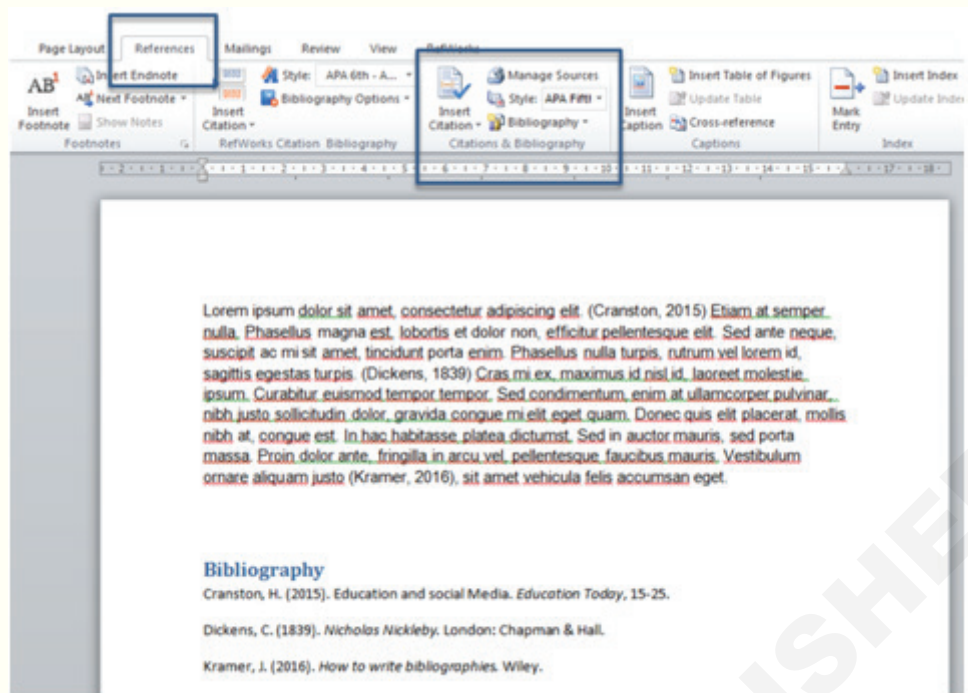
Microsoft Word मध्ये रेफरेन्स टॅबमध्ये एक साधे अंगभूत रेफरेन्स कार्य आहे. येथे तुम्ही व्यक्तिचलितपणे रेफरेन्स प्रविष्ट करू शकता आणि त्यांचा टेक्स्ट त रेफरेन्स घेऊ शकता आणि सर्वात सामान्य शैलींमध्ये (APA, Harvard, MLA, इत्यादी) रेफरेन्स लिस्ट तयार करू शकता. तुमच्याकडे मर्यादित संख्येने रेफरेन्स असल्यास आणि तुम्हाला वेगवेगळ्या संगणकांवर तुमच्या संदर्भांमध्ये एक्सेस करण्याची आवश्यकता नसल्यास हे अंगभूत कार्य चांगले आहे.

रिबनवर, संदर्भावर जा. डावीकडील कंटेंट गटाच्या खाली, कंटेंट टेबल बटणावर क्लिक करा. तुम्ही डीफॉल्ट पर्याय समाविष्ट करणे निवडू शकता किंवा मेनूच्या तळाशी कस्टम सामुग्री टेबल वर क्लिक करू शकता. कस्टम कंटेंट टेबल घालण्यासाठी, मेनूमधून पर्याय निवडा.

आपण Word मध्ये कंटेंट ची टेबल तयार केल्यानंतर, आपण ते कसे दिसेल ते कस्टमाइज करू शकता. तुमची कंटेंट टेबल कस्टमाइज केल्याने तुमची प्राधान्ये तुमच्या विद्यमान टेबल वर लागू होतात. तुम्ही जे पाहता ते आवडले, ओके निवडा. तुम्ही दिसण्यात आनंदी नसल्यास, फक्त रद्द करा निवडा आणि सर्व बदल विसरले जातील. तुमची कंटेंट टेबल अजूनही तेथे असेल.

### तुमची विद्यमान कंटेंट टेबल कस्टमाइज करण्यासाठी:

- 1 जा रेफरेन्स > टेबल ऑफ़ कंटेंट .
- 2 निवडा कस्टम टेबल ऑफ़ कंटेंट.
- 3 पृष्ठ क्रमांक दर्शविण्यासाठी, लपवण्यासाठी आणि एलाइनमेंट करण्यासाठी सेटिंग्ज वापरा, टॅब लीडर ऑड करा किंवा बदला, फॉर्मेट सेट करा आणि किती स्तर हेडिंग दाखवायचे ते स्पेसिफाइड करा. अधिक इनफार्मेशन साठी, कस्टम कंटेंट टेबल पहा.



## चित्रे आणि टेक्स्ट बॉक्स इन्सर्ट , फॉर्मॅटिंग करा (Insert, format illustrations and text boxes)

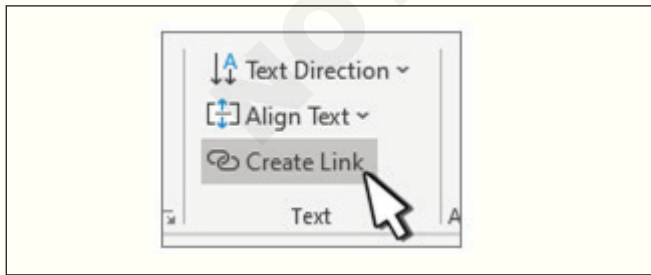
उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टेक्स्ट बॉक्स
- ग्राफिक एलिमेंटमध्ये टेक्स्ट ऑड करा.

टेक्स्ट बॉक्स हा एक ऑब्जेक्ट आहे जो तुम्ही तुमच्या डॉक्युमेंट्स त जोडू शकता जो तुम्हाला तुमच्या फाइलमध्ये कुठेही टेक्स्ट टाकू आणि टाइप करू देतो. टेक्स्ट बॉक्स स्पेसिफिक टेक्स्ट कडे लक्ष वेधण्यासाठी सबयुक्त ठरू शकतात आणि जेव्हा तुम्हाला तुमच्या डॉक्युमेंट्स त टेक्स्ट फिरवायचा असेल तेव्हा ते सबयुक्त ठरू शकतात.

**Insert > Text Box वर जा आणि नंतर लिस्ट मधून पूर्व-फॉर्मॅटिंग टेक्स्ट बॉक्स निवडा, Office.com वरून अधिक टेक्स्ट बॉक्स निवडा किंवा टेक्स्ट बॉक्स काढा निवडा.** तुम्ही टेक्स्ट बॉक्स काढा निवडल्यास, डॉक्युमेंट्स त क्लिक करा, आणि नंतर टेक्स्ट बॉक्स तुम्हाला पाहिजे त्या आकारात काढण्यासाठी ड्रॅग करा.

- टेक्स्ट बॉक्समध्ये टेक्स्ट फॉर्मॅट करण्यासाठी, टेक्स्ट निवडा, आणि नंतर होम टॅबवरील फॉन्ट गटातील फॉर्मॅटिंग पर्याय वापरा.
- टेक्स्ट बॉक्स स्वतःच फॉर्मॅटिंग करण्यासाठी, फॉर्मॅटिंग रेफरेन्स टॅबवरील कमांड वापरा, जे तुम्ही टेक्स्ट बॉक्स निवडता तेव्हा रेखापिक्चर साधने इंटर्नल दिसतात.
- टेक्स्ट बॉक्स ठेवण्यासाठी, त्यावर क्लिक करा, आणि नंतर पॉइंटर बनल्यावर, टेक्स्ट बॉक्सला नवीन ठिकाणी ड्रॅग करा.
- तुम्ही टेक्स्ट बॉक्स किंवा आकारामधून बॉर्डर बदलू किंवा काढू शकता.
- तुमच्याकडे अनेक टेक्स्ट बॉक्स असल्यास, तुम्ही त्यांना एकत्र जोडू शकता जेणेकरून टेक्स्ट एका बॉक्समधून दुसऱ्या बॉक्समध्ये जाईल. टेक्स्ट बॉक्सपैकी एकावर क्लिक करा आणि फॉर्मॅट टॅबवर, ड्रॉइंग टूल्स इंटर्नल , टेक्स्ट ग्रुपमध्ये, लिंक तयार करा क्लिक करा.

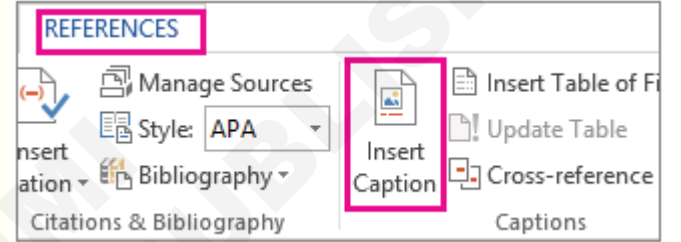


### चित्रासाठी कॅप्शन इन्सर्ट करा

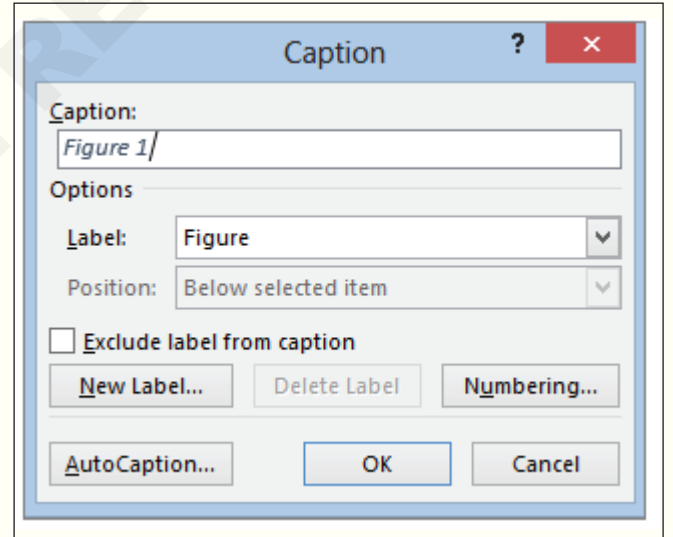
कॅप्शन इन्सर्ट Word मधील वैशिष्ट्य डॉक्युमेंट्स तील चित्रांना मेथड्स शीरपणे कॅप्शन जोडणे सोपे करते.

PowerPoint सारख्या इतर Office ॲप्समध्ये, तुम्ही व्यक्तिचलितपणे प्रतिमेजवळ टेक्स्ट बॉक्स जोडता आणि नंतर टेक्स्ट बॉक्स आणि प्रतिमा ग्रुप बद्ध करा. खालील सूचना पहा. जर तुमच्याकडे मालिकेत अनेक चित्रे असतील, तर तुम्हाला ती व्यक्तिचलितपणे क्रमांकित करावी लागतील.

- 1 तुम्हाला कॅप्शन ऑड करायचा असलेल्या चित्रावर क्लिक करा.
- 2 रेफरेन्स क्लिक करा > कॅप्शन इन्सर्ट



- 3 डीफॉल्ट लेबल (आकृती) वापरण्यासाठी, कॅप्शन बॉक्समध्ये तुमचा कॅप्शन टाइप करा.

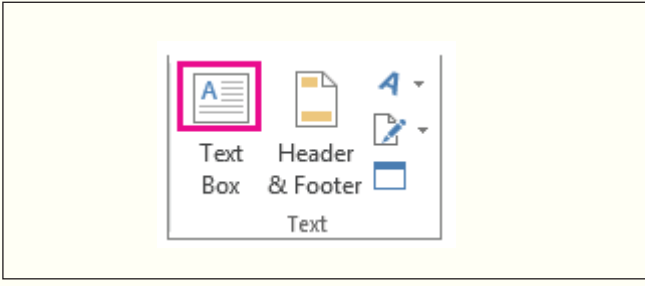


**टीप:** तुम्ही क्लिक करून तुमचे स्वतःचे डीफॉल्ट कॅप्शन लेबल देखील तयार करू शकतानवीन लेबल, आणि नंतर मध्ये आपले कॅप्शन जोडणेलेबलबॉक्स. शब्द ऑटोमैटिक तुमच्यासाठी नवीन लेबले क्रमांकित करतो.

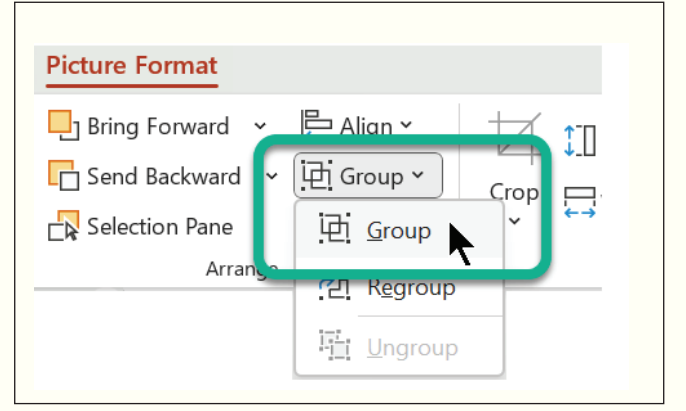
### इतर सर्व ऑफिस ॲप्स

- 1 वापरा इन्सर्ट > टेक्स्ट बॉक्स चित्राजवळ एक बॉक्स काढण्यासाठी. (अतिरिक्त तपशीलांसाठी टेक्स्ट बॉक्स ऑड करा पहा.)





2. टेक्स्ट बॉक्समध्ये क्लिक करा आणि तुम्हाला केषन साठी वापरायचा असलेला टेक्स्ट टाईप करा.
3. टेक्स्ट निवडा. होम टॅबवर, तुम्हाला हवे तसे केषन स्टाइल बद्ध करण्यासाठी फॉन्ट पर्याय वापरा.
4. पिक्चर आणि टेक्स्ट बॉक्स निवडण्यासाठी Ctrl+क्लिक वापरा, आणि नंतर पिक्चर फॉर्मॅट टॅबवर, ग्रुप व्यवस्थित करा, ग्रुप > ग्रुप निवडा.



© NIMI  
NOT TO BE REPUBLISHED

## कमेंट बदल ट्रॅकिंग आणि मेलिंग मॅनेज करा (Manage comments change tracking and mailings)


उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- Word मध्ये कोलायब्रेशन
- डॉक्युमेंट्स को -एडिट करा
- चालू आणि बंद बदलांचा मागोवा घ्या.

### Word मध्ये कोलायब्रेशन करा

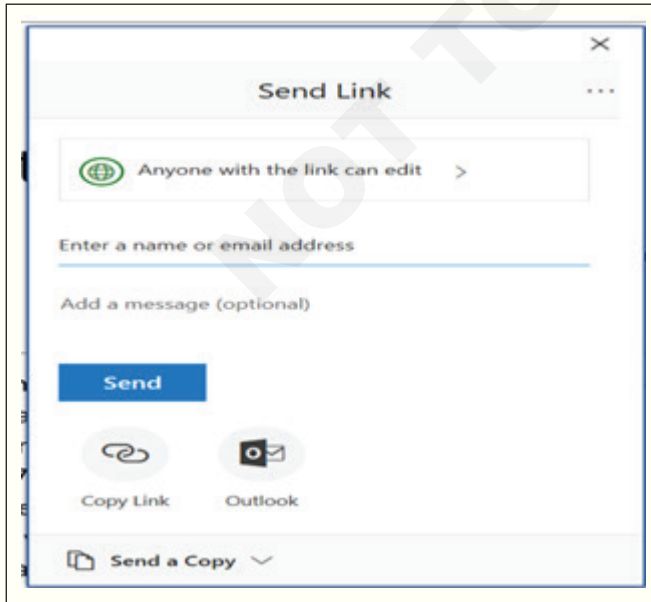
- 1 शेअर निवडा. रिबन वर. किंवा, फाइल > शेअर निवडा. टीप: तुमची फाईल आधीच OneDrive वर सेव्ह केलेली नसल्यास, तुम्हाला तुमची फाईल शेअर करण्यासाठी OneDrive वर अपलोड करण्यास सांगितले जाईल.
- 2 ड्रॉप-डाउनमधून तुम्हाला कोणाशी शेअर करायचे आहे ते निवडा किंवा नाव किंवा ईमेल एड्रेस प्रविष्ट करा.
- 3 संदेश ऑड करा (पर्यायी) आणि पाठवा निवडा.

तुमचा डॉक्युमेंट्स शेअर करा Word:

- 1 निवडा शेअर  रा रिबन वर.  
किंवा, फाइल > शेअर निवडा.

टीप: तुमची फाईल आधीच OneDrive वर सेव्ह केलेली नसल्यास, तुम्हाला तुमची फाईल शेअर करण्यासाठी OneDrive वर अपलोड करण्यास सांगितले जाईल.

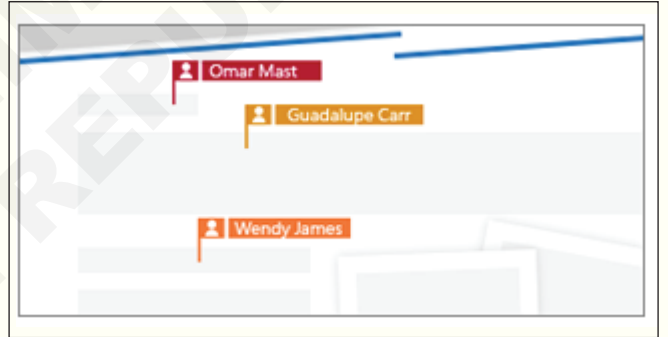
- 2 ड्रॉप-डाउनमधून तुम्हाला कोणाशी शेअर करायचे आहे ते निवडा किंवा नाव किंवा ईमेल एड्रेस प्रविष्ट करा.
- 3 संदेश ऑड करा (पर्यायी) आणि पाठवा निवडा.



### डॉक्युमेंट्स को -एडिट करा

तुम्ही तुमचा डॉक्युमेंट्स शेअर केल्यानंतर, तुम्ही त्या फाईलवर त्याच वेळी इतरांसोबत काम करू शकता.

- सर्वोत्तम अनुभवासाठी, वेबसाठी Word मध्ये एकत्र काम करा आणि रिअल-टाइम बदल पहा.
- सामायिक करा इंटरनल, तुम्हाला इतर कोण फाइल एडिट करत आहे त्यांची नावे दिसेल.
- रंगीत ध्वज तुम्हाला डॉक्युमेंट्स त प्रत्येक व्यक्ती नेमके कुठे काम करत आहे हे दाखवतात.



### बदलांचा मागोवा घ्या आणि रिव्यु करा

- 1 बदलांचा मागोवा घेण्यासाठी, निवडारिव्यु > बदलांचा मागोवा घ्या.
- 2 बदलांचे रिव्यु करण्यासाठी, बदलापूर्वी कर्सर ठेवा आणि निवडा:
  - स्वीकारा बदल ठेवण्यासाठी, किंवा
  - नाकारणे ते काढण्यासाठी.

### ट्रॅक चेंज चालू आणि बंद करा

रिव्यु टॅब वर, निवडा ट्रॅक चेंज.

- ट्रॅक चेंज चालू असताना, विपार्ट हायलाइट केला जातो. हटवण्याला स्ट्राइकथ्रूने चिन्हांकित केले आहे आणि जोडण्यांना अधोरेखित करून चिन्हांकित केले आहे. वेगवेगळ्या लेखकांचे बदल वेगवेगळ्या रंगांनी सूचित केले जातात.
- ट्रॅक चेंज बंद असताना, विपार्ट हायलाइट केला जात नाही. शब्द बदल चिन्हांकित करणे थांबवतो, परंतु रंगीत अधोरेखित आणि स्ट्राइकथ्रू अजूनही डॉक्युमेंट्स त आहेत.

**टीप:** तुम्ही स्टेटस बारमध्ये ट्रॅक चेंजेस इंडिकेटर देखील जोडू शकता. स्टेटस बारवर उजवे-क्लिक करा आणि ट्रॅक चेंज निवडा.

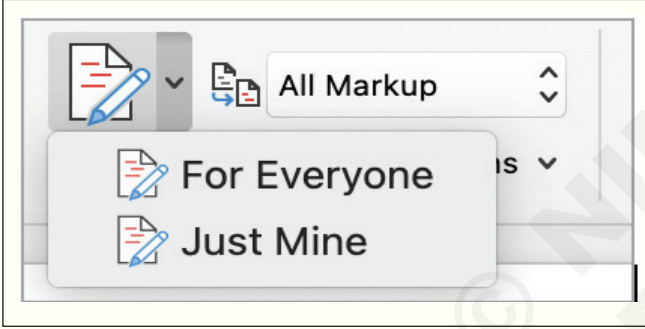
### ट्रॅक केलेले बदल पहा

**टीप:** जेव्हा एखादे डॉक्युमेंट्स एडिटिंग किंवा रिव्यु मोडमध्ये असते, तेव्हा तुम्ही कोलॅबोरेटरचे पूर्ण, सुचवलेले बदल पाहण्यासाठी ट्रॅक केलेला बदल निवडू शकता. ती क्रिया तुमच्या अंतिम डॉक्युमेंट्स चे काय करेल याचे प्रीव्यू पाहण्यासाठी स्वीकारा किंवा नकार द्या बटणावर फिरवा.

### कोणाचे बदल ट्रॅक करायचे ते निवडा

तुम्ही फक्त तुमचे स्वतःचे बदल किंवा प्रत्येकाच्या बदलांचा मागोवा घेणे निवडू शकता.

- फक्त तुमचे स्वतःचे बदल ट्रॅक करण्यासाठी - रिव्यु टॅबवर, निवडा ट्रॅक चेंजस > फक्त माझे.
- प्रत्येकाच्या बदलांचा मागोवा घेण्यासाठी - रिव्यु टॅबवर, निवडा ट्रॅक चेंजस > प्रत्येकासाठी



**टीप:** इतरांना ट्रॅक चेंज बंद करण्यापासून रोखण्यासाठी पासवर्ड वापरण्यासाठी - रिव्यु टॅबवर, ट्रॅक चेंज > लॉक ट्रॅकिंग निवडा.

### तुम्हाला डॉक्युमेंट मधील बदल कसे पहायचे आहेत ते निवडा

तुम्हाला कोणता मार्कअप पहायचा आहे ते तुम्ही निवडू शकता.

- 1 रिव्यु टॅबवर, ट्रॅकिंग निवडा.
  - 2 प्रदर्शित करण्यासाठी ड्रॉप-डाउन लिस्ट साठी सर्व मार्कअप निवडा.
  - 3 खालील पर्यायांपैकी एक निवडा.
- सिंपल मार्कअप मार्जिनमध्ये लाल रेषेसह ट्रॅक केलेले बदल प्रदर्शित करते.
  - ऑल मार्कअप प्रत्येक समीक्षकासाठी टेक्स्ट आणि ओळींच्या वेगवेगळ्या रंगांसह ट्रॅक केलेले बदल प्रदर्शित करते
  - नो मार्कअप दर्शविण्यासाठी मार्कअप लपवते समाविष्ट बदलांसह डॉक्युमेंट्स
  - मूळ डॉक्युमेंट्स ट्रॅक केलेले बदल आणि कमेंट दर्शविल्याशिवाय प्रदर्शित करते. तथापि, डॉक्युमेंट्स तील कोणतेही ट्रॅक केलेले बदल किंवा कमेंट ज्या स्वीकारल्या गेल्या नाहीत, नाकारल्या गेल्या किंवा हटविल्या गेल्या नाहीत त्या डॉक्युमेंट्स त राहतील.

ट्रॅक चेंज प्रदर्शन पाथ निवडा

तुम्ही प्रदर्शित होणाऱ्या पुनरावृत्तीचे टाइप आणि ते प्रदर्शित करण्याचा पाथ निवडू शकता. जर तुम्ही पुनरावृत्ती फुगे म्हणून दर्शविणे निवडले, तर ते डॉक्युमेंट्स च्या मार्जिनमध्ये प्रदर्शित होतील. जर तुम्ही त्यांना थेट डॉक्युमेंट्स त प्रदर्शित करणे निवडले तर. इन लाइन रिव्हिजनमध्ये आतल्या बलून ऐवजी स्ट्राइकथ्रूसह सर्व हटवणे प्रदर्शित केले जातात.

- 1 रिव्यु टॅबवर, ट्रॅक > मार्कअप दर्शवा निवडा.
- 2 बलून निवडा आणि नंतर तुम्हाला हवा असलेला डिस्प्ले टाइप निवडा.

- बलून मध्ये आवर्तने दाखवा
- सर्व पुनरावृत्ती इनलाइन दाखवा
- बलून मध्ये फक्त फॉर्मॅटिंग दाखवा

**टीप:** जर तुम्हाला बलून मधील ट्रॅक केलेले बदल पहायचे असतील, तर तुम्ही प्रिंट लेआउट व्हॅ किंवा वेब लेआउट दृश्यात असणे आवश्यक आहे.

### रिव्यु च्या प्रकारानुसार बदल प्रदर्शित करा

- 1 रिव्यु टॅबवर, बदलांचा मागोवा घ्या > मार्कअप दर्शवा निवडा.
  - 2 रिव्यु चा टाइप निवडा.
- समाविष्ट करणे आणि हटवणे
  - फॉर्मॅटिंग

आयटमच्या पुढील खूण सूचित करते की ती निवडली आहे.

**टीप:** मार्कअप दाखवा मेनूवर क्लिक करून तुम्ही मार्कअपचा टाइप लपवला तरीही, प्रत्येक वेळी तुम्ही किंवा समीक्षकाने डॉक्युमेंट्स उघडल्यावर मार्कअप ऑटोमैटिक प्रदर्शित होतो.

### रिव्यु कर्त्याद्वारे बदल प्रदर्शित करा

- 1 रिव्यु > ट्रॅकिंगमध्ये, मार्कअप दर्शवा निवडा.
- 2 स्पेसिफिक लोक निवडा.
- 3 तुम्ही ज्यांचे बदल दर्शवू इच्छिता त्या समीक्षकांच्या नावांपुढील चेकबॉक्ससह वगळता सर्व चेक बॉक्स क्लिक करणेसाठी निवडा.

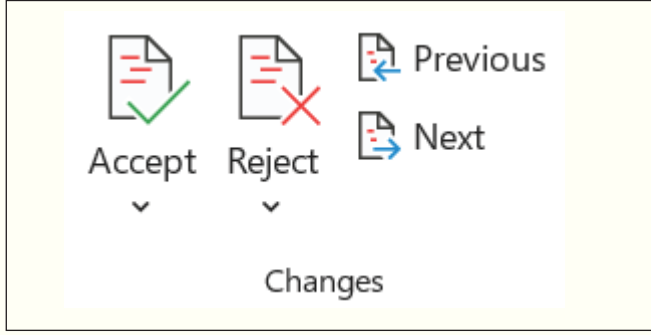
**टीप:** लिस्ट मधील सर्व रिव्यु कर्त्यासाठी सर्व चेक बॉक्स निवडण्यासाठी किंवा क्लिक करणेसाठी, सर्व रिव्यु कर्ते निवडा.

### बदल मेनू विभागात ट्रॅक केलेले बदल नॅव्हिगेट करा

तुम्ही एका ट्रॅक केलेल्या बदलावरून दुसऱ्यावर नेव्हिगेट करू शकता.

- 1 रिव्यु > बदल, पाहण्यासाठी पुढील निवडा पुढे ट्रॅक केलेला बदल.

2 मध्ये रिव्यु > बदल, पाहण्यासाठी मागील निवडा मागील ट्रॅक केलेला बदल.



### ट्रॅक केलेले बदल स्वीकारा किंवा नकार द्या

ट्रॅक केलेले बदल संबोधित करण्याचे अनेक पाथ आहेत. तुम्ही चिन्हांकित बदलावर क्लिक करता तेव्हा कार्ड डिस्प्लेद्वारे समालोचक ओळखणे आणि ट्रॅक केलेल्या बदलाला प्रतिसाद देणे ही सर्वात प्रवीण मेथड्स आहे. कार्ड कमेंट कर्त्याचे नाव आणि सूचना स्वीकारण्याचा किंवा नाकारण्याचा पर्याय प्रदर्शित करते.

मुख्य मेनू वापरून क्रमाने ट्रॅक केलेले बदल स्वीकारा किंवा नकार द्या

तुम्ही डॉक्युमेंट्स च्या सुरुवातीपासून डॉक्युमेंट्स च्या शेवटपर्यंत शिकवलेल्या ट्रॅक केलेल्या बदलाचे निराकरण करू शकता.

- 1 रिव्यु > चेंज > पुढील निवडा.
- 2 स्वीकारा किंवा नकार द्या निवडा. पुढील क्रमवारी (सॉर्टिंग)चा मागोवा घेतला

तुमच्या पुनरावलोकनासाठी आणि कृतीसाठी बदल हायलाइट केला आहे.

टीप: तुम्ही मुख्य मेनूचे ड्रॉपडाउन मेनू वापरून वैयक्तिक ट्रॅक केलेले बदल स्वीकारू किंवा नाकारू शकता. ट्रॅक केलेल्या बदलांचे रिव्यु करण्यासाठी ही मेथड्स सहसा मेथड्स शीर दृष्टिकोन म्हणून वापरली जाते.

### मुख्य मेनू वापरून बदल स्वीकारा

- 1 रिव्यु > चेंज मध्ये, स्वीकारा निवडा.
- 2 पर्यायांपैकी एक निवडा.
  - स्वीकारा आणि पुढे जा
  - हा बदल स्वीकारा
  - सर्व बदल स्वीकारा
  - सर्व बदल स्वीकारा आणि ट्रॅकिंग थांबवा

### मुख्य मेनू वापरून बदल नकार द्या

- 1 पुनरावलोकनात > बदल, > नकार द्या निवडा.
- 2 पर्यायांपैकी एक निवडा.

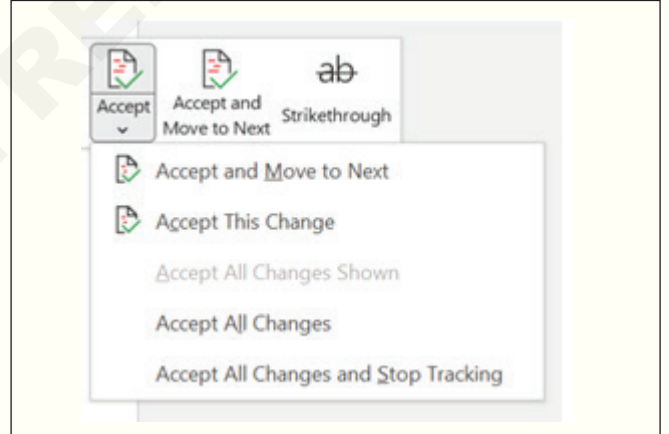
- नकार द्या आणि पुढे जा
- हा बदल नाकार
- सर्व बदल नाकारा
- सर्व बदल नाकारा आणि ट्रॅकिंग थांबवा

### एका डॉक्युमेंट्स त उजवे क्लिक करून बदल स्वीकारा किंवा नकार द्या

डॉक्युमेंट्स त काम करताना, तुम्ही बदल स्वीकारण्यासाठी किंवा नाकारण्यासाठी ट्रॅक केलेल्या बदलावर उजवे-क्लिक करू शकता. उजवे-क्लिक मेथड्स वापरल्याने बॅनर मेनू पर्याय आणि ड्रॉप-डाउन मेनू पर्याय प्रदर्शित होतो.

### उजवे-क्लिक बॅनर पर्यायासह इन-लाइन बदल स्वीकारा

- 1 सिंगल सूचनेचे रिव्यु करण्यासाठी डॉक्युमेंट्स तील ट्रॅक केलेल्या बदलावर उजवे-क्लिक करा.
- 2 बॅनरवर, खालीलपैकी एक निवडा.
  - स्वीकारा > स्वीकारा आणि पुढे जा
  - स्वीकारा > हा बदल स्वीकारा
  - स्वीकारा > सर्व बदल स्वीकारा
  - स्वीकारा > सर्व बदल स्वीकारा आणि ट्रॅकिंग थांबवा
  - स्वीकारा आणि पुढे जा



### उजवे-क्लिक ड्रॉप-डाउन मेनू पर्यायासह इन-लाइन बदल स्वीकारा

- 1 सिंगल सूचनेचे रिव्यु करण्यासाठी डॉक्युमेंट्स तील ट्रॅक केलेल्या बदलावर उजवे-क्लिक करा.
- 2 ड्रॉप-डाउन मेनूमध्ये, खालीलपैकी एक निवडा.
  - समाविष्ट करणे स्वीकारा (किंवा हटवणे स्वीकारा)
  - समाविष्ट करणे नाकारणे (किंवा हटवणे नाकारणे)

मुख्य मेनूवर सर्व ट्रॅक केलेले बदल एकाच वेळी स्वीकारा किंवा नाकारा. सर्व ट्रॅक केलेले बदल स्वीकारा

- 1 रिव्यु > बदलांमध्ये, स्वीकारा निवडा.

2 पर्यायांपैकी एक निवडा.

- दाखवलेले सर्व बदल स्वीकारा
- सर्व बदल स्वीकारा
- सर्व बदल स्वीकारा आणि ट्रॅकिंग थांबवा

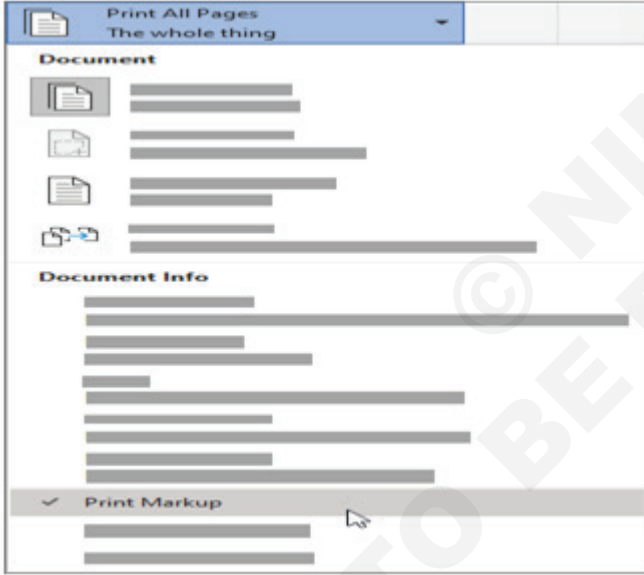
#### सर्व ट्रॅक केलेले बदल नकार द्या

1 रिव्यु > बदलांमध्ये, नकार द्या निवडा.

2 पर्यायांपैकी एक निवडा.

- दाखवलेले सर्व बदल नाकारा
- सर्व बदल नाकारा
- सर्व बदल नाकारा आणि ट्रॅकिंग थांबवा

**टीप:** जेव्हा रिव्यु > ट्रॅकिंग > मार्कअप दर्शवा > स्पेसिफिक लोक > सर्व रिव्यु कर्ते निवडले जातात, तेव्हा दाखवलेले सर्व बदल स्वीकारा आणि दाखवलेले सर्व बदल नाकारू नका हे पर्याय प्रदर्शित होत नाहीत.



#### प्रिंट करताना ट्रॅक केलेले बदल आणि कमेंट लपवा

बदल लपविल्याने ते डॉक्युमेंट्स तून काढले जात नाहीत.

तुमच्या डॉक्युमेंट्स तून मार्कअप काढून टाकण्यासाठी तुम्ही बदल गटातील स्वीकारा आणि नकार द्या कमांड वापरणे आवश्यक आहे.

1 फाइल > प्रिंट > सेटिंग्ज > सर्व पेजेज प्रिंटेड करा वर जा.

2 डॉक्युमेंट्स इनफार्मेशन इंटरनल, चेक मार्क क्लियर करण्यासाठी प्रिंट मार्कअप निवडा.

#### रिव्यु सबखंडावरील सारांश लिस्ट मध्ये तुमचे सर्व बदल पहा

1 रिव्यु > ट्रॅकिंगमध्ये, रिव्यु पॅन निवडा.

2 तुम्हाला तुमच्या डॉक्युमेंट्स च्या पुढे किंवा खाली रिव्यु पॅन पहायचे असल्यास निवडा.

• तुमच्या डॉक्युमेंट्स च्या शेजारी सर्व बदलांची लिस्ट पाहण्यासाठी रिव्हिजिंग पेन व्हर्टिकल निवडा.

• तुमच्या डॉक्युमेंट्स च्या खाली सर्व बदलांची लिस्ट पाहण्यासाठी हॉरिजॉन्टल सबखंडाचे रिव्यु करा निवडा.

सर्व ट्रॅक केलेले बदल तुमच्या डॉक्युमेंट्स तून काढून टाकले गेले आहेत हे तपासण्यासाठी तुम्ही रिव्यु पॅन एक सुलभ साधन म्हणून वापरू शकता जेणेकरून ते तुमचे डॉक्युमेंट्स पाहू शकतील अशा इतरांना दिसतील. रिव्यु सबखंडाच्या शीर्षस्थानी सारांश विपार्ट व्ह मान ट्रॅक केलेल्या बदलांची अचूक नंबर आणि तुमच्या डॉक्युमेंट्स त राहिलेल्या कमेंट प्रदर्शित करतो.

रिव्यु पॅन तुम्हाला कमेंट बबलमध्ये बसत नसलेल्या लांब टिप्पण्यांचे रिव्यु करण्यास देखील अनुमती देते.

**टीप:** डॉक्युमेंट्स किंवा कमेंट बलून प्रमाणे रिव्यु पॅन, तुमच्या डॉक्युमेंट्स त बदल करण्यासाठी सर्वोत्तम साधन नाही. टेक्स्ट किंवा कमेंट हटविण्याऐवजी किंवा रिव्यु पॅन इतर बदल करण्याऐवजी, डॉक्युमेंट्स तील सर्व संपादकीय बदल करा. नंतर रिव्यु पॅन बदल व्ह मान होतील.

महत्वाचे: कमेंट यापुढे ट्रॅक चेंज कार्याचा पार्ट नाहीत

## मेल मर्ज करा (Perform mail merge)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- ऑब्जेक्ट इन्सर्ट
- मेल मर्ज
- टेम्पलेट्स.

ऑब्जेक्ट, मॅक्रो, मेल मर्ज, टेम्पलेट्स आणि इतर साधनांसह कार्य करणे

### ऑब्जेक्ट:

तुम्ही वापरत असलेल्या Word किंवा Outlook च्या आवृत्तीच्या आधारावर, तुम्ही Word डॉक्युमेंट्स किंवा ईमेल संदेशामध्ये लिंक करून किंवा एम्बेड करून विविध वस्तू (जसे की PDF फाइल्स, Excel चार्ट किंवा वर्कशीट्स किंवा PowerPoint प्रेझेंटेशन) घालू शकता. ऑब्जेक्ट घालण्यासाठी, इन्सर्ट टॅबवरील ऑब्जेक्टवर क्लिक करा.

तुमच्या Word डॉक्युमेंट्स त किंवा ईमेल संदेशामध्ये समाविष्ट केलेली नवीन फाइल तयार करण्यासाठी:

- 1 ऑब्जेक्ट डायलॉग बॉक्समध्ये, नवीन तयार करा टॅबवर क्लिक करा आणि नंतर ऑब्जेक्ट टाइप लिस्ट मधून एक पर्याय निवडा.

त्या फाइल प्रकारासाठी डीफॉल्ट प्रोग्राम उघडतो, जिथे तुम्ही तुम्हाला पाहिजे असलेला कोणताही टेक्स्ट किंवा डेटा प्रविष्ट करू शकता. जेव्हा तुम्ही प्रोग्राम बंद करता, तेव्हा कोणतीही जोडलेली कंटेंट किंवा बदल तुमच्या Word डॉक्युमेंट्स त किंवा ईमेल संदेशात दिसतात.

- 2 तुम्हाला नवीन फाइल तुमच्या नवीन फाइलच्या पहिल्या पेज ऐवजी क्लिक करण्यायोग्य चिन्ह म्हणून दिसावी असे वाटत असल्यास, चिन्ह म्हणून प्रदर्शित करा निवडा. जर हा चेक बॉक्स निवडला असेल, तर तुम्ही बदला चिन्हावर क्लिक करून वेगळे चिन्ह निवडू शकता.

### विद्यमान फाइल लिंक किंवा एम्बेड करा

आधीपासून तयार केलेल्या ऑब्जेक्टला लिंक किंवा एम्बेड करण्यासाठी:

- 1 ऑब्जेक्ट डायलॉग बॉक्समध्ये, फाइलमधून तयार करा टॅब निवडा, आणि नंतर तुम्हाला समाविष्ट करायची असलेली फाइल सर्च ष्यासाठी ब्राउझ क्लिक करा.
- 2 सोर्स फाइलशी लिंक करण्यासाठी, ती तुमच्या Word डॉक्युमेंट्स त किंवा ईमेल संदेशामध्ये एम्बेड करण्याऐवजी, फाइलशी लिंक निवडा.
- 3 जर तुम्हाला फाइलच्या पहिल्या पृष्ठाऐवजी, समाविष्ट केलेली फाइल क्लिक करता येणार्या आयकॉन म्हणून दिसावी असे वाटत असल्यास, आयकॉन म्हणून प्रदर्शित करा निवडा. जर हा चेक बॉक्स निवडला असेल, तर तुम्ही बदला चिन्हावर क्लिक करून वेगळे चिन्ह निवडू शकता.

### मेल मर्ज:

तुमच्याकडे नावे आणि पत्त्यांची एक लांबलचक यादी आहे ज्यावर तुम्हाला पत्रे पाठवायची आहेत? मेल मर्ज प्रक्रिया वर्ड डॉक्युमेंटला डेटा स्त्रोतासह एकत्रित करते जे वैयक्तिक वाटणारी अक्षरे क्लिकली तयार करते.

### मेल मर्ज व्याख्या

**स्टार्ट डॉक्युमेंट्स :** (मुख्य डॉक्युमेंट्स ) एक डॉक्युमेंट्स ज्यामध्ये प्रत्येक विलीन केलेल्या डॉक्युमेंट्स साठी समान इनफार्मेशन असते. सुरुवातीच्या डॉक्युमेंट्स व्हेरिअबल इनफार्मेशन साठी फील्डची नावे असतात, जसे की नावे आणि एड्रेस समाविष्ट केले जातात.

**डेटा सोर्स किंवा प्राप्तकर्त्यांची यादी:** मेल मर्ज दरम्यान मुख्य डॉक्युमेंट्स त समाविष्ट करण्याची इनफार्मेशन असलेली फाइल. उदाहरणार्थ, त्यात लोकांची नावे आणि एड्रेस असलेले रेकॉर्ड आहेत ज्यांना मेल मर्ज पत्र पाठवले जाते. एक्सेल स्प्रेडशीट्स, एक्सेस डेटाबेसेस किंवा वर्ड डॉक्युमेंट टेबल ही डेटा स्रोतांची उत्तम उदाहरणे आहेत.

**फील्ड:** इनफार्मेशन चा स्पेसिफिक पार्ट संचयित करणारी डेटा रेन्ज . उदाहरणार्थ, "LastName" फील्डमध्ये फक्त लोकांची आडनावे असतील.

**रेकॉर्ड:** रेकॉर्ड हा डेटा फील्डचा संपूर्ण संच आहे जो एकाच गोष्टीशी किंवा व्यक्तीशी संबंधित असतो. उदाहरणार्थ, एका रेकॉर्डमध्ये एखाद्या व्यक्तीचे नाव आणि आडनाव, एड्रेस , फोन नंबर आणि जन्मdate समाविष्ट असेल.

**फील्ड मर्ज करा:** एक मर्ज फील्ड आहे जिथे तुम्ही डेटा स्त्रोतामधील इनफार्मेशन मुख्य डॉक्युमेंट्स त समाविष्ट करू इच्छिता. विलीन फील्ड शेवट " (" ") सह दिसतात. एक उदाहरण असेल: प्रिय "प्रथम नाव".

**एड्रेस ब्लॉक:** मर्ज फील्डचा एक समूह जो मेल मर्ज डॉक्युमेंट्स त एड्रेस बनवतो. उदाहरणार्थ, एकच एड्रेस नाव, रस्त्याचा एड्रेस , शहर, राज्य आणि पिन कोड यांनी बनलेला असतो. वर्ड ऑटोमैटिक सर्व योग्य अॅड्रेस फील्ड एकाच वेळी समाविष्ट करू शकतो, त्यामुळे तुम्हाला स्वतःला पाच किंवा सहा मर्ज फील्ड घालण्याची गरज नाही.

**ग्रीटिंग लाइन:** मर्ज फील्डचा एक समूह जो मेल मर्ज डॉक्युमेंट्स ची ग्रीटिंग लाइन बनवतो, जसे की "प्रिय श्री. मॅकडोनाल्ड". शब्द ऑटोमैटिक सर्व योग्य ग्रीटिंग टेक्स्ट , टाईटल आणि नाव फील्ड एकाच वेळी समाविष्ट करू शकतो, त्यामुळे तुम्हाला टेक्स्ट आणि आवश्यक मर्ज फील्ड स्वतः घालण्याची गरज नाही.

**हेडर रो** :डेटा सोर्स इनफार्मेशन टेबलमध्ये स्टोर केली जाते. टेबल ची पहिली रो हेडर रो आहे आणि त्यात डेटा स्त्रोतासाठी फील्डची नावे आहेत. उदाहरणार्थ, FirstName, LastName आणि Address हे हेडर रो आहेत.

### सेट करा आणि डॉक्युमेंट्स टाइप निवडा

मेल मर्ज प्रक्रिया सुरू करण्यासाठी, आपण प्रथम कोणत्या टाइप चे डॉक्युमेंट्स तयार करू इच्छिता ते निवडणे आवश्यक आहे.

- 1 मेलिंग टॅबवर क्लिक करा.
- 2 स्टार्ट मेल मर्ज बटणावर क्लिक करा.
- 3 स्टेप बाय स्टेप मेल मर्ज विझार्ड निवडा.

मेल मर्ज पॅन उजवीकडे दिसेल, मेल मर्ज द्वारे तुम्हाला चालण्यासाठी तयार आहे.

- 4 तयार करण्यासाठी डॉक्युमेंट्स चा टाइप निवडा.
- 5 पुढील क्लिक करा: डॉक्युमेंट्स सुरू करा.

मेल मर्ज विझार्ड पुढच्या टप्प्यावर जातो.

### एक डॉक्युमेंट्स निवडा

ही पुढील स्टेप म्हणजे स्टार्टिंग डॉक्युमेंट्स निवडणे.

- 1 स्टार्ट डॉक्युमेंट्स निवडा.

तुम्ही मेल मर्ज चा आधार म्हणून करंट डॉक्युमेंट्स वापरू शकता किंवा त्याऐवजी तुम्ही टेम्पलेट किंवा विद्यमान डॉक्युमेंट्स निवडू शकता.

- 2 पुढील क्लिक करा: प्राप्तकर्ते (Recipients) निवडा.

मेल मर्ज विझार्ड चरण 3 वर जातो.

### प्राप्तकर्ते(Recipients) निवडा

आता, तुम्हाला तुमच्या पत्त्यांची यादी कोठून मिळेल ते निवडणे आवश्यक आहे. हे उदाहरण डेटाबेसमधील विद्यमान लिस्ट वापरते, परंतु तुम्ही Outlook संपर्क देखील निवडू शकता किंवा स्वतःची लिस्ट स्वतः तयार करू शकता.

- 1 विद्यमान लिस्ट वापरा निवडा.
- 2 ब्राउझ वर क्लिक करा.
- 3 तुमची डेटा फाइल निवडा.

मेल मर्ज प्राप्तकर्त्यांची(Recipients) डेटा फाइल डेटाबेस फाइल, एक्सेल स्प्रेडशीट, दुसरा वर्ड डॉक्युमेंट किंवा इतर टाइप च्या डेटा फाइल्समध्ये असू शकते.

- 4 उघडा क्लिक करा.

मेल मर्ज प्राप्तकर्ता डायलॉग बॉक्स वापरले जाणारे एड्रेस दाखवतो. तुम्ही डेटा सोर्स म्हणून एक्सेल स्प्रेडशीट वापरत असल्यास, तुम्हाला एड्रेस

असलेली वर्कशीट निवडण्यासाठी देखील सूचित केले जाईल.

तुम्हाला वापरायचा नसलेला एड्रेस असल्यास, तुम्ही तो अनचेक करू शकता.

- 5 योग्य प्राप्तकर्ते निवडले असल्याची खात्री करा आणि ओके क्लिक करा.

- 6 पुढील क्लिक करा: तुमचे पत्र लिहा.

मेल मर्ज विझार्ड चरण 4 वर जातो.

### तुमचे पत्र लिहा

मुख्य डॉक्युमेंट्स सेट केल्यानंतर आणि प्राप्तकर्त्यांची लिस्ट कनेक्ट केल्यानंतर आणि एडिट केल्यानंतर, तुम्ही डॉक्युमेंट्स त मर्ज फील्ड घालण्यास तयार आहात. मर्ज फील्ड हे प्राप्तकर्त्यांच्या यादीतील अद्वितीय इनफार्मेशन साठी डॉक्युमेंट्स तील प्लेसहोल्डर आहेत.

जेव्हा तुम्ही मुख्य डॉक्युमेंट्स त मर्ज फील्ड ठेवता, तेव्हा त्या फील्डमधील इनफार्मेशन त्या प्राप्तकर्त्यांसाठी अद्वितीय असलेल्या डॉक्युमेंट्स साठी दिसून येईल.

- 1 तुम्हाला जिथे इनफार्मेशन हवी आहे तिथे क्लिक करा.

- 2 प्लेसहोल्डर पर्यायांपैकी एक निवडा.

तुम्ही विझार्डमधून किंवा रिबनवरील फील्ड लिहा आणि इन्सर्ट गटातून मर्ज फील्ड जोडू शकता:

- **एड्रेस ब्लॉक**: प्राप्तकर्त्यांची नावे आणि एड्रेस घालण्यासाठी हे फील्डचे संयोजन आहे.
- **ग्रीटिंग लाइन**: हे ग्रीटिंग लाइनमध्ये प्राप्तकर्त्यांचे नाव समाविष्ट करण्यासाठी फील्डचे संयोजन आहे.
- **मर्ज फील्ड इन्सर्ट**: जेव्हा तुम्ही या बटणावर क्लिक करता, तेव्हा तुम्ही समाविष्ट करू शकता अशा अतिरिक्त मर्ज फील्डची लिस्ट दिसते.

प्लेसहोल्डर कस्टमाइज करा.

ओके क्लिक करा.

(पर्यायी) तुम्हाला समाविष्ट करायचे असलेले प्रत्येक मर्ज फील्ड जोडण्यासाठी स्टेप्स ची पुनरावृत्ती करा. पुढील क्लिक करा: तुमच्या अक्षरांचे प्रीव्यू करा.

मेल मर्ज विझार्ड चरण 5 वर जातो.

### तुमच्या मेल मर्जचे प्रीव्यू करा

काहीवेळा, केवळ मर्ज फील्डची नावे पाहण्याऐवजी, कागदपत्रात टाकल्यानंतर डेटा कसा दिसेल हे पाहणे सबयुक्त ठरते.

मेल मर्ज पूर्ण करण्यापूर्वी मेल मर्ज कसे दिसेल याचे तुम्ही सहजपणे प्रीव्यू करू शकता. तुम्हाला हवे तसे रिझल्ट दिसतील याची खात्री करण्यासाठी हे प्रोत्साहन दिले जाते.

- 1 प्रत्येक विलीन केलेल्या डॉक्युमेंट्स चे प्रीव्यू करण्यासाठी मेल मर्ज सबखंडातील बाण बटणे वापरा.

तुम्ही रिबनवरील प्रीव्यू रिझल्ट गटातील बाण बटणे देखील वापरू शकता. स्पेसिफिक प्राप्तकर्त्याचा सर्च घेण्यासाठी प्रीव्यू रिझल्ट गटातील किंवा मेल मर्ज सबखंडातील प्राप्तकर्ता शोधा बटणावर क्लिक करा.

2 पुढील क्लिक करा: मर्ज पूर्ण करा.

मेल मर्ज विझार्ड अंतिम टप्प्यावर जातो.

### मर्ज पूर्ण करा

एकदा तुम्ही प्राप्तकर्त्याची यादी जोडली आणि मर्ज फील्डसह डॉक्युमेंट्स भरला की, प्रत्येक प्राप्तकर्त्यासाठी डॉक्युमेंट्स ची वेगळी आवृत्ती बनवून विलीन करा.

तुम्ही मेल मर्ज पूर्ण करू शकता असे दोन भिन्न पाथ आहेत:

- वैयक्तिक डॉक्युमेंट्स एडिट करा: मेल मर्जचे रिझल्ट नवीन डॉक्युमेंट्स त ठेवा. तुम्ही मेल मर्ज चे रिझल्ट एडिट करण्यासाठी आणि इतर कोणत्याही डॉक्युमेंट्स प्रमाणेच ते सेव्ह आणि प्रिंटेड करण्यासाठी मोकळे आहात.
- डॉक्युमेंट्स प्रिंटेड करा: रेकॉर्ड विलीन करते आणि ते थेट प्रिंटरकडे पाठवते.

1 मेल मर्ज पूर्ण करण्यासाठी तुम्हाला वापरायचा असलेला पर्याय निवडा.

तुम्ही रिबनवरील फिनिश आणि मर्ज बटणावर क्लिक करू शकता आणि तेथे मर्ज पर्याय निवडा.

तुम्हाला कोणते रेकॉर्ड विलीन करायचे ते निवडण्याचा पर्याय देखील दिला जातो. तुम्ही लिस्ट तील सर्व रेकॉर्ड विलीन करू शकता, फक्त सध्या प्रदर्शित केलेले रेकॉर्ड किंवा रेन्ज स्पेसिफाइड करू शकता.

2 तुम्हाला विलीन करायचे असलेले रेकॉर्ड निवडा.

3 ओके क्लिक करा.

Word मुख्य डॉक्युमेंट्स आणि डेटा स्रोतातील इनफार्मेशन एका नवीन Word डॉक्युमेंट्स त विलीन करतो, किंवा तुम्ही निवडलेल्या पर्यायावर आधारित, विलीन करून प्रिंटरला पाठवतो.

### टेम्पलेट्स

#### वर्ड डॉक्युमेंटमध्ये टेम्पलेट कसे तयार करावे

टेम्पलेट हा पूर्व-निर्मित डॉक्युमेंट्स आहे ज्यामध्ये मॅक्रो, हेडर आणि फुटनोट, कस्टम डिक्शनरी, सिन्टेक्स, प्रतिमा आणि ऑटोटेक्स्ट एंटी यांसारखे काही आधीच तयार केलेले स्पेसिफिक फॉर्मॅटिंग पर्याय समाविष्ट आहेत.

समान कंटेंट आणि संरचनेसह डॉक्युमेंट्स तयार करताना टेम्पलेट आपल्याला वेळ वाचविण्यात मदत करते.

#### वर्ड डॉक्युमेंटमध्ये टेम्पलेट तयार करा

वर्ड डॉक्युमेंटमध्ये टेम्पलेट तयार करण्यासाठी खाली नमूद केलेल्या सर्वात सोप्या पायऱ्या फॉलो करा -

**स्टेप 1 :** वर्ड डॉक्युमेंट उघडा.

**स्टेप 2:** रिबनवरील फाइल टॅबवर जा आणि नवीन पर्यायावर क्लिक करा.

**स्टेप 3:** रिक्त डॉक्युमेंट्स वर क्लिक करा.

**स्टेप 4:** तुम्ही पुढे टेम्पलेट म्हणून वापरत असलेली इष्ट कंटेंट ऑड करा.

#### मायक्रोसॉफ्ट वर्ड टेम्पलेट सेव्ह करा

एकदा तुम्ही Microsoft Word मध्ये टेम्पलेट तयार केल्यावर, तुम्ही ते तुमच्या पुढील वापरासाठी सेव्ह करू शकता.

Microsoft Word मध्ये, तुम्ही Microsoft Word टेम्पलेट .dotx, .dot, किंवा .dotm फाइल फॉर्मॅट म्हणून सेव्ह करू शकता.

मायक्रोसॉफ्ट वर्डमध्ये टेम्पलेट सेव्ह करण्यासाठी खालील सूचना फॉलो करा -

- 1 तुम्ही ज्या वर्ड डॉक्युमेंटमध्ये टेम्पलेट तयार करता ते उघडा.
- 2 रिबनवरील फाइल टॅबवर जा. स्क्रीनवर पर्यायांची यादी दिसेल ज्यामध्ये Save As वर क्लिक करा.
- 3 जिथे तुम्हाला फाइल सेव्ह करायची आहे ते पोजिशन ब्राउझ करा. Save as type सह ड्रॉप-डाउन असोसिएट वर क्लिक करा. वर्ड टेम्पलेट (\*.dotx) निवडा आणि स्क्रीनच्या तळाशी असलेल्या सेव्ह बटणावर क्लिक करा.

आता, तुमचा वर्ड टेम्पलेट वर्ड डॉक्युमेंटमध्ये सेव्ह झाला आहे हे तुम्ही पाहू शकता.

#### वर्ड टेम्पलेट एडिट करा

एकदा तुम्ही Microsoft Word मध्ये टेम्पलेट तयार केल्यावर, तुम्ही तुमच्या गरजेनुसार ते एडिट देखील करू शकता आणि नंतर ते आगामी, समान Word डॉक्युमेंट्स साठी सेव्ह करू शकता.

वर्ड टेम्पलेट एडिट करण्यासाठी खालील सर्वात सोप्या पायऱ्या आहेत -

**स्टेप 1:** वर्ड डॉक्युमेंटच्या वरच्या डाव्या कोपऱ्यात असलेल्या फाइल टॅबवर जा आणि ओपन बटणावर क्लिक करा.

**स्टेप 2:** स्क्रीनवर एक ओपन डायलॉग बॉक्स दिसेल. तुम्ही पूर्वी तयार केलेले टेम्पलेट सेव्ह केलेले पोजिशन ब्राउझ करा आणि स्क्रीनच्या तळाशी असलेल्या ओपन बटणावर क्लिक करा.

**स्टेप 3:** तुम्हाला तुमच्या टेम्पलेटमध्ये ऑड करायचे असलेले बदल करा.

**स्टेप 4:** एकदा आपण आपल्या टेम्पलेटमध्ये आपले सर्व इच्छित बदल केले की, आपण आपले टेम्पलेट सेव्ह करू शकता आणि ते बंद करू शकता.

#### मायक्रोसॉफ्ट वर्ड टेम्पलेट वापरा

मायक्रोसॉफ्ट वर्ड इन-बिल्ट वर्ड टेम्पलेट्स देखील प्रदान करते जे तुम्ही तुमच्या वर्ड डॉक्युमेंटवर वापरू शकता.

मायक्रोसॉफ्ट वर्ड टेम्पलेट वापरण्यासाठी खालील पायऱ्या आहेत -

**स्टेप 1 :** नवीन Word डॉक्युमेंट्स उघडा.



**स्टेप 2:** रिबनवरील फाइल टॅबवर जा. स्क्रीनवर फाइल मेनू पर्याय दिसतील. नवीन वर क्लिक करा.

**स्टेप 3:** एक सबलब्ध टेम्पलेट विंडो दिसेल. वैयक्तिक वर क्लिक करा.

**स्टेप 4:** वर्ड टेम्पलेट्सची लिस्ट स्क्रीनवर दिसते. तुम्ही लागू करू इच्छित टेम्पलेट निवडा आणि त्यावर डबल-क्लिक करा.

आता, तुम्ही निवडलेले टेम्पलेट वर्ड डॉक्युमेंटमध्ये जोडले असल्याचे पाहू शकता.

© NIMI  
NOT TO BE REPUBLISHED

## MS Excel मध्ये फाइल्स उघडा (Open files in MS Excel)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एक्सेलचा परिचय
- सेल रेफरेन्स आणि लिंकिंग शीट
- वर्कशीटवर प्रिंट एरिया सेट करा किंवा क्लियर करा.

हा मायक्रोसॉफ्टने विकसित केलेला स्प्रेडशीट प्रोग्राम आहे. एक्सेल कॉलम आणि रो मध्ये डेटा आयोजित करते आणि तुम्हाला गणितीय फंक्शन करण्यास अनुमती देते. हे Windows, macOS, Android आणि iOS वर चालते.

पहिली आवृत्ती 1985 मध्ये रिलीझ झाली होती आणि अनेक वर्षांमध्ये ती अनेक बदलांमधून गेली आहे. तथापि, मुख्य कार्यक्षमता बहुतेक समान राहते.

**एक्सेल सामान्यतः यासाठी वापरले जाते:**

- विश्लेषण
- डेटा विश्लेषण
- डेटा भरणे
- व्हिज्युअल आणि आलेख
- डेटा मॅनेजमेन्ट
- प्रोग्रामिंग
- लेखा
- आर्थिक मॉडेलिंग
- बजेटिंग
- आणि बरेच काही, बरेच काही!

**एक्सेल का वापरावे?**

- हा जगातील सर्वात लोकप्रिय स्प्रेडशीट प्रोग्राम आहे
- शिकणे आणि सुरुवात करणे सोपे आहे.
- कौशल्य कमाल मर्यादा उच्च आहे, याचा अर्थ असा की तुम्ही अधिक एडवॉन्स गोष्टी करू शकता जसे तुम्ही चांगले व्हाल
- हे काम आणि दैनंदिन जीवनात दोन्ही वापरले जाऊ शकते, जसे की कौटुंबिक बजेट तयार करण्यासाठी
- याला समुदायाचा मोठा पाठिंबा आहे
- हे सतत Microsoft द्वारे समर्थित आहे
- टेम्पलेट आणि फ्रेमवर्क स्वतः आणि इतरांद्वारे पुन्हा वापरता येऊ शकतात, निर्मिती खर्च कमी करतात

**दुसऱ्या वर्कशीटसाठी सेल रेफरेन्स तयार करा: क्लिक करा**

ज्या सेलमध्ये तुम्ही फॉर्मूला प्रविष्ट करू इच्छिता. , type = (समान चिन्ह) आणि आपण वापरू इच्छित फॉर्मूला . संदर्भित करण्यासाठी वर्कशीटसाठी टॅबवर क्लिक करा. संदर्भित करण्यासाठी सेल किंवा सेलची रेन्ज निवडा.

- 1 ज्या सेलमध्ये तुम्ही फॉर्मूला प्रविष्ट करू इच्छिता त्या सेलवर क्लिक करा.

2 फॉर्म्युला बारमध्ये,  type = (समान चिन्ह).

3 खालीलपैकी एक करा

- **एक किंवा अधिक सेलचा रेफरेन्स द्या:** रेफरेन्स तयार करण्यासाठी, त्याच वर्कशीटवर सेल किंवा सेलची रेन्ज निवडा.
- तुम्ही निवड हलवण्यासाठी सेल निवडीची बॉर्डर ड्रॅग करू शकता किंवा निवड विस्तृत करण्यासाठी बॉर्डरचा कोपरा ड्रॅग करू शकता.
- **परिभाषित नावाचा रेफरेन्स द्या:** परिभाषित नावाचा रेफरेन्स तयार करण्यासाठी, खालीलपैकी एक करा:
- नाव टाइप करा.
- F3 दाबा, नाव पेस्ट करा बॉक्समधील नाव निवडा, आणि नंतर ओके क्लिक करा.

**टीप: कलरकोड केलेल्या सीमेवर कोणताही चौरस कोपरा नसल्यास, रेफरेन्स नामित रेन्ज चा आहे.**

4 खालीलपैकी एक करा:

- तुम्ही एकाच सेलमध्ये रेफरेन्स तयार करत असल्यास, एंटर दाबा.
- तुम्ही अरे फॉर्म्युला (जसे की A1:G4) मध्ये रेफरेन्स तयार करत असल्यास, Ctrl+Shift+Enter दाबा.

रेफरेन्स सिंगल सेल किंवा सेलची रेन्ज असू शकते आणि अरे फॉर्मूला सिंगल किंवा मल्टिपल परिणामांची गणना करणारा असू शकतो.

**वर्कशीटवर प्रिंट एरिया सेट करा किंवा क्लियर करा:** तुम्ही वर्कशीटवर स्पेसिफिक निवड वारंवार प्रिंटेड केल्यास, तुम्ही प्रिंट एरिया परिभाषित करू शकता ज्यामध्ये फक्त ती निवड समाविष्ट आहे. प्रिंट एरिया हे सेलच्या एक किंवा अधिक रेन्ज असतात जे तुम्ही संपूर्ण वर्कशीट प्रिंटेड करू इच्छित नसताना प्रिंटेड करण्यासाठी नियुक्त करता. जेव्हा तुम्ही प्रिंट एरिया परिभाषित केल्यानंतर वर्कशीट प्रिंटेड करता तेव्हा फक्त प्रिंट एरिया छापले जाते. आवश्यकतेनुसार प्रिंट एरिया विस्तृत करण्यासाठी तुम्ही सेल जोडू शकता आणि संपूर्ण वर्कशीट प्रिंटेड करण्यासाठी तुम्ही प्रिंट एरिया क्लियर करू शकता.

वर्कशीटमध्ये अनेक प्रिंट एरिया असू शकतात. प्रत्येक प्रिंट एरिया स्वतंत्र पेज म्हणून प्रिंटेड करेल.

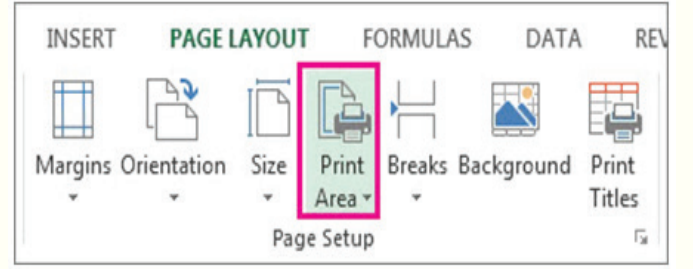
## एक किंवा अधिक प्रिंट एरिया सेट करा

1 वर्कशीटवर, तुम्ही प्रिंट एरिया म्हणून परिभाषित करू इच्छित सेल निवडा.

**टीप:** मल्टिपल प्रिंट एरिया सेट करण्यासाठी, Ctrl की दाबून ठेवा आणि तुम्हाला प्रिंटेड करायचे असलेल्या भागात क्लिक करा. प्रत्येक प्रिंट एरिया त्याच्या स्वतःच्या पेज वर छापतो.

2 पेज लेआउट टॅबवर, पेज सेटअप गटामध्ये, प्रिंट एरिया क्लिक करा आणि नंतर प्रिंट एरिया सेट करा क्लिक करा.

टीप: तुम्ही सेट केलेले प्रिंट एरिया तुम्ही कार्यपुस्तिका सेव्ह करता तेव्हा सेव्ह केले जाते.



© NIMI  
NOT TO BE REPUBLISHED

## डेटा हाताळा (Manipulate data)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- डेटा हाताळा (Manipulate data)
- सेल रेन्ज .

डेटा विश्लेषण हे एक आव्हानात्मक कार्य आहे, स्पेशल तः जर तुमच्याकडे डेटा हाताळणी कौशल्ये नसेल. या लेखात, आम्ही डेटा विश्लेषणासाठी एक्सेलमध्ये वापरल्या जाणाऱ्या काही सर्वात सामान्य डेटा मॅनिप्युलेशन तंत्रांवर चर्चा करू.

या तंत्रांचे सामर्थ्य काही वास्तविक जीवनातील उदाहरणे वापरून दाखवले जाईल. चर्चा केलेल्या 9 सामान्य डेटा हाताळणी तंत्रे आहेत:

- 1 फिल्टरिंग
- 2 वर्गीकरण
- 3 ग्रुप बद्ध करणे
- 4 पिव्होटिंग
- 5 ट्रान्सपोजिंग
- 6 डेटा टाइप बदलणे
- 7 कॉलम आणि रो जोडणे
- 8 कॉलम किंवा रो चे नाव देणे
- 9 कॉलम किंवा रो इन्सर्ट करणे.

यातील प्रत्येक तंत्र तुम्हाला तुमचा डेटा आणि ते कसे कार्य करते याबद्दल अधिक चांगल्या प्रकारे समजून देईल - विविध टाइप च्या व्हिज्युअलायझेशनपासून ते आउटलायर्स एक्सप्लोर करण्यापर्यंत. या सौप्या युक्त्या केवळ तुमची कार्यक्षमतेत सुधारणा करणार नाहीत तर ज्या लोकांना Excel माहित नाही त्यांना तुम्ही काय करत आहात हे समजून घेणे देखील सोपे करेल.

### डेटा टाइप बदलणे

एकदा तुम्ही ही कमांड सिलेक्ट केल्यानंतर, "डेटा टाइप निवडा" विंडो दिसेल. डेटाच्या तीन सामान्य रेन्ज आहेत: केवळ टेक्स्ट डेटा; केवळ अंकीय डेटा; आणि मिश्रित डेटा (टेक्स्ट आणि अंकीय).

यापैकी प्रत्येक रेन्ज मधून एक किंवा अधिक पर्याय निवडून तुम्ही तुमची निवड आणखी परिष्कृत करू शकता: टेक्स्ट ; नंबर त्मक; date वेळ; लॉजिकल ; आणि ऑब्जेक्ट लिंकिंग आणि एम्बेडिंग

**कॉलम आणि रो जोडणे:** तुमच्या डेटामध्ये कॉलम किंवा रो जोडणे हा तुमचे कार्य अधिक कार्यक्षम बनवण्याचा एक उत्तम पाथ आहे. उदाहरणार्थ, जर तुम्ही वेगवेगळ्या विषयांवरील डेटाच्या टेबल वर काम करत असाल आणि त्यांची उत्तरे एकमेकांच्या संदर्भात पहायची असतील,

तर तुमच्यासाठी (आणि तुम्ही ज्यांच्याशी डेटा शेअर करत आहात) ते अधिक सोयीचे होईल. एकाच कॉलम त दोन्ही उत्तरे.

**कॉलम आणि रो चे नाव देणे:** Excel मध्ये कॉलम आणि रो ना नाव देण्यासाठी, प्रथम नामकरण आवश्यक असलेल्या सेल निवडा. त्यानंतर Data > Data Tools > Name Columns and Rows वर जा आणि पहिल्या डायलॉग बॉक्समध्ये पहिल्या सेलचे नाव टाइप करा. तुमच्या सर्व सेलची नावे येईपर्यंत टाइप करणे किंवा क्लिक करणे सुरू ठेवा.

### कॉलम किंवा रो चे नाव देणे

स्प्रेडशीटमधील प्रत्येक कॉलम आणि रो चे डीफॉल्ट नाव असते, परंतु ही नावे बदलली जाऊ शकतात. जेव्हा तुम्ही डेटा सारांशित करत असाल आणि समान कॉलम किंवा रो टाईटल सातत्याने लागू करू इच्छित असाल तेव्हा हे सबयुक्त आहे.

कॉलम चे नाव बदलण्यासाठी, त्या कॉलम तील कोणत्याही सेलवर उजवे-क्लिक करा आणि "कॉलम चे नाव" निवडा. नवीन नाव टाइप करा आणि एंटर दाबा. रो चे नाव बदलण्यासाठी, त्या रो मधील कोणत्याही सेलवर उजवे-क्लिक करा आणि " रो लेबले" निवडा. नवीन नाव टाइप करा आणि एंटर दाबा.

### कॉलम किंवा रो इन्सर्ट करणे

Excel मधील सर्वात सोप्या डेटा हाताळणी तंत्रांपैकी एक म्हणजे कॉलम किंवा रो इन्सर्ट करणे.

हे तंत्र तुम्हाला तुमच्या स्प्रेडशीटमध्ये अधिक कॉलम किंवा रो जोडून तुमच्या डेटाचे अधिक स्पष्टतेने आणि अचूकतेने विश्लेषण करू देते. तुमच्या डेटाचे वेगवेगळे पैलू जसे की वेगवेगळी वर्षे, प्रदेश, उत्पादने इत्यादी दाखवण्यासाठी याचा वापर केला जाऊ शकतो.

हे तंत्र वापरले जाऊ शकते अशी दोन उदाहरणे आहेत:

- 1 तुम्हाला एका स्पेसिफिक प्रदेशासाठी वेगवेगळ्या वर्षांची कम्प्यारीजन करायची आहे.
- 2 एखादे उत्पादन विविध क्षेत्रांमध्ये कसे कार्य करते ते तुम्हाला पहायचे आहे.

### निष्कर्ष

डेटा मॅनिप्युलेशन हे वेळखाऊ आणि क्लिष्ट काम असू शकते. परंतु योग्य तंत्र आपल्याला बराच वेळ वाचविण्यात आणि चुका टाळण्यास मदत करू शकते. या लेखातील 9 बेसिक तंत्रे तुम्हाला Excel मध्ये डेटा हाताळणी नेव्हिगेट करण्यात मदत करू शकतात.

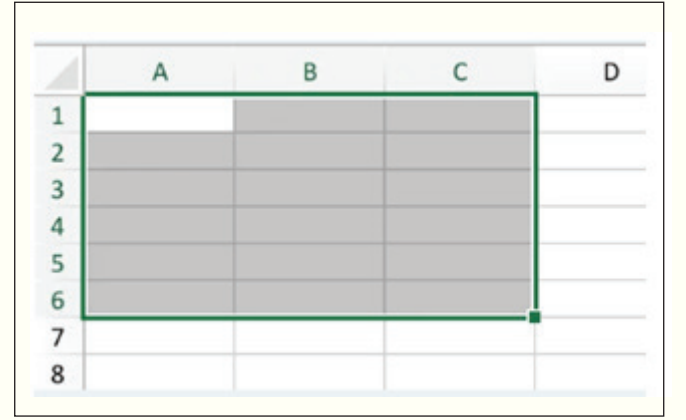
## सेल रेन्ज म्हणजे काय

एक्सेल फाईलमधील सेल रेन्ज निवडलेल्या सेलचा संग्रह आहे. ही रेन्ज सामान्यतः सममितीय (चौरस) असते, परंतु स्वतंत्र सेल समान असू शकतात. सेल रेन्ज चा रेफरेन्स फॉर्मूला मध्ये देखील केला जाऊ शकतो.

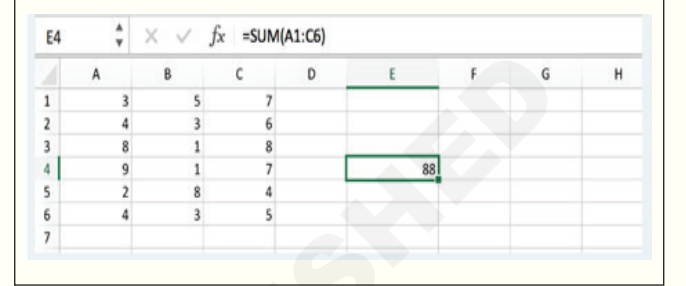
स्प्रेडशीटमध्ये, रेन्ज च्या वरच्या डाव्या सेलच्या (किमान व्हॅल्यु ) आणि रेन्ज च्या खालच्या उजव्या सेलच्या (कमाल व्हॅल्यु ) संदर्भाद्वारे सेल रेन्ज परिभाषित केली जाते. अखेरीस या निवडीमध्ये वेगळे सेल जोडले जाऊ शकतात, नंतर Excel मध्ये, किमान आणि कमाल व्हॅल्यु समाविष्ट केले जातात. हे गणितीय रेन्ज पेक्षा वेगळे आहे, ज्यामध्ये ते कमाल आणि किमान व्हॅल्यु मधील व्हॅल्यु चा संग्रह आहे.

एक सममितीय सेल रेन्ज खालीलप्रमाणे दिसू शकते. या रेन्ज साठी नोटेशन आहे (A1:C6); वरच्या डाव्या सेल A1 पासून खालच्या उजव्या सेल C6 पर्यंत.

सेल रेन्ज फॉर्मूला मध्ये वापरली जाऊ शकते, उदाहरणार्थ निवडलेल्या सेलमधील व्हॅल्यु ची बेरीज मोजण्यासाठी. सेल रेन्ज (A1:C6) मधील सर्व व्हॅल्यु च्या बेरीजसाठी नोटेशन =SUM(A1:C6) आहे.



	A	B	C	D
1				
2				
3				
4				
5				
6				
7				
8				



E4	A	B	C	D	E	F	G	H
	3	5	7					
	4	3	6					
	8	1	8					
	9	1	7		88			
	2	8	4					
	4	3	5					

## टेबल तयार करा आणि फॉर्मॅटिंग करा (Create and format tables)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टेबल तयार करा आणि फॉर्मॅटिंग करा
- क्रमवारी (सॉर्टिंग) लावा आणि फिल्टर करा
- टेबलला टेक्स्ट मध्ये रूपांतरित करा.

### टेबल तयार करा आणि फॉर्मॅटिंग करा

तुमची रेन्ज टेबल म्हणून फॉर्मॅटिंग करणे Excel ला सांगते की त्या रो आणि कॉलम सर्व संबंधित आहेत आणि पहिल्या रांगेत हेडर आहेत. आणि असे केल्याने, आपल्या रेन्ज ला आता अर्थ आहे. एक्सेल ते चांगले समजते. आणि त्याबरोबर, बरेच अतिरिक्त फायदे जन्माला येतात.

### फॉर्मॅट टेबल

टेबल फॉर्मॅट ऑब्जेक्ट. टेबलमध्ये केषन , रो ग्रुप आणि कॉलम ग्रुप असू शकतात. रो गटामध्ये रो असतात, तर कॉलम गटामध्ये कॉलम असतात. रो आणि कॉलम मध्ये सेल असतात. टेबल्स एका स्पेसिफिक क्रमाने लेयर्स म्हणून रेंडर केल्या जातात तळापासून वर: टेबल, कॉलम ग्रुप , कॉलम , रो ग्रुप , रो आणि सेल.

### वर्गीकरण

क्रमवारी (सॉर्टिंग) लावणे हे डेटा विश्लेषणाचे आणखी एक तंत्र आहे आणि ते तुमच्या डेटाच्या क्रमाची पुनर्रचना करण्यासाठी वापरले जाते. तुमचा डेटा एक्सप्लोर करण्याचा आणि समजून घेण्याचा हा एक सोपा पाथ आहे.

उदाहरणार्थ, समजा तुमच्याकडे 5 भिन्न नंबर ची लिस्ट होती:

1, 10, 2, 3, 4

जर आम्हाला ही यादी चढत्या क्रमाने (सर्वात कमी ते सर्वोच्च) क्रमाने लावायची असेल, तर आम्ही या लिस्ट च्या कॉलम शीर्षावर क्लिक करू आणि नंतर "चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावा" ("Sort Ascending") निवडा. हे याप्रमाणे लिस्ट व्यवस्था करेल:

1,2,3,4,10

नंबर ची चढत्या क्रमाने क्लास वारी करून आपण पाहू शकतो की त्यांचा आकार वाढत आहे. जर आम्हाला क्रमवारी (सॉर्टिंग)चा क्रम उतरत्या क्रमामध्ये बदलायचा असेल (सर्वोच्च ते सर्वात कमी) आम्ही कॉलम A वर क्लिक करू आणि याप्रमाणे "सॉर्ट डिसेंडिंग" ("Sort Descending") निवडा:

4,3,2,10,1

पुन्हा सर्वात जास्त संख्येपासून सर्वात कमी संख्येपर्यंत उतरत्या क्रमाने क्रमवारी (सॉर्टिंग) लावल्यास आपण पाहू शकतो की त्यांचा आकार कमी होत आहे.

### फिल्टरिंग

फिल्टरिंग ही स्पेसिफिक निकषानुसार डेटाची क्रमवारी (सॉर्टिंग) लावण्याची प्रक्रिया आहे. मोठ्या डेटासेटवरून डेटाचे सबसंच ओळखण्याचा हा एक प्रभावी पाथ आहे.

खालील उदाहरणामध्ये, आमच्याकडे 2012-2015 मधील मासिक विक्री डेटासह डेटासेट आहे. जर तुम्हाला फक्त वर्षाची एकूण विक्री पहायची असेल किंवा तुम्हाला किती महिन्यांत सकारात्मक वाढ झाली हे जाणून घ्यायचे असेल तर फिल्टरिंग सबयुक्त आहे.

प्रथम, तुमचा डेटा फिल्टर करण्यासाठी वापरला जाणारा फिल्टर निकष कोणत्या कॉलम त आहे ते ओळखा. या प्रकरणात, मी "विक्री वाढ" नावाचा एक नवीन कॉलम तयार केला आहे. पुढे, तुमचा "विक्री वाढ" कॉलम हायलाइट करा आणि टूलबारवरील डेटा मेनूमधून "फिल्टर" निवडा. हे एक डायलॉग बॉक्स उघडेल जिथे तुम्ही तुमचे फिल्टरिंग निकष इनपुट करू शकता. या उदाहरणात, मी सकारात्मक महसूल वाढ > 0% सह वर्षासाठी एकूण विक्रीची गणना करण्यासाठी माझा फिल्टर निकष म्हणून ">0%" वापरत आहे.

### टेबल ला टेक्स्ट मध्ये रूपांतरित करा

- 1 तुम्ही टेक्स्ट मध्ये रूपांतरित करू इच्छित असलेल्या रो किंवा टेबल निवडा.
- 2 लेआउट टॅबवर, डेटा विभागात, टेक्स्ट मध्ये रूपांतरित करा क्लिक करा.
- 3 टेक्स्ट मध्ये रूपांतरित करा बॉक्समध्ये, यासह विभक्त टेक्स्ट इंटरनल , कॉलम सीमांच्या जागी तुम्ही वापरू इच्छित विभाजक कॅरक्टर क्लिक करा. ...
- 4 ओके क्लिक करा.

## MS-Excel 2010 मधील फंक्शन आणि सूत्रे (Functions and formulas in MS-Excel 2010)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- मायक्रोसॉफ्ट एक्सेलची वैशिष्ट्ये आणि फंक्शन
- सूत्रे आणि फंक्शन
- Excel 2010 मध्ये फिरा
- कंडीशनल फॉर्मॅटिंग
- Excel स्प्रेडशीट डेटा लिंक करा

### एमएस एक्सेल मधील सूत्रे

फॉर्म्युला, वर्कशीट हे डेटाचे साथे टेबल चे प्रतिनिधित्व असेल. फॉर्म्युला मध्ये स्पेशल कोड असतो, जो सेलमध्ये प्रविष्ट केला जातो. हे काही आकडेमोड करते आणि रिझल्ट देते, जे सेलमध्ये प्रदर्शित केले जाते.

व्हॅल्यू आणि टेक्स्ट सह कार्य करण्यासाठी सूत्रे विविध ऑपरेटर आणि वर्कशीट फंक्शन्स वापरतात. सूत्रांमध्ये वापरलेली व्हॅल्यू आणि टेक्स्ट इतर सेलमध्ये असू शकतो, ज्यामुळे डेटा बदलणे सोपे होते आणि वर्कशीटला त्यांचे डायनॅमिक फॉर्मॅट मिळते. उदाहरणार्थ, ते वर्कशीटमधील डेटा क्लिकली बदलू शकते आणि सूत्रे कार्य करतात.

### सूत्रांचे एलिमेंट

फॉर्म्युला मध्ये यापैकी कोणतेही एलिमेंट असू शकतात?

- गणितीय ऑपरेटर, जसे की + (जोडण्यासाठी) आणि \* (गुणाकारासाठी)

### उदाहरण -

=A1+A2 सेल A1 आणि A2 मधील व्हॅल्यू जोडते.

- व्हॅल्यू किंवा टेक्स्ट

### उदाहरण -

=200\*0.5 गुणाकार 200 वेळा 0.5. हे फॉर्म्युला फक्त व्हॅल्यू वापरते आणि ते नेहमी 100 सारखेच रिझल्ट देते.

- सेल रेफरन्स (नामांकित सेल आणि रेन्ज सह)

### उदाहरण -

=A1=C12 सेल A1 ची सेल C12 शी कम्प्यारीजन करते. जर सेल एकसमान असतील, तर फॉर्म्युला TRUE परत करेल; अन्यथा, ते FALSE परत करते.

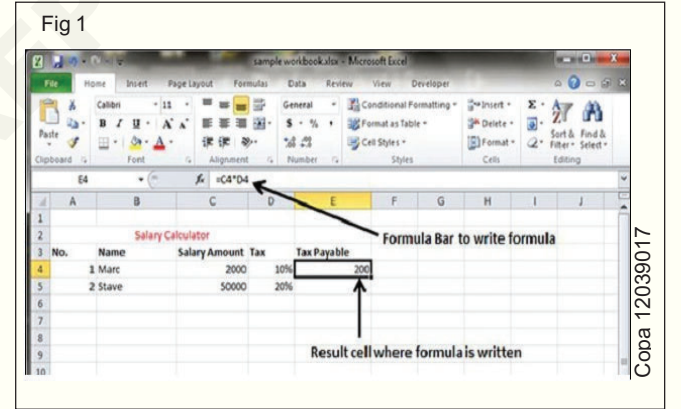
- वर्कशीट फंक्शन्स (जसे की SUM or AVERAGE)

### उदाहरण -

=SUM(A1:A12) रेन्ज A1:A12 मधील व्हॅल्यू जोडते.

### फॉर्म्युला तयार करणे

फॉर्म्युला तयार करण्यासाठी फॉर्म्युला बारमध्ये टाइप करणे आवश्यक आहे. फॉर्म्युला '=' चिन्हाने सुरू होते. सूत्रे स्वहस्ते तयार करताना, आणि एकतर सेल एड्रेस टाइप करू शकतात किंवा वर्कशीटमध्ये सूचित करू शकतात. सूत्रांसाठी सेल एड्रेस पुरवण्यासाठी पॉइंटिंग मेथड्स वापरणे ही फॉर्म्युला तयार करण्याची अधिक सोपी आणि अधिक शक्तिशाली मेथड्स आहे. बिल्ट-इन फंक्शन्स वापरताना, सेलवर क्लिक करण्यासाठी किंवा सेल रेंजमधून ड्रॅग करा जे आकृती 1 मध्ये दाखवल्याप्रमाणे फंक्शन आर्ग्युमेंट्स डायलॉग बॉक्समध्ये फंक्शनचे आर्ग्युमेंट्स परिभाषित करताना वापरू इच्छित आहेत.



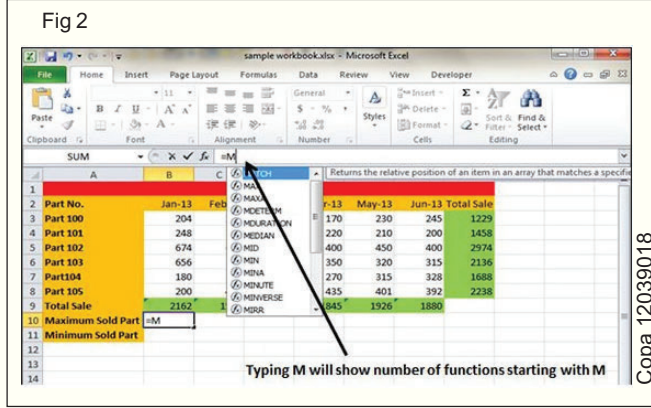
फॉर्म्युला एंट्री पूर्ण करताच, Excel निकालाची गणना करते, जो नंतर वर्कशीटमधील सेलमध्ये प्रदर्शित केला जातो (तथापि, सेल ऍक्टिव्ह असताना फॉर्म्युला बारवर फॉर्म्युलाची कॅटेंट व्हा मान राहते). जर तुम्ही फॉर्म्युला मध्ये त्रुटी केली असेल जी Excel ला फॉर्म्युला ची गणना करण्यास सक्षम होण्यापासून प्रतिबंधित करते, तर एक्सेल एक अलर्ट डायलॉग बॉक्स प्रदर्शित करते जे समस्येचे निराकरण कसे करावे हे सूचित करते.

### फॉर्म्युला तील फंक्शन

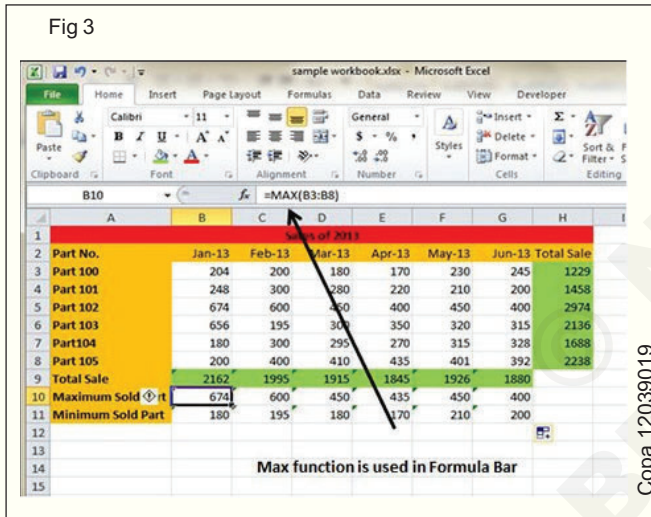
सबलब्ध वर्कशीट फंक्शन्स वापरून अनेक फॉर्म्युला तयार केली जातात. ही फंक्शन्स फॉर्म्युला ची शक्ती मोठ्या प्रमाणात वाढवण्यास सक्षम करतात आणि केवळ ऑपरेटर वापरल्यास कठीण गणना करू शकतात. उदाहरणार्थ, आणि लॉगरिथम किंवा सिन गुणोत्तर मोजण्यासाठी LOG किंवा SIN फंक्शन वापरू शकतो. आणि ही क्लिक गणना केवळ गणितीय ऑपरेटर वापरून करू शकत नाही.

## फंक्शन्स वापरणे

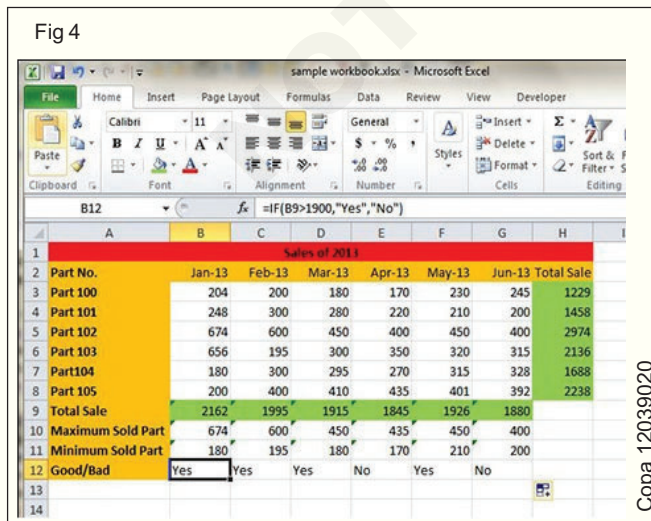
टाइप = चिन्ह आणि नंतर कोणतेही कॅरक्टर माला टाइप केल्यावर सर्व लेली फंक्शन पिक्चर 2 दर्शवतील.v



समजा रेन्ज तील सर्वात मोठे व्हॅल्यु निर्धारित करणे आवश्यक आहे. फंक्शन वापरल्याशिवाय फॉर्मूला उत्तर सांगू शकत नाही. B3:B8 या रेन्ज तील सर्वात मोठे व्हॅल्यु परत करण्यासाठी MAX फंक्शन वापरणारे फॉर्मूला आम्ही वापरू=MAX(A1:D100)आकृती 3 मध्ये दर्शविल्याप्रमाणे.



फंक्शन्सचे आणखी एक उदाहरण. समजा महिन्याचा सेल 1900 पेक्षा जास्त असल्यास आपण विक्री प्रतिनिधीला बोनस देऊ शकतो. आकृती 4 मध्ये दाखवल्याप्रमाणे =IF(B9>1900,"होय","नाही") IF फंक्शन्सह फॉर्मूला लिहून आपण ते साध्य करू शकतो.



## फंक्शन्स आर्ग्यूमेंट्स

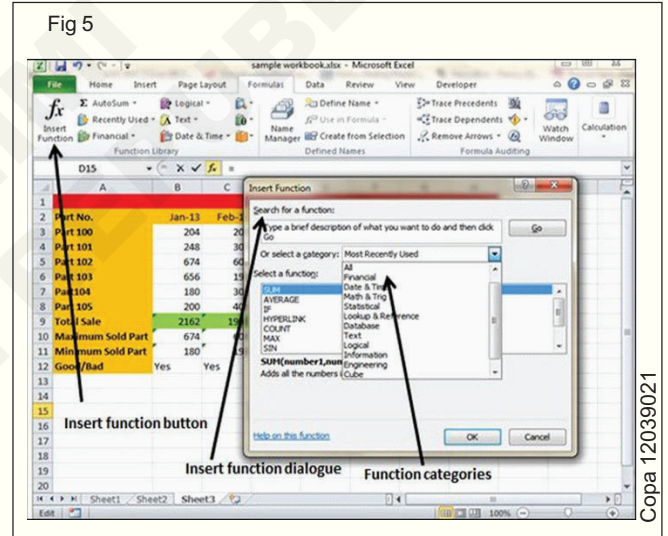
वरील उदाहरणांमध्ये, लक्षात घ्या की सर्व फंक्शन्स कंस वापरतात. कंसातील इनफार्मेशन ही युक्तिवादांची यादी आहे.

ते आर्ग्यूमेंट्स कसे वापरतात यानुसार फंक्शन बदलतात. ते काय करायचे आहे यावर अवलंबून, फंक्शन वापरू शकते.

- कोणतेही आर्ग्यूमेंट्स नाहीत (No arguments) -उदाहरणे? Now(), Date(), इ.
- एक आर्ग्यूमेंट्स -UPPER(), LOWER(), इ.
- आर्ग्यूमेंट्स ची निश्चित नंबर -IF(), MAX(), MIN(), AVERAGE(), इ.
- अनंत संख्येतील आर्ग्यूमेंट्स
- पर्यायी आर्ग्यूमेंट्स

## अंगभूत फंक्शन

एमएस एक्सेलमध्ये अनेक अंगभूत फंक्शन्स आहेत, जी आपण आपल्या फॉर्मूला त वापरू शकतो. रेन्ज नुसार सर्व फंक्शन्स पाहण्यासाठी, फॉर्म्युला टॅब निवडा " आकृती 5 मध्ये दर्शविल्याप्रमाणे फंक्शन इन्सर्ट . नंतर फंक्शन इन्सर्ट डायलॉग दिसेल ज्यामधून आपण फंक्शन निवडू शकतो.



## रेन्ज नुसार फंक्शन

MS Excel मधील काही अंगभूत फंक्शन्स पाहू.

## टेक्स्ट फंक्शन

**लोअर:**पुर्वलेल्या टेक्स्ट स्ट्रिंगमधील सर्व कॅरक्टर लोअरकेसमध्ये रूपांतरित करते

**अप्पर:**पुर्वलेल्या टेक्स्ट स्ट्रिंगमधील सर्व कॅरक्टर ना अप्पर केसमध्ये रूपांतरित करते

**ट्रिम:** टेक्स्ट स्ट्रिंगच्या सुरुवातीला आणि शेवटी ड्युप्लिकेट स्पेस आणि स्पेस काढून टाकते

**कॉन्केटनेट:**दोन किंवा अधिक टेक्स्ट स्ट्रिंग एकत्र जोडतात.

**डावीकडे:**पुर्वलेल्या टेक्स्ट स्ट्रिंगच्या प्रारंभापासून कॅरक्टर ची स्पेसिफाइड नंबर मिळवते.



**मध्यभागी:**पुरवलेल्या टेक्स्ट स्ट्रिंगच्या मध्यभागी कॅरक्टर ची स्पेसिफाइड नंबर मिळवते

**उजवीकडे:**पुरवलेल्या टेक्स्ट स्ट्रिंगच्या शेवटी कॅरक्टर ची स्पेसिफाइड नंबर मिळवते.

**LEN :**पुरवलेल्या टेक्स्ट स्ट्रिंगची लांबी मिळवते

**Find :**पुरवलेल्या टेक्स्ट स्ट्रिंगमधून (केस-सेन्सिटिव्ह) पुरवलेल्या कॅरक्टर किंवा टेक्स्ट स्ट्रिंगची स्थिती मिळवते.

### डेट आणि टाइम

**Date :**यूजर ने पुरवठा केलेल्या वर्ष, महिना आणि दिवसाची date परत करते.

**टाइम:**यूजर ने पुरवलेल्या तास, मिनिट आणि सेकंदावरून, वेळ मिळवते.

**DATEVALUE:**एक्सेलच्या date-वेळ कोडमधील तारखेचे प्रतिनिधित्व करणाऱ्या पूर्णांकामध्ये, date दर्शविणाऱ्या टेक्स्ट स्ट्रिंगमध्ये रूपांतरित करते.

**TIMEVALUE:**वेळ दर्शविणाऱ्या टेक्स्ट स्ट्रिंगला, Excel मध्ये वेळ दर्शविणाऱ्या दशांशामध्ये रूपांतरित करते.

**Now :**करंट date आणि वेळ मिळवते.

**Today:**आजची date परत करते.

### सांख्यिकी (Statistical)

**MAX:** पुरवलेल्या नंबर च्या लिस्ट मधून सर्वात मोठे व्हॅल्यु मिळवते.

**MIN:**पुरवलेल्या नंबर च्या लिस्ट मधून सर्वात लहान व्हॅल्यु मिळवते.

**AVERAGE :**पुरवलेल्या नंबर च्या लिस्ट ची सरासरी मिळवते.

**COUNT:**सेल किंवा व्हॅल्यु च्या पुरवलेल्या संचामधील नंबर त्मक व्हॅल्यु ची नंबर मिळवते.

**COUNTIF:**दिलेल्या निकषांची पूर्तता करणाऱ्या सेलची नंबर (पुरवलेल्या रेन्ज ची) मिळवते.

**SUM:**नंबर च्या पुरवलेल्या लिस्ट ची बेरीज मिळवते

लॉजिकल

**AND :**अनेक युजर -परिभाषित अटींची टेस्टिंग करते आणि सर्व अटींचे व्हॅल्यु मापन सत्य किंवा असत्य असे झाल्यास सत्य मिळवते

**OR :**अनेक युजर -परिभाषित अटींची टेस्टिंग घेते आणि कोणत्याही परिस्थितीचे व्हॅल्यु मापन सत्य किंवा असत्य असे झाल्यास TRUE मिळवते.

**NOT :**एक लॉजिकल व्हॅल्यु मिळवते जे यूजर ने पुरवलेल्या लॉजिकल व्हॅल्यु च्या किंवा अभिव्यक्तीच्या विरुद्ध असते म्हणजे पुरवठा केलेला आर्ग्युमेंट्स TRUE असल्यास FALSE मिळवते आणि पुरवठा केलेला आर्ग्युमेंट्स FALSE असल्यास TRUE मिळवते

### गणित आणि Trig

**ABS:**पुरवलेल्या संख्येचे परिपूर्ण व्हॅल्यु (म्हणजे मॉड्यूलस) मिळवते

**.साइन:**पुरवलेल्या क्रमांकाचे चिन्ह (+1, -1 किंवा 0) मिळवते.

**SQRT:**दिलेल्या संख्येचे धनात्मक क्लास मूळ मिळवते.

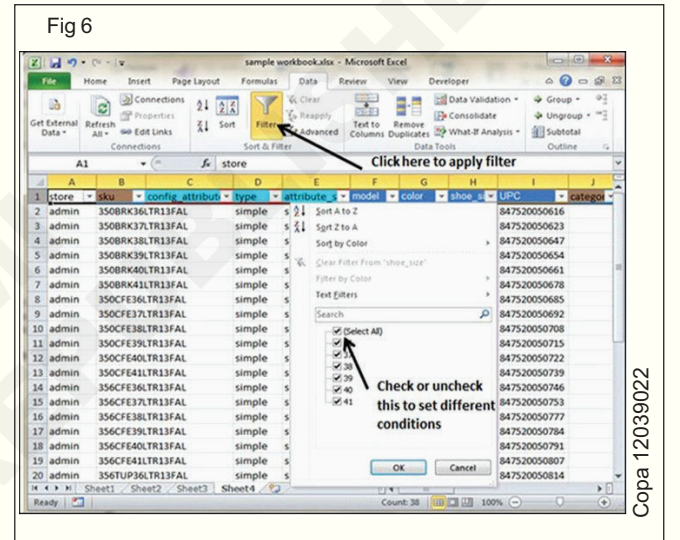
**MOD :**दोन पुरविलेल्या नंबर मध्ये भागाकारातून उर्वरित मिळवते.

### एमएस एक्सेल मध्ये एमएस एक्सेलमधील फिल्टर

एमएस एक्सेलमधील डेटा फिल्टर करणे म्हणजे स्पेसिफिक अटी पूर्ण करणाऱ्या रो प्रदर्शित करणे होय. (इतर रो लपवल्या जातात.)

स्टोअर डेटा वापरून, जर यूजर ला शूचा आकार 36 आहे तेथे डेटा पाहण्यात स्वारस्य असल्यास, हे करण्यासाठी फिल्टर सेट करा. हे करण्यासाठी खाली नमूद केलेल्या स्टेप्स चे अनुसरण करा.

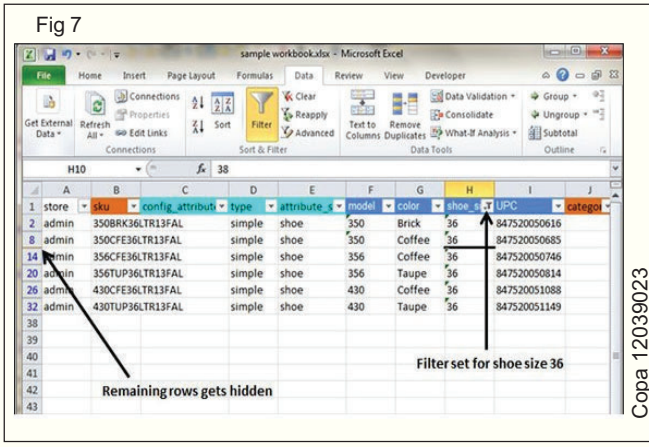
- हेडर रो वर कर्सर ठेवा.
- निवडा डेटा टॅब > फिल्टर पिकवर 6 मध्ये दाखवल्याप्रमाणे फिल्टर सेट करण्यासाठी.



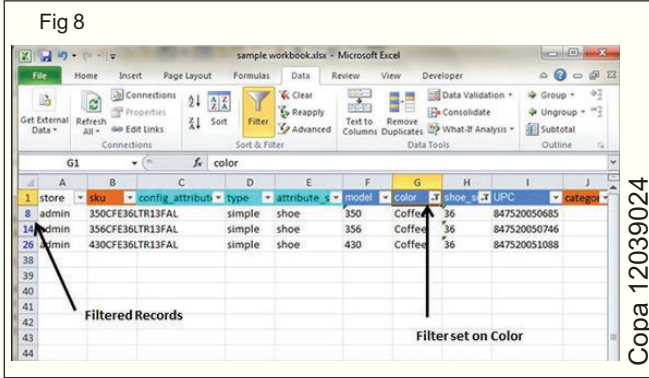
- एरिया रो हेडरमधील ड्रॉप-डाउन बाणावर क्लिक करा आणि सर्व निवडा मधून चेक मार्क काढून टाका, जे प्रत्येक गोष्टीची निवड रद्द करते.
- नंतर आकार 36 साठी चेक मार्क निवडा जे डेटा फिल्टर करेल आणि आकृती -23 मध्ये दर्शविल्याप्रमाणे शू साइज 36 चा डेटा प्रदर्शित करेल.
- काही रो क्रमांक गहाळ आहेत; या रो मध्ये फिल्टर केलेला (लपलेला) डेटा असतो.
- एरिया कॉलममध्ये ड्रॉप-डाउन अॅरो आहे आता वेगळे ग्राफिक दाखवते -एक आयकॉन जो कॉलम फिल्टर झाला आहे हे सूचित करतो. (पिकचर 7)

### मल्टिपल फिल्टर वापरणे

अनेक अटीनुसार रेकॉर्डचे फिल्टरिंग म्हणजे मल्टिपल कॉलम व्हॅल्यु द्वारे. समजा आकार 36 फिल्टर केल्यानंतर, फिल्टर असणे आवश्यक आहे जेथे रंग कॉफी सारखा असेल. शू साइजसाठी फिल्टर सेट केल्यानंतर, कलर कॉलम निवडा आणि नंतर रंगासाठी फिल्टर सेट करा. (पिकचर 8)



Copa 12039023



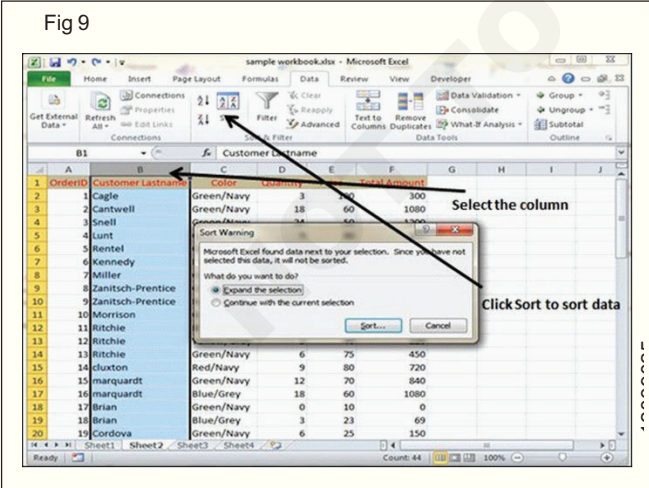
Copa 12039024

### एमएस एक्सेल मध्ये क्रमवारी (सॉर्टिंग) (सॉर्टिंग) लावणे

एमएस एक्सेलमधील डेटाची क्रमवारी (सॉर्टिंग) स्पेसिफिक कॉलम तील कॅटॅगरी वर आधारित रो ची पुनर्रचना करते. कॅरक्टर क्रमानुसार नावे ठेवण्यासाठी टेबल क्रमवारी (सॉर्टिंग) लावा किंवा लहान ते सर्वात मोठ्या किंवा सर्वात मोठ्या ते सर्वात लहान रकमेनुसार डेटा क्रमवारी (सॉर्टिंग) लावा.

डेटा क्रमवारी (सॉर्टिंग) लावण्यासाठी खाली नमूद केलेल्या स्टेप्स चे अनुसरण करा.

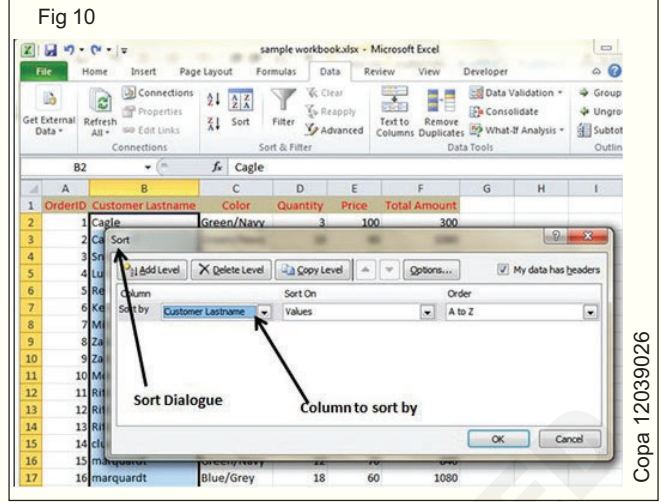
- आकृती 9 प्रमाणे डेटा क्रमवारी (सॉर्टिंग) लावण्यासाठी कॉलम निवडा.



Copa 12039027

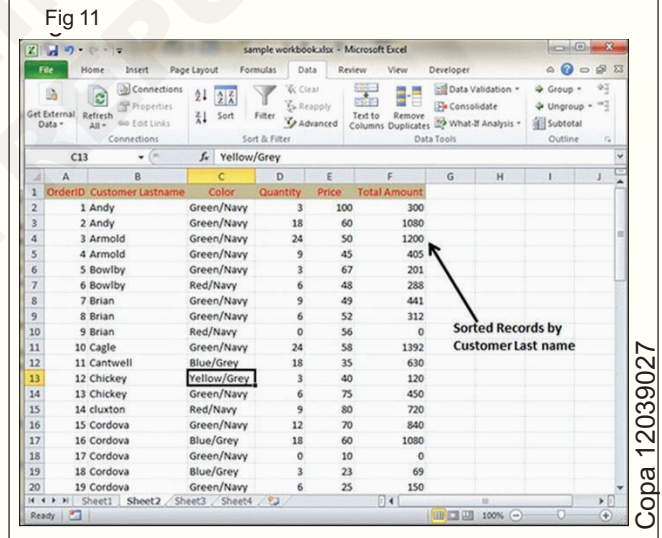
- डेटा टॅब निवडा " क्रमवारी (सॉर्टिंग) खाली डायलॉग दिसेल.
- यूजर ने निवडलेल्या कॉलम वर आधारित डेटा क्रमवारी (सॉर्टिंग) लावण्यास, निवडीसह सुरु ठेवा निवडा किंवा इतर स्तंभांवर आधारित डेटा क्रमवारी (सॉर्टिंग) असल्यास, निवड विस्तृत करा निवडा.

- पिक्चर 10 प्रमाणे खालील अटींवर आधारित डेटाची क्रमवारी (सॉर्टिंग) लावा.



Copa 12039026

- व्हॅल्यू -कॅरक्टर क्रमानुसार किंवा अंकानुसार.
- सेल रंग -सेलच्या रंगावर आधारित.
- फॉन्टचा रंग -फॉन्ट रंगावर आधारित.
- सेल चिन्ह -सेल आयकॉनवर आधारित.
- ओके वर क्लिक केल्याने पिक्चर 11 प्रमाणे डेटा क्रमवारी (सॉर्टिंग) लावला जाईल.



Copa 12039027

होम टॅबमधून सॉर्टिंग पर्याय देखील सबलब्ध आहे. होम टॅब निवडा " क्रमवारी (सॉर्टिंग) लावा आणि फिल्टर करा. तुम्ही पिक्चर 12 प्रमाणे रेकॉर्ड क्रमवारी (सॉर्टिंग) लावण्यासाठी समान संवाद पाहू शकता.

### MS Excel मधील रेन्ज

सेल हा वर्कशीटमधील एक एलिमेंट आहे ज्यामध्ये व्हॅल्यू , काही टेक्स्ट किंवा फॉर्मूला असू शकते. सेल त्याच्या पत्त्याद्वारे ओळखला जातो, ज्यामध्ये त्याचे कॉलम अक्षर आणि रो क्रमांक असतात. उदाहरणार्थ, सेल B1 हा दुसऱ्या कॉलम तील आणि पहिल्या रो मधील सेल आहे.

सेल च्या समूहाला रेन्ज म्हणतात. तुम्ही रेन्ज चा एड्रेस स्पेसिफाइड करून त्याचा वरचा-डावा सेल एड्रेस आणि त्याचा खालचा-उजवा सेल एड्रेस , कोलनने विभक्त केला आहे.

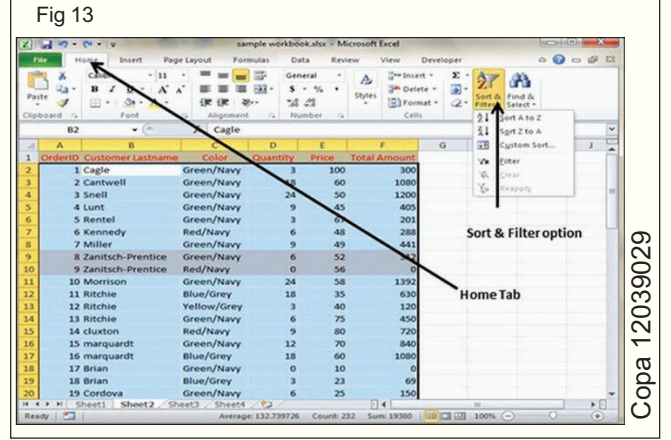
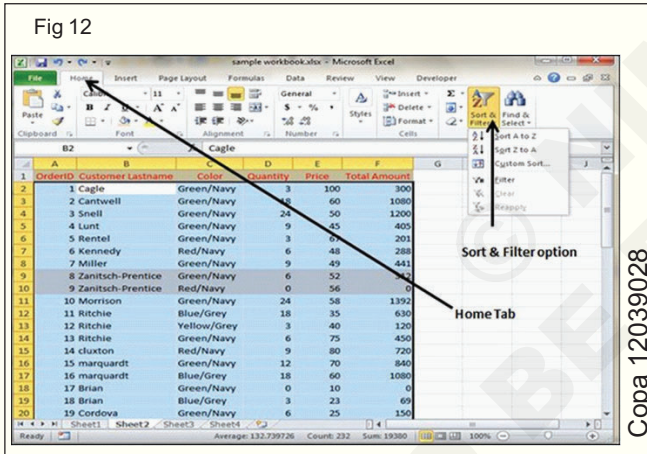
## रेन्ज चे उदाहरण:

- **C24** -सिंगल सेलचा समावेश असलेली रेन्ज .
- **A1:B1** -दोन सेल जे एक रो आणि दोन कॉलम व्यापतात.
- **A1:A100** -कॉलम A मध्ये 100 सेल .
- **A1:D4** -16 सेल (चार स्तंभांनी चार रो ) .

## रेन्ज निवडणे

अनेक प्रकारे रेन्ज निवडणे?

- माऊसचे डावे बटण दाबा आणि रेन्ज हायलाइट करून ड्रॅग करा. नंतर माऊस बटण सोडा. स्क्रीनच्या शेवटी ड्रॅग केल्यास, वर्कशीट स्क्रोल होईल.
- रेन्ज निवडण्यासाठी नेव्हिगेशन की वापरत असताना Shift की दाबा.
- F8 दाबा आणि नंतर रेन्ज हायलाइट करण्यासाठी नेव्हिगेशन कीसह सेल पॉइंटर हलवा. नेव्हिगेशन की सामान्य हालचालीवर परत येण्यासाठी पुन्हा F8 दाबा.
- नाव बॉक्समध्ये सेल किंवा रेन्ज एड्रेस टाईप करा आणि एंटर दाबा. एक्सेल स्पेसिफाइड केलेला सेल किंवा रेन्ज निवडतो. (पिक्चर 13)

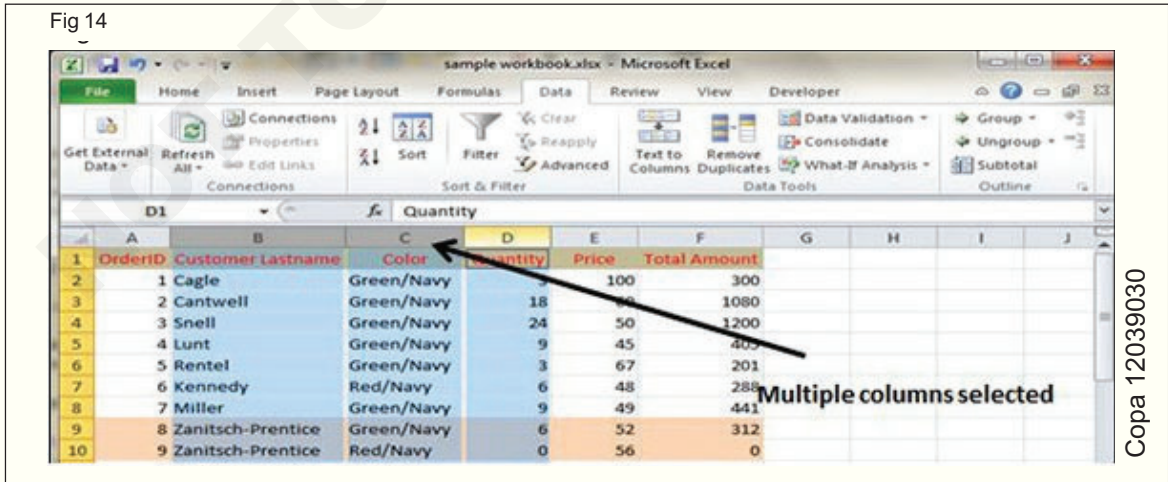


## पूर्ण रो आणि कॉलम निवडणे

जेव्हा यूजर ला संपूर्ण रो किंवा कॉलम निवडण्याची आवश्यकता असते. आणि निवडलेल्या रेन्ज प्रमाणेच संपूर्ण रो आणि कॉलम निवडू शकतात:

- सिंगल रो किंवा कॉलम निवडण्यासाठी रो किंवा कॉलम बॉर्डर क्लिक करा.
- मल्टिपल समीप रो किंवा कॉलम निवडण्यासाठी, रो किंवा कॉलम बॉर्डर वर क्लिक करा आणि अतिरिक्त रो किंवा कॉलम हायलाइट करण्यासाठी ड्रॅग करा.
- मल्टिपल (नजीक नसलेल्या) रो किंवा कॉलम निवडण्यासाठी, रो किंवा कॉलम बॉर्डर वर क्लिक करताना Ctrl दाबा. (पिक्चर 14)

Fig 14



## चार्ट मॅनेज करा (Manage Charts)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- चार्ट तयार करा
- चार्ट सुधारा
- फॉर्मॅट चार्ट.

### चार्ट

डेटा स्पष्ट करण्यासाठी आणि कम्प्यारीजन करण्यासाठी चार्ट इन्सर्ट करतात.

बार, पाई, रेखा, एरिया आणि सरफेस एरिया हे काही सबलब्ध टाइप आहेत.

तुमच्या डेटासाठी शिफारस केलेला चार्ट (ग्राफ) तयार करा, जवळजवळ आता सबलब्ध नसलेला चार्ट विझार्ड वापरण्या इतकाच जलद.

### एक चार्ट तयार करा

- 1 डेटा निवडा ज्यासाठी तुम्हाला चार्ट तयार करायचा आहे.
- 2 INSERT > शिफारस केलेले चार्ट वर क्लिक करा.
- 3 शिफारस केलेले चार्ट टॅबवर, Excel ने तुमच्या डेटासाठी शिफारस केलेल्या चार्टच्या लिस्ट मधून स्क्रोल करा आणि तुमचा डेटा कसा दिसेल हे पाहण्यासाठी कोणत्याही चार्टवर क्लिक करा.  
तुम्हाला आवडलेला चार्ट दिसत नसल्यास, पाहण्यासाठी सर्व चार्ट क्लिक करा सर्व सबलब्ध चार्ट टाइप .
- 4 तुम्हाला तुम्हाला आवडलेला चार्ट सापडल्यावर, त्यावर > ओके क्लिक करा.
- 5 चार्ट एलिमेंट , चार्ट स्टाइल आणि चार्ट फिल्टर बटणे वापरा, चार्टच्या वरच्या उजव्या कोपऱ्यात, एक्सिस टायटल किंवा डेटा लेबले सारखे चार्ट एलिमेंट जोडण्यासाठी, तुमच्या चार्टचे फॉर्मॅट कस्टमाइज करा किंवा मध्ये दर्शविलेला डेटा बदला. चार्ट
- 6 अतिरिक्त डिझाइन आणि फॉर्मॅटिंग वैशिष्ट्यांमध्ये एक्सेस करण्यासाठी, चार्ट टूल्स रिबनमध्ये जोडण्यासाठी चार्टमध्ये कुठेही क्लिक करा आणि नंतर डिझाइन आणि फॉर्मॅट टॅबवर तुम्हाला हवे असलेल्या पर्यायांवर क्लिक करा.

### एक्सेल चार्टमध्ये डेटा सीरीज म्हणजे काय?

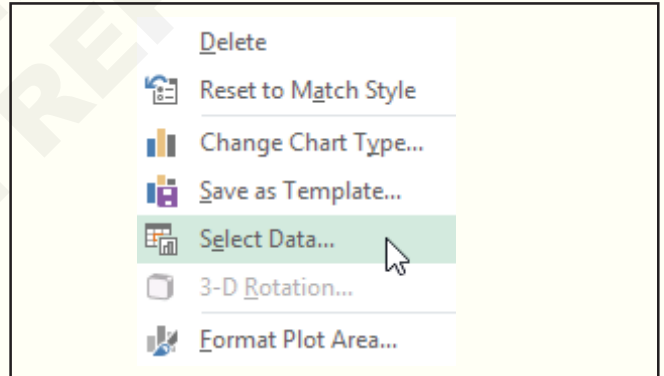
एक्सेलमध्ये डेटा सीरीज म्हणजे काय? डेटा सीरीज म्हणजे स्प्रेडशीटवरील कॉलम तील सलग व्हॅल्यू किंवा नंबर चा संग्रह. आम्ही डेटाच्या स्टार्टिंग

मालिकेतून आलेख तयार करून डेटा सीरीज तयार करू शकतो, त्यानंतर, आम्ही डेटाची दुसरी सीरीज तयार करू शकतो आणि पूर्व-अस्तित्वात असलेल्या डेटा मालिकेसह हायलाइट करू शकतो.

### वेगळ्या चार्ट शीटवर चार्टमध्ये डेटा सीरीज ऑड करा

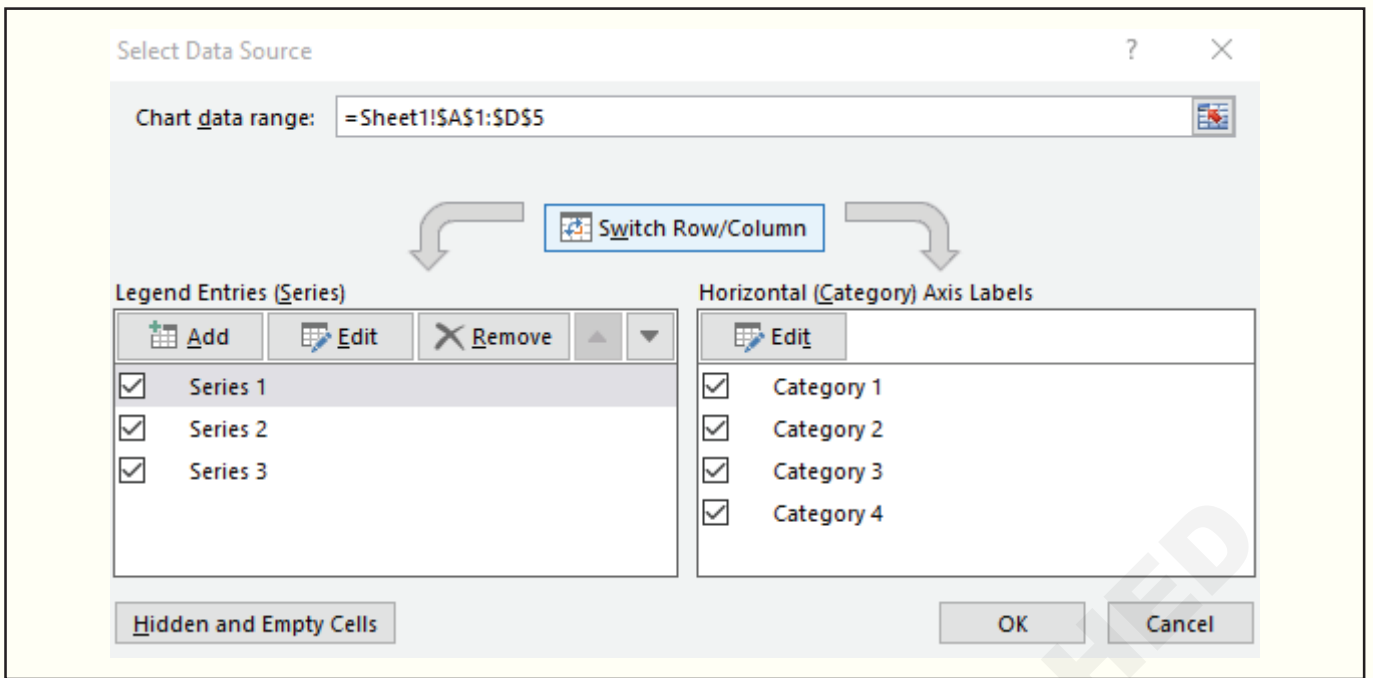
तुमचा चार्ट वेगळ्या वर्कशीटवर असल्यास, नवीन डेटा सीरीज जोडण्यासाठी ड्रॅग करणे हा सर्वोत्तम पाथ असू शकत नाही. त्या बाबतीत, तुम्ही डेटा सोर्स निवडा डायलॉग बॉक्समध्ये चार्टसाठी नवीन डेटा प्रविष्ट करू शकता.

- 1 तुमचा चार्ट डेटा असलेल्या वर्कशीटवर, चार्टसाठी तुमच्या विद्यमान सोर्स डेटाच्या थेट पुढे किंवा खाली सेलमध्ये, तुम्हाला ऑड करायची असलेली नवीन डेटा सीरीज प्रविष्ट करा.
- 2 तुमचा चार्ट असलेल्या वर्कशीटवर क्लिक करा.
- 3 चार्टवर उजवे-क्लिक करा, आणि नंतर डेटा निवडा निवडा.



वर्कशीटवर डेटा सोर्स निवडा डायलॉग बॉक्स दिसेल ज्यामध्ये चार्टसाठी सोर्स डेटा आहे.

- 4 डायलॉग बॉक्स उघडा ठेवून, वर्कशीटमध्ये क्लिक करा, आणि नंतर नवीन डेटा मालिकेसह, चार्टसाठी तुम्ही वापरू इच्छित असलेला सर्व डेटा निवडण्यासाठी क्लिक करा आणि ड्रॅग करा.  
डेटा सोर्स सिलेक्ट डायलॉग बॉक्समध्ये लीजेंड एंट्रीज (सीरीज ) इंटर्नल नवीन डेटा सीरीज दिसते.
- 5 डायलॉग बॉक्स बंद करण्यासाठी आणि चार्ट शीटवर परत येण्यासाठी ओके क्लिक करा.



## पिव्होट टेबल तयार करा (Create Pivot Tables)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- पिव्होट टेबल
- पिव्होट टेबल कार्य
- ग्रुप डेटा.

### पिव्होट टेबल

PivotTable हा मोठ्या प्रमाणात डेटा क्लिकली सारांशित करण्याचा एक संवादी पाथ आहे. अंकीय डेटाचे तपशीलवार विश्लेषण करण्यासाठी आणि तुमच्या डेटाबद्दल अनपेक्षित प्रश्नांची उत्तरे देण्यासाठी तुम्ही PivotTable वापरू शकता. PivotTable स्पेशल तः यासाठी डिझाइन केले आहे: अनेक युजर -अनुकूल मार्गांनी मोठ्या प्रमाणात डेटाची क्वेरी करणे.

### पिव्होट टेबल काम

पिव्होट टेबल मध्ये संचयित केलेला सारांश, क्रमवारी (सॉर्टिंग), पुनर्रचना, ग्रुप , गणना, एकूण किंवा सरासरी डेटा करण्यासाठी वापरली जाते. हे आपल्याला कॉलम चे रो मध्ये आणि रो चे कॉलम मध्ये रूपांतर करण्यास अनुमती देते. हे कोणत्याही फील्ड (कॉलम ) द्वारे ग्रुप बद्ध करण्यास आणि त्यावर एडवान्स गणना वापरण्यास अनुमती देते.

### ग्रुप डेटा

PivotTable मधील डेटाचे ग्रुप बद्ध करणे तुम्हाला विश्लेषण करण्यासाठी डेटाचा सबसंच दाखवण्यात मदत करू शकते. उदाहरणार्थ, तुम्हाला पिव्होटटेबलमधील अनाठायी यादी date आणि वेळ फील्ड क्वार्टर आणि महिन्यांमध्ये ग्रुप बद्ध करू शकता

## एमएस पॉवरपॉइंट प्रेझेंटेशनमध्ये फाइल्स उघडा (Open files in MS PowerPoint Presentations)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एमएस पॉवर पॉइंट प्रेझेंटेशनचा परिचय
- पिक्चर इफेक्ट लागू आणि एडिट करा.

मायक्रोसॉफ्ट पॉवरपॉइंट हा अधिक प्रभावी प्रेझेंटेशन तयार करण्यासाठी मायक्रोसॉफ्टने विकसित केलेला एक साधा परंतु शक्तिशाली सॉफ्टवेअर प्रोग्राम आहे. हे प्रामुख्याने मायक्रोसॉफ्ट ऑफिस सूटचा एक पार्ट आहे. प्रोग्राममध्ये स्लाइड्स आणि वर्ड प्रोसेसिंग, ड्रॉइंग, आलेख आणि एक्सटर्नल रेखा यांसारख्या विविध सबयुक्त साधनांचा समावेश आहे. अशा प्रकारे स्लाइड्समध्ये टेक्स्ट, टेबल्स, टेबल्स, ग्राफिक्स आणि मीडिया प्रदर्शित करण्यासाठी याचा वापर केला जाऊ शकतो.

मायक्रोसॉफ्ट पॉवरपॉइंट विंडोची प्रामुख्याने तीन लक्षवेधी वैशिष्ट्ये आहेत ज्यावर पॉवरपॉइंटबद्दल अधिक शिकत असताना त्यावर लक्ष केंद्रित करणे आवश्यक आहे. या वैशिष्ट्यांना मायक्रोसॉफ्ट ऑफिस बटण, क्लिक ऍक्सेस टूलबार आणि रिबन म्हणतात.

पॉवरपॉइंट स्लाइडशो (PPT) हे Microsoft च्या सॉफ्टवेअरवर तयार केलेले एक प्रेझेंटेशन आहे जे वापरकर्त्यांना सादरीकरणामध्ये ऑडिओ, व्हिज्युअल आणि ऑडिओ/व्हिज्युअल वैशिष्ट्ये जोडण्याची परवानगी देते. हे मल्टीमीडिया तंत्रज्ञान मानले जाते आणि ते कोलायब्रेशन आणि कंटेंट सामायिकरणाने साधन म्हणून देखील कार्य करते. PowerPoint हे Microsoft Office मध्ये समाविष्ट केले आहे, ज्यामुळे ते प्रेझेंटेशन सॉफ्टवेअरच्या सर्वात सुप्रसिद्ध आणि मोठ्या प्रमाणावर वापरले जाणारे ब्रँड बनले आहे.

पॉवरपॉइंट स्लाइडशोला पॉवरपॉइंट प्रेझेंटेशन असेही म्हणतात.

### चित्रे एडिट करा

- 1 ब्राइटनेस, कॉन्ट्रास्ट किंवा शार्पनेस अडजस्ट करा. पिक्चर निवडा. पिक्चर फॉर्मॅट निवडा आणि सुधारणा निवडा. ...
- 2 कलात्मक इफेक्ट लागू करा. पिक्चर निवडा. ...
- 3 रंग बदला. पिक्चर निवडा. ...
- 4 पिक्चर इफेक्ट लागू करा. पिक्चर निवडा. ...
- 5 बॉर्डर ऑड करा. पिक्चर निवडा. ...
- 6 पिक्चर कॉम्प्रेस करा. पिक्चर निवडा.
  - ब्राइटनेस, कॉन्ट्रास्ट किंवा शार्पनेस अडजस्ट करा
  - पिक्चर निवडा.
  - पिक्चर फॉर्मॅट निवडा आणि सुधारणा निवडा.
  - त्यांचे प्रीव्यू करण्यासाठी पर्यायांवर फिरवा आणि निवडा तुम्हाला पाहिजे ते.

अधिक इनफार्मेशन साठी, ब्राइटनेस, कॉन्ट्रास्ट किंवा बदला पहा चित्राची शार्पनेस .

### कलात्मक इफेक्ट लागू करा

- 1 पिक्चर निवडा.
- 2 पिक्चर फॉर्मॅट निवडा आणि कलात्मक इफेक्ट निवडा.
- 3 त्यांचे प्रीव्यू करण्यासाठी पर्यायांवर फिरवा, नंतर तुम्हाला हवा असलेला एक निवडा.

टीप: तुम्ही एका चित्रावर एका वेळी फक्त एक कलात्मक इफेक्ट लागू करू शकता, म्हणून भिन्न कलात्मक इफेक्ट लागू केल्याने पूर्वी लागू केलेला कलात्मक इफेक्ट काढून टाकला जाईल.

- 4 रंग बदला
- 5 पिक्चर निवडा.
- 6 चित्राचे फॉर्मॅट निवडा आणि रंग निवडा.
- 7 त्यांचे प्रीव्यू करण्यासाठी पर्यायांवर फिरवा, नंतर तुम्हाला हवा असलेला एक निवडा.

### पिक्चर इफेक्ट लागू करा

- 1 पिक्चर निवडा.
- 2 पिक्चर फॉर्मॅट निवडा आणि पिक्चर इफेक्ट निवडा.
- 3 तुम्हाला हवे असलेले निवडा: छाया, परावर्तन, चमक, मऊ कडा, बेव्हल किंवा 3-डी रोटेशन.

अधिक इनफार्मेशन साठी, चित्रासाठी इफेक्ट ऑड करा किंवा बदला पहा.
- 4 बॉर्डर ऑड करा
- 5 पिक्चर निवडा.
- 6 पिक्चर फॉर्मॅट > पिक्चर बॉर्डर निवडा, नंतर बॉर्डर निवडा.

अधिक इनफार्मेशन साठी, पिक्चर बॉर्डर ऑड करा आणि काढा पहा.
- 7 बॅकग्राऊंड काढा  
इनफार्मेशन साठी, चित्राची बॅकग्राऊंड काढा पहा.
- 8 पिक्चर कॉम्प्रेस करा
- 9 पिक्चर निवडा.
- 10 पिक्चर फॉर्मॅट निवडा आणि कॉम्प्रेस पिक्चर्स निवडा.
- 11 तुम्हाला हवे असलेले पर्याय निवडा, नंतर ओके निवडा.

## इन्सर्ट , टेक्स्ट आणि प्याराग्राफ फॉरमॅटिंग करा (Insert, Format text and paragraphs)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टेक्स्ट एलाइनमेंट लागू करा.

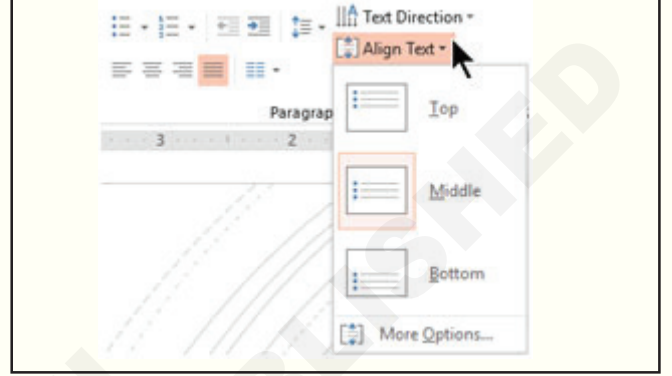
टेक्स्ट एलाइनमेंट कसे बदलावे

- आवश्यक टेक्स्ट निवडा
- आता, होम टॅबमध्ये पैराग्राफ ग्रुप शोधा
- प्याराग्राफ गटाच्या तळाशी चार एलाइनमेंट पर्याय सबलब्ध आहेत
- डाव्या क्लिकने तुमचा इच्छित एलाइनमेंट पर्याय निवडा.

चार एलाइनमेंट पर्याय सबलब्ध आहेत:

- **टेक्स्ट डावीकडे एलाइनमेंट करा** : ते तुमचा टेक्स्ट डाव्या समासात एलाइनमेंट करते
- **सेंटर**: टेक्स्ट मध्यभागी आणतो

- **टेक्स्ट उजवीकडे एलाइनमेंट करा**: तुमचा टेक्स्ट उजव्या समासात एलाइनमेंट करतो
- **जस्टिफाई**: टेक्स्ट डाव्या आणि उजव्या समासात एलाइनमेंट करते





## टेबल तयार करा, टेबल्स मॉडिफाई करा, बुलेट केलेला टेक्स्ट सुधारा (Create tables, modify tables, modify bulleted text)

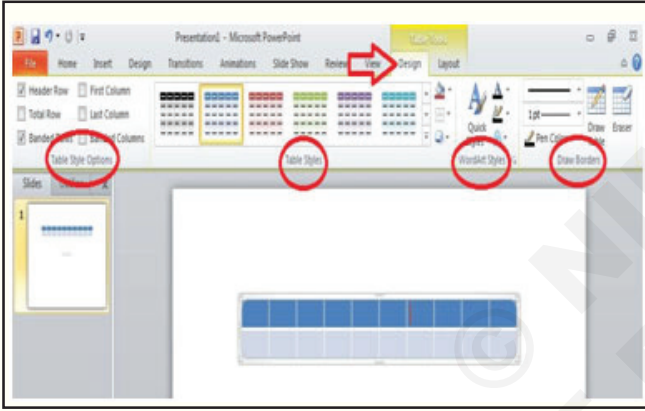
उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- टेबल मॉडिफाई किंवा फॉर्मॅटिंग करा
- बुलेट केलेली आणि क्रमांकित यादी लागू करा.

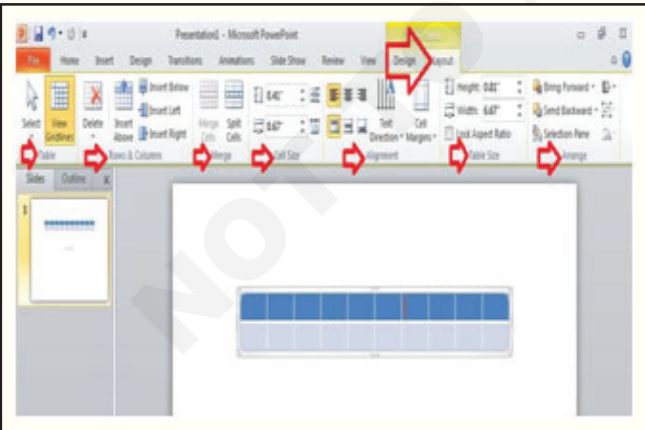
टेबल कसे बदलायचे किंवा फॉर्मॅटिंग करायचे

तुम्हाला सुधारित करण्याचे असलेले स्पेसिफिक टेबल निवडा. डिझाइन आणि लेआउट नावाचे दोन नवीन टॅब आता रिबनमध्ये दिसतील. हे टॅब तुमच्या टेबल्स फॉर्मॅट करण्यासाठी कमांडचे विविध नवीन ग्रुप देतात.

जेव्हा तुम्ही डिझाइन टॅबवर क्लिक करता तेव्हा ते तुम्हाला कमांडचे चार नवीन ग्रुप दाखवतात; टेबल स्टाइल पर्याय, टेबल स्टाइल, वर्ड आर्ट स्टाइल आणि रेखापिक्चर बॉर्डर.

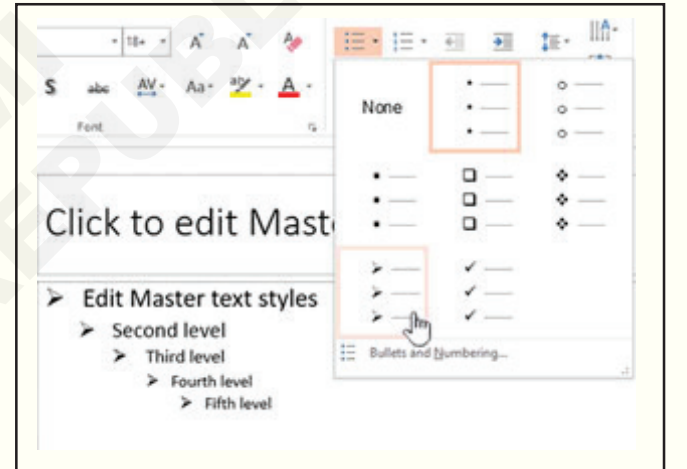


जेव्हा तुम्ही लेआउट टॅबवर क्लिक कराल तेव्हा ते कमांडचे सहा ग्रुप प्रदर्शित करेल; टेबल, रो आणि कॉलम, मर्ज, सेल आकार आणि एलाइनमेंट.



बुलेट केलेल्या किंवा नम्बरड याद्या कशा तयार करायच्या

- प्रथम, तुम्हाला बुलेट किंवा क्रमांक जोडण्याचा टेक्स्ट किंवा लिस्ट निवडा
- आता, होम टॅब निवडा आणि पैराग्राफ ग्रुप शोधा
- बुलेट किंवा नंबरिंग बटणावर क्लिक करा किंवा आणखी बुलेट स्टाइल आणि क्रमांकन फॉर्मॅट पाहण्यासाठी या बटणांपुढील ड्रॉप-डाउन बाणावर क्लिक करा
- तुमची स्टाइल निवडा जी तुम्हाला टेक्स्ट वर लागू करायची आहे



## चित्रे, फॉरमॅट चित्रे आणि टेक्स्ट बॉक्स इन्सर्ट (Insert illustrations, Format illustrations and text boxes)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- स्लाइडमध्ये टेक्स्ट ऑड करा
- लेआउट रंग किंवा स्टाइल बदला
- स्मार्ट आर्ट फॉरमॅट करा.

### स्लाइडमध्ये टेक्स्ट ऑड करा

तुम्ही टेक्स्ट बॉक्स टाकून आणि त्या बॉक्समध्ये टाइप करून PowerPoint स्लाइड किंवा स्लाइड मास्टरमध्ये टेक्स्ट जोडू शकता. त्यानंतर तुम्ही टेक्स्ट किंवा संपूर्ण बॉक्स निवडून तो टेक्स्ट फॉरमॅट करू शकता. तुम्ही प्लेसहोल्डर आणि आकारांमध्ये टेक्स्ट देखील जोडू शकता. ते उघडण्यासाठी खालील टाईटल निवडा आणि तपशीलवार सूचना पहा.

### टेक्स्ट बॉक्समध्ये टेक्स्ट ऑड करा

- कोणीही एडिट करू शकतील अशा टेक्स्ट बॉक्समध्ये टेक्स्ट जोडण्यासाठी, सामान्य दृश्यात, टेक्स्ट बॉक्समध्ये क्लिक करा आणि नंतर टेक्स्ट टाइप करा किंवा पेस्ट करा.
- कायमस्वरूपी आणि एडिटिंग न करता येणाऱ्या टेक्स्ट बॉक्समध्ये टेक्स्ट जोडण्यासाठी, स्लाइड मास्टर दृश्यात, टेक्स्ट बॉक्समध्ये क्लिक करा, आणि नंतर टेक्स्ट टाइप करा किंवा पेस्ट करा.

स्लाइडवर कुठेही टेक्स्ट ठेवण्यासाठी टेक्स्ट बॉक्स वापरा, जसे की टेक्स्ट प्लेसहोल्डरच्या बाहेर. उदाहरणार्थ, चित्राला क्लिक जोडण्यासाठी, एक टेक्स्ट बॉक्स तयार करा आणि त्यास चित्राजवळ ठेवा.

### शेप चा पार्ट असलेला टेक्स्ट ऑड करा

चौरस, वर्तुळे, कॉलआउट फुगे आणि ब्लॉक बाण यांसारख्या आकारांमध्ये टेक्स्ट असू शकतो. जेव्हा तुम्ही आकारात टेक्स्ट टाइप करता, तेव्हा टेक्स्ट आकाराला जोडला जातो आणि त्यासोबत फिरतो आणि फिरतो.

- आकाराचा पार्ट बनलेला टेक्स्ट जोडण्यासाठी, आकार निवडा आणि नंतर टेक्स्ट टाइप करा किंवा पेस्ट करा.

### शेप चा स्वतंत्र असलेला टेक्स्ट ऑड करा

जर तुम्हाला आकारात टेक्स्ट ऑड करायचा असेल तर टेक्स्ट बॉक्स सुलभ आहे, परंतु टेक्स्ट आकारात जोडू इच्छित नाही. तुम्ही टेक्स्ट बॉक्समध्ये टेक्स्ट वर बॉर्डर, भरा, सावली किंवा त्रिमितीय (3-डी) इफेक्ट जोडू शकता.

- आकारापेक्षा स्वतंत्रपणे फिरणारा टेक्स्ट जोडण्यासाठी, टेक्स्ट बॉक्स ऑड करा आणि नंतर टेक्स्ट टाइप किंवा पेस्ट करा.

### टेक्स्ट बॉक्स ऑड करा

- समाविष्ट करा टॅबवर, टेक्स्ट गटामध्ये, टेक्स्ट बॉक्सवर क्लिक करा.

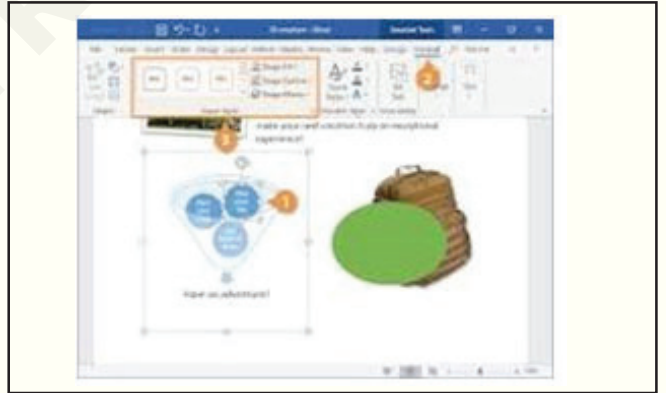
**टीप:** जर तुम्ही पूर्व आशियाई भाषा वापरत असाल तर, टेक्स्ट बॉक्स ड्रॉप-डाउन मेनूमधून, हॉरिजॉन्टल किंवा व्हर्टिकल एलाइनमेंट वर क्लिक करा.

- स्लाइडवर क्लिक करा, आणि नंतर टेक्स्ट बॉक्स काढण्यासाठी पॉइंटर ड्रॉग करा.

### लेआउट, रंग किंवा स्टाइल बदला

SmartArt निवडा. SmartArt Tools > Design टॅब निवडा. लेआउट्सचे प्रीव्यू करण्यासाठी त्यावर फिरवा आणि तुम्हाला हवा असलेला एक निवडा. रंग बदला निवडा, त्यांचे प्रीव्यू करण्यासाठी पर्यायांवर फिरवा आणि तुम्हाला हवा असलेला निवडा.

### तुम्ही स्मार्ट आर्टचे फॉरमॅट कसे करता?



### SmartArt फॉरमॅट

तुम्हाला फॉरमॅट करायचा असलेला SmartArt आकार क्लिक करा.

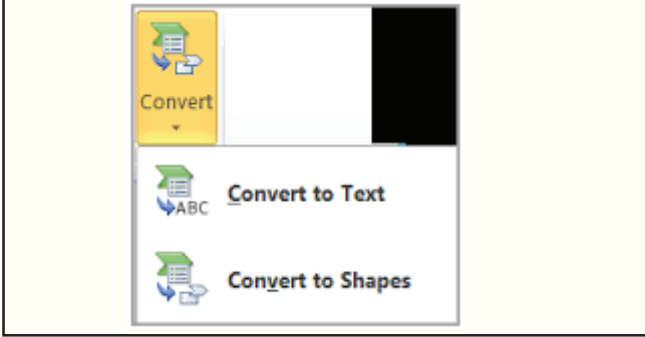
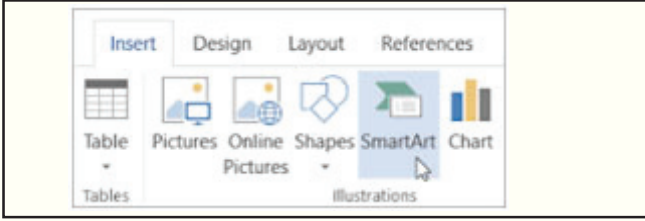
**SmartArt Tools इंटरनल Format** टॅबवर क्लिक करा. प्रत्येक आकार फॉरमॅट करण्यासाठी शेप स्टाइल ग्रुपमधील पर्याय वापरा.

तुम्ही स्मार्ट आर्ट ग्राफिक कसे घालता आणि फॉरमॅटिंग करता?

SmartArt ग्राफिक इन्सर्ट आणि त्यात टेक्स्ट ऑड करा

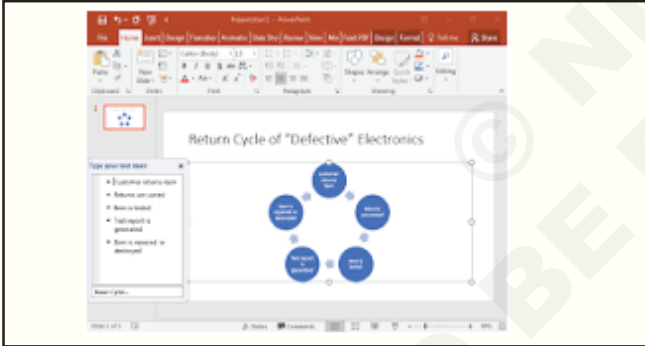
**इन्सर्ट मेनूवर, SmartArt ग्राफिक निवडा. रिबनच्या SmartArt टॅबवर, Insert SmartArt Graphic** गटामध्ये, तुम्हाला हव्या असलेल्या ग्राफिकच्या प्रकारावर निवडा (लिस्ट, प्रक्रिया इ.), आणि नंतर लेआउट निवडा.

PowerPoint मध्ये मी स्मार्ट आर्टचा आकार कसा बदलू शकतो?



SmartArt Tools इंटर्नल , डिझाईन टॅबवर, रीसेट ग्रुपमध्ये, **Convert** वर क्लिक करा आणि नंतर **Convert to Shapes** वर क्लिक करा.तुम्हाला स्मार्टआर्ट टूल्स किंवा डिझाईन टॅब दिसत नसल्यास, तुम्ही स्मार्टआर्ट ग्राफिक निवडले असल्याची खात्री करा. डिझाईन टॅब उघडण्यासाठी तुम्हाला आकारावर डबल-क्लिक करावे लागेल.

PowerPoint मध्ये स्मार्ट आकारांचा यूजेस काय आहे ते उदाहरणासह समजावून सांगा?



SmartArt ही एक डायनॅमिक टाइप ची प्रतिमा आहे जी तुम्हाला PowerPoint स्लाइडमध्ये अनेकदा दिसेल. SmartArt चा वापर इनफार्मेशनच्या याद्या एकत्रित करण्यासाठी, उदाहरणार्थ बुलेट लिस्टद्वारे किंवा सायकलद्वारे प्रक्रिया दर्शविण्यासाठी केला जाऊ शकतो.

**स्मार्टआर्ट ग्राफिकचे टाइप कोणते आहेत?**

लेआउट टाइप . स्मार्टआर्ट ग्राफिक निवडा गॅलरी अकरा वेगवेगळ्या प्रकारांमध्ये विभाट लेले सर्व सबलब्ध लेआउट प्रदर्शित करते -सर्व, लिस्ट , प्रक्रिया, सायकल, पदानुक्रम, संबंध, मॅट्रिक्स, पिरॅमिड, पिक्चर , Office.com आणि इतर.



**PowerPoint मध्ये SmartArt महत्वाचे का आहे?**

स्मार्ट आर्ट संस्था, पदानुक्रम, टप्पे आणि प्रक्रिया तयार करण्यात मदत करते. हे दृष्यदृष्ट्या आनंददायक रंग आणि आकृत्या सादर करते आणि स्पीकर काय म्हणत आहे ते व्हॅल्यू जोडते. आवश्यकता आणि इनफार्मेशनच्या आधारावर अनेक टाइपचे SmartArts वापरता येतात.

## ऑडिओ आणि व्हिडिओ एलिमेंट (Audio & Video Elements)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- PPP मध्ये ऑडिओ ऑड करा किंवा डिलेट करा
- ऑडिओ रेकॉर्ड करा
- प्लेबॅक पर्याय बदला.

### तुमच्या PowerPoint सादरीकरणामध्ये ऑडिओ ऑड करा किंवा डिलेट करा

तुम्ही तुमच्या PowerPoint प्रेझेंटेशनमध्ये ऑडिओ, जसे की संगीत, कथन किंवा ध्वनी जोडू शकता. कोणताही ऑडिओ रेकॉर्ड करण्यासाठी आणि ऐकण्यासाठी, तुमचा संगणक साउंड कार्ड, मायक्रोफोन आणि स्पीकरने सुसज्ज असणे आवश्यक आहे.

### तुमच्या PC वरून ऑडिओ ऑड करा

- 1 निवडा **इन्सर्ट > ऑडिओ**.
- 2 निवडा **माझ्या PC वर ऑडिओ**.
- 3 मध्ये ऑडिओ इन्सर्ट डायलॉग बॉक्समध्ये, तुम्हाला ऑड करायची असलेली ऑडिओ फाइल निवडा.
- 4 इन्सर्ट निवडा.

### ऑडिओ रेकॉर्ड करा

- 1 इन्सर्ट > ऑडिओ निवडा.
- 2 ऑडिओ रेकॉर्ड करा निवडा.
- 3 तुमच्या ऑडिओ फाइलसाठी नाव टाइप करा, रेकॉर्ड निवडा आणि नंतर बोला.

**टीप: ऑडिओ रेकॉर्ड करण्यासाठी तुमच्या डिव्हाइसमध्ये मायक्रोफोन सक्षम असणे आवश्यक आहे.**

- 4 तुमच्या रेकॉर्डिंगचे रिव्यु करण्यासाठी, थांबवा निवडा आणि नंतर प्ले निवडा.
  - 5 तुमची क्लिप पुन्हा रेकॉर्ड करण्यासाठी रेकॉर्ड निवडा, किंवा तुम्ही समाधानी असल्यास ठीक निवडा.
  - 6 तुमची क्लिप हलवण्यासाठी, स्लाइडवर तुम्हाला हच्या ठिकाणी ऑडिओ आयकॉन निवडा आणि ड्रॅग करा.
- तुम्ही प्रति स्लाइड एकापेक्षा जास्त ऑडिओ फाइल वापरत असल्यास, आम्ही ऑडिओ चिन्ह सहजपणे सर्च करण्यासाठी स्लाइडवर त्याच ठिकाणी ठेवण्याची शिफारस करतो.
- 7 प्ले निवडा.

### प्लेबॅक पर्याय बदला

ऑडिओ चिन्ह निवडा आणि नंतर ऑडिओ साधने प्लेबॅक टॅब निवडा. त्यानंतर तुम्ही कोणते पर्याय वापरू इच्छिता ते निवडा:

- ऑडिओ ट्रिम करण्यासाठी, ट्रिम निवडा आणि त्यानंतर ऑडिओ फाइल ट्रिम करण्यासाठी लाल आणि हिरवा स्लाइडर वापरा.
- ऑडिओ फेड इन किंवा फेड आउट करण्यासाठी, फेड कालावधी बॉक्समधील नंबर बदला. • आवाज अडजस्ट करण्यासाठी, व्हॉल्यूम निवडा आणि तुम्हाला प्राधान्य देणारी सेटिंग निवडा.
- ऑडिओ फाइल कशी सुरू होते हे निवडण्यासाठी, ड्रॉपडाउन बाण निवडा आणि एक पर्याय निवडा:
- क्लिक क्रमामध्ये: एका क्लिकने ऑडिओ फाइल ऑटोमैटिक प्ले होते.
- ऑटोमैटिक : ऑडिओ फाइल चालू असताना तुम्ही स्लाइडवर जाता तेव्हा ऑटोमैटिक प्ले होते.
- क्लिक केल्यावर: आयकॉनवर क्लिक केल्यावरच ऑडिओ प्ले होते. • तुमच्या प्रेझेंटेशनमध्ये ऑडिओ कसा प्ले होतो हे निवडण्यासाठी, एक पर्याय निवडा: • स्लाइड्सवर प्ले करा: सर्व स्लाइड्सवर एक ऑडिओ फाइल प्ले करा.
- थांबेपर्यंत लूप: प्ले/पॉज बटणावर क्लिक करून मॅन्युअली थांबेपर्यंत लूपवर ऑडिओ फाइल प्ले करते.
- बॅकग्राउंड तील सर्व स्लाइड्सवर सतत ऑडिओ प्ले होण्यासाठी, बॅकग्राउंड त प्ले करा निवडा.

### ऑडिओ डिलेट करा

ऑडिओ क्लिप हटवण्यासाठी, स्लाइडवरील ऑडिओ चिन्ह निवडा आणि डिलेट करा दाबा.

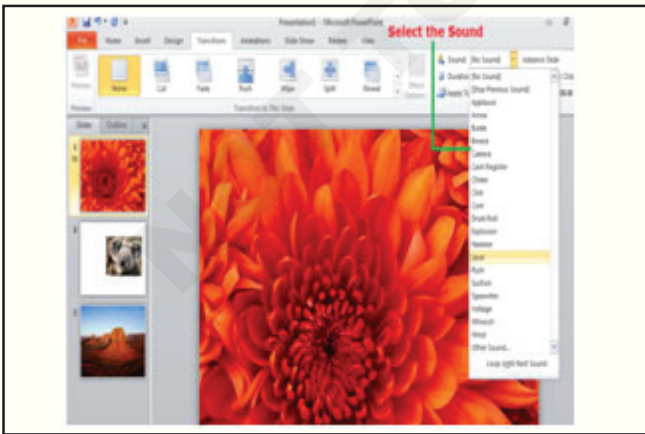
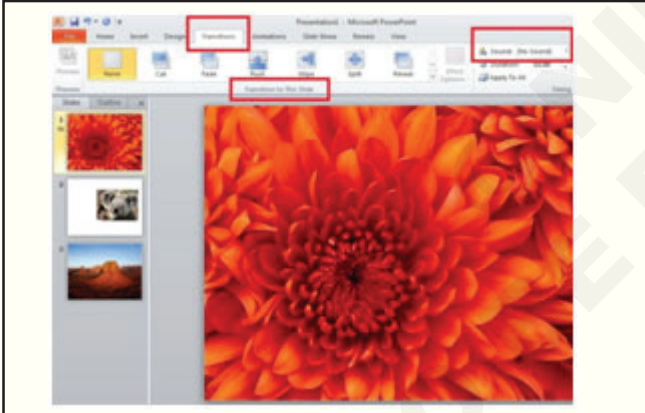
## स्लाइड ट्रांजिशन आणि अॅनिमेशन ऑड करा (Add slide transitions & animations)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- स्लाइड ट्रांजिशन आवाज सेट करा
- स्लाइड ट्रांजिशन इफेक्ट लागू करा
- कस्टम अॅनिमेशन इफेक्ट लागू करा.

स्लाइड ट्रांजिशन साउंड कसा सेट करायचा

- प्रथम, ज्या स्लाइडमध्ये तुम्हाला संक्रमणाचा आवाज ऑड करायचा आहे ती निवडा
- ट्रांजिशन टॅब निवडा
- या स्लाईड गटातील संक्रमणामध्ये ट्रांजिशन साउंड पर्यायाच्या पुढील ड्रॉपडाउन बाणावर क्लिक करा
- अनेक भिन्न ट्रांजिशन ध्वनींची लिस्ट दिसते
- तुमचा इच्छित ट्रांजिशन आवाज निवडा
- सर्व स्लाइड्सवर इफेक्ट लागू करण्यासाठी सर्व लागू करा वर क्लिक करा



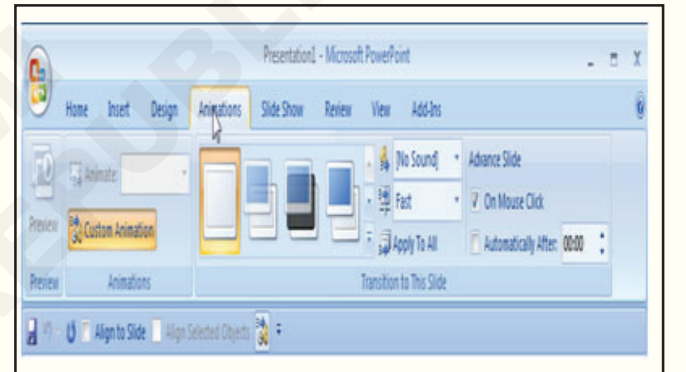
स्लाइड ट्रांजिशन इफेक्ट्स कसे लागू करावे

स्लाइड शोमध्ये एक स्लाइड पुढच्या स्लाइडमध्ये बदलल्यावर सहसा ट्रांजिशन इफेक्ट दिसून येतात.

- प्रथम, ज्या स्लाइडवर तुम्हाला इफेक्ट लागू करायचा आहे ती निवडा

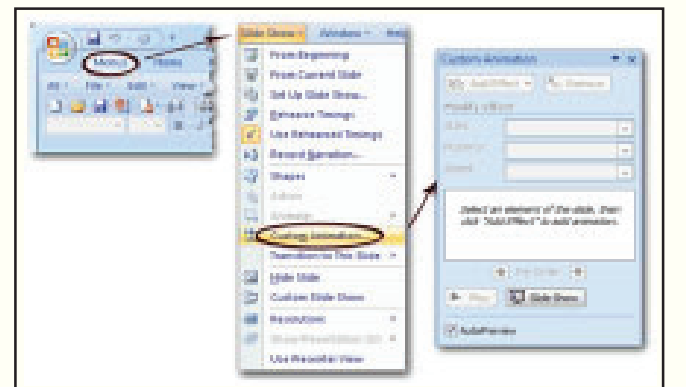
- अॅनिमेशन टॅब निवडा

- आता, या स्लाईड गटातील संक्रमणामध्ये तुम्हाला ट्रांजिशन इफेक्ट दिसतील
- ट्रांजिशन प्रभावांचा मेनू पाहण्यासाठी ड्रॉप-डाउन बाणावर क्लिक करा
- तुमचा इच्छित ट्रांजिशन इफेक्ट निवडा
- सर्व स्लाइड्सवर इफेक्ट लागू करण्यासाठी सर्व लागू करा वर क्लिक करा



कस्टम अॅनिमेशन इफेक्ट कसा लागू करायचा

- प्रथम, आवश्यक टेक्स्ट किंवा ऑब्जेक्ट निवडा
- नंतर, अॅनिमेशन टॅब निवडा
- अॅनिमेशन ग्रुपमध्ये कस्टम अॅनिमेशन वर क्लिक करा
- कस्टम अॅनिमेशन कार्य पॅन आता उजवीकडे दिसेल
- Add Effect वर क्लिक करा, ते आणखी चार पर्याय प्रदर्शित करेल जे पुढे वेगवेगळ्या पर्यायांमध्ये विपार्ट लेले आहेत
- तुमचा इच्छित इफेक्ट निवडा



**उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल (Add and manage comments)**

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कमांड ऑड करा
- कमांड दाखवा किंवा लपवा
- कमांड एडिट करा आणि डिलेट करा.

सादरीकरणामध्ये कमेंट ऑड करा, बदला, लपवा किंवा डिलेट करा

तुम्ही तयार केलेल्या प्रेझेंटेशनचे लोकांनी रिव्यु करावे आणि फीडबॅक द्यावा असे तुम्हाला वाटत असेल किंवा सहकारी प्रेझेंटेशनवर तुमचा फीडबॅक मागतील तेव्हा कमेंट वापरा. कमेंट ही एक टीप आहे जी तुम्ही स्लाइडवरील अक्षर किंवा शब्दाला किंवा संपूर्ण स्लाइडवर जोडू शकता.

**एक कमेंट ऑड करा**

- 1 तुम्हाला कमेंट करायची असलेली वस्तू किंवा स्लाइड निवडा. रिव्यु > नवीन कमेंट निवडा. किंवा कमेंट पॅन उघडल्यास नवीन निवडा. तुम्ही इन्सर्ट > कमेंट निवडून कमेंट देखील जोडू शकता.
- 2 कमेंट पॅन , बॉक्समध्ये तुमचा संदेश टाइप करा आणि पोस्ट निवडा किंवा Ctrl+Enter दाबा.

**कमेंट दर्शवा किंवा लपवा**

- 1 व्ह टॅबवर, सामान्य क्लिक करा.
- 2 रिव्यु टॅबवर, कमेंट दर्शवा क्लिक करा.

**कमेंट पहा आणि प्रत्युत्तर द्या**

- 1 कमेंट सबखंडातील कमेंट निवडा.
- 2 कमेंट ला प्रतिसाद देण्यासाठी प्रत्युत्तर निवडा.

**कमेंट एडिट करा**

- 1 नेव्हिगेशन पॅन , सामान्य दृश्यात, तुम्हाला ज्या स्लाइडवर कमेंट करायची आहे त्यावर क्लिक करा.
- 2 कमेंट पॅन , तुम्हाला एडिट करायच्या असलेल्या कमेंट वर क्लिक करा आणि नंतर बदल करण्यासाठी पेन्सिल चिन्हावर क्लिक करा.

**टीप: लक्षात ठेवा की इतरांना तुमच्या कमेंट एडिट करणे शक्य आहे. ऑफिस डॉक्युमेंट्स तील कमेंट फाइलमध्ये स्टोर केल्या जातात, त्यामुळे तुमच्या फाइलमध्ये एडिटिंग एक्सेस असलेले कोणीही तुमची कमेंट एडिट करू शकतात.**

**कमेंट डिलेट करा**

कमेंट पॅन , तुम्हाला डिलेट करायची असलेली कमेंट निवडा, अधिक थ्रेड क्रिया (...) निवडा आणि थ्रेड डिलेट करा निवडा.

## MySQL मध्ये डेटाबेस इंस्टाल करा, समस्यानिवारण करा, तयार करा आणि वापरा (Install, Troubleshoot, Create and Use of database in MySQL)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- DBMS आणि RDBMS ची संकल्पना
- DBMS आणि RDBMS मधील फरक
- डेटा मॉडेल परिभाषित करा
- DBA ची संकल्पना
- डेटाबेस वापरकर्ते परिभाषित करा
- डेटाबेस स्कीमा परिभाषित करा.

### DBMS आणि RDBMS च्या संकल्पना

रिलेशनल डेटाबेस मॅनेजमेंट सिस्टम (RDBMS) हा प्रोग्राम आणि क्षमतांचा संग्रह आहे जो IT टीम आणि इतरांना रिलेशनल डेटाबेस तयार करण्यास, अपडेट करण्यास, एडमिनिस्ट्रेशन करण्यास आणि अन्यथा संवाद साधण्यास सक्षम करतो. RDBMSes डेटाबेसमध्ये एक्सेस करण्यासाठी स्ट्रक्चर्ड क्वेरी लॅंग्वेज (SQL) वापरून बहुतेक व्यावसायिक रिलेशनल डेटाबेस मॅनेजमेंट सिस्टमसह टेबलच्या स्वरूपात डेटा स्टोर करतात. तथापि, रिलेशनल मॉडेलच्या स्टार्टिंग विकासानंतर SQL चा सर्च लावला गेला असल्याने, RDBMS वापरण्यासाठी ते आवश्यक नाही.

RDBMS ही जगभरातील संस्थांमध्ये सर्वात लोकप्रिय डेटाबेस सिस्टिम आहे. हे सिस्टिम कार्यप्रदर्शन आणि अंमलबजावणी सुलभतेचे संयोजन ऑफर करताना मोठ्या प्रमाणात डेटा संचयित आणि पुनर्प्राप्त करण्याची एक विश्वासार्ह मेथड्स प्रदान करते.

### RDBMS वि. DBMS

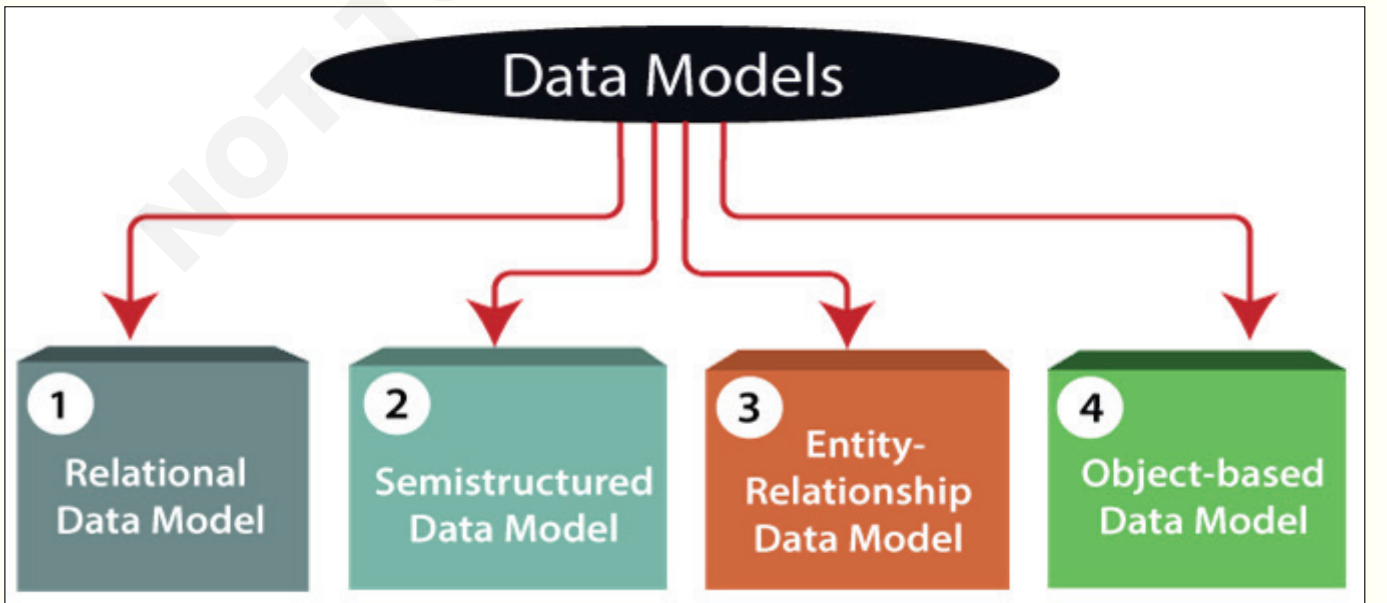
सर्वसाधारणपणे, डेटाबेस डेटाचे संच संचयित करतात जे इतर एप्लिकेशन मध्ये वापरण्यासाठी क्वेरी करू शकतात. डेटाबेस मॅनेजमेंट सिस्टिम डेटाबेस प्लॅटफॉर्मच्या विकास, एडमिनिस्ट्रेशन आणि वापरास सपोर्ट देते.

आरडीबीएमएस हा डेटाबेस मॅनेजमेंट सिस्टिम (डीबीएमएस) चा एक टाइप आहे जो रो-आधारित टेबल स्ट्रक्चरमध्ये डेटा स्टोर करतो जे संबंधित डेटा एलिमेंटांना जोडते. RDBMS मध्ये अशी फंक्शन समाविष्ट असतात जी डेटाची सुरक्षा, अचूकता, अखंडता आणि सातत्य राखतात. हे DBMS मध्ये वापरलेल्या फाइल स्टोरेजपेक्षा वेगळे आहे.

### डेटा मॉडेल्स

डेटा मॉडेल हे डेटा डिस्क्रिप्शन, डेटा अर्थशास्त्र आणि डेटाच्या सातत्य मर्यादांचे मॉडेलिंग आहे. हे डेटा ऑब्स्ट्रॅक्शनच्या प्रत्येक लेयर वर डेटाबेसच्या डिझाइनचे डिस्क्रिप्शन करण्यासाठी संकल्पनात्मक साधने प्रदान करते. म्हणून, डेटाबेसची रचना समजून घेण्यासाठी खालील चार डेटा मॉडेल वापरले जातात:

- 1 रिलेशनल डेटा मॉडेल: या टाइपचे मॉडेल टेबलमधील रो आणि स्तंभांच्या स्वरूपात डेटा डिझाइन करते. अशा प्रकारे, रिलेशनल मॉडेल डेटाचे प्रतिनिधित्व करण्यासाठी आणि संबंधांमधील संबंधांसाठी सारण्या वापरते. टेबलांना संबंध देखील म्हणतात. या मॉडेलचे डिस्क्रिप्शन एडगर एफ. कॉड यांनी 1969 मध्ये केले होते. रिलेशनल डेटा मॉडेल हे मोठ्या प्रमाणावर वापरले जाणारे मॉडेल आहे जे प्रामुख्याने व्यावसायिक डेटा प्रोसेसिंग एप्लिकेशन्सद्वारे वापरले जाते.



- 2 अस्तित्व-संबंध(Entity-Relationship) डेटा मॉडेल:ER मॉडेल म्हणजे डेटाचे ऑब्जेक्ट्स आणि रिलेशनशिप म्हणून लॉजिकल प्रतिनिधित्व. या वस्तूंना अस्तित्व म्हणून ओळखले जाते आणि रिलेशनशिप या एलिमेंटमधील एक संबंध आहे. हे मॉडेल पीटर चेन यांनी डिझाइन केले होते आणि 1976 च्या पेपरमध्ये प्रकाशित केले होते. डेटाबेस डिझायनिंगमध्ये याचा मोठ्या प्रमाणावर वापर केला गेला. गुणधर्मांचा संच एलिमेंटचे डिस्क्रिप्शन करतो. उदाहरणार्थ, विद्यार्थी\_नाव, विद्यार्थी\_आयडी 'विद्यार्थी' एलिमेंटचे डिस्क्रिप्शन करतो. समान टाइप च्या एलिमेंटचा संच 'एंटिटी सेट' म्हणून ओळखला जातो आणि त्याच टाइप च्या संबंधांचा संच 'रिलेशनशिप सेट' म्हणून ओळखला जातो.
- 3 ऑब्जेक्ट-आधारित डेटा मॉडेल:फंक्शनस, एन्कॅप्सुलेशन आणि ऑब्जेक्ट आयडेंटिटीच्या कल्पनांसह ER मॉडेलचा एकसमान्यन . हे मॉडेल समृद्ध टाइप च्या सिस्टिम चे सपोर्ट करते ज्यामध्ये संरचित आणि संग्रह टाइप समाविष्ट आहेत. अशा प्रकारे, 1980 च्या दशकात, ऑब्जेक्ट-ओरिएंटेड पध्दतीचे अनुसरण करणारे विविध डेटाबेस सिस्टिम विकसित केले गेले. येथे, वस्तू काही नसून त्यांचे प्रॉपर्टीज वाहून नेणारा डेटा आहे.
- 4 सेमीस्ट्रक्चर्ड डेटा मॉडेल:या टाइप चे डेटा मॉडेल इतर तीन डेटा मॉडेलपेक्षा वेगळे आहे (वर स्पष्ट केले आहे). सेमीस्ट्रक्चर्ड डेटा मॉडेल अशा ठिकाणी डेटा स्पेसिफिकेशन्सना अनुमती देते जेथे एकाच टाइप च्या वैयक्तिक डेटा आयटममध्ये भिन्न स्पेशल ता सेट असू शकतात. एक्सटेंसिबल मार्कअप लॅंग्वेज, ज्याला XML असेही म्हणतात, अर्धसंरचित डेटाचे प्रतिनिधित्व करण्यासाठी मोठ्या प्रमाणावर वापरले जाते. जरी XML सुरुवातीला टेक्स्ट डॉक्युमेंट्स त मार्कअप इनफार्मेशन समाविष्ट करण्यासाठी डिझाइन केले गेले असले तरी, डेटाच्या देवाणघेवाणीमध्ये त्याचा वापर केल्यामुळे त्याचे महत्त्व प्राप्त होते.

### डीबीएची संकल्पना

असे अनेक टाइप चे डेटाबेस आहेत जे एकमेकांपासून थोडे वेगळे आहेत. जरी डीबीए हे सामान्य डेटाबेस एडमिनिस्ट्रेटर आहेत ज्यांना बहुतेक गोष्टी कशा हाताळतात हे माहित आहे, असे डीबीए आहेत जे स्पेसिफिक डेटाबेस मॅनेज करण्यात अनुभवी आहेत. MySQL DBA एक डेटाबेस एडमिनिस्ट्रेटर आहे जो MySQL डेटाबेस मॅनेज करण्यात तज्ञ आहे.

MySQL हे आजच्या बिग डेटा इकोसिस्टिममधील सर्वात प्रसिद्ध, मोठ्या प्रमाणावर वापरले जाणारे आणि कार्यक्षम तंत्रज्ञानांपैकी एक आहे. एंटरप्राइझ डेटा किंवा सामान्य IT सह निगडित असलेल्या प्रत्येकासाठी MySQL चे बेसिक ज्ञान आवश्यक आहे, आणि ज्यांना रिलेशनल सिस्टिमशी अपरिचित आहे ते देखील MySQL वापरून डेटा स्टोरेज सिस्टिम त्वरीत किकस्टार्ट करू शकतात. या डेटाबेसचे ओपन सोर्स आणि संबंधात्मक फॉर्मॅट हे त्याचे सर्वात मोठे फायदे आहेत, मोठ्या आणि लहान व्यवसायांमध्ये त्याचा वापर करण्याचे कारण आहे.

असे म्हटले जात आहे, आणि जरी MySQL डेटाबेस चांगले दस्तऐवजीकरण मानले गेले आहे आणि त्याला सपोर्ट देण्यासाठी एक मोठा समुदाय आहे, तरीही MySQL डेटाबेस वापरताना काही गोष्टी विचारात घेतल्या पाहिजेत.

डेटाबेस वापरकर्ते:कधीकधी तुम्हाला MySQL मध्ये डेटाबेस मॅनेज करायचा असतो. अशावेळी, आम्हाला डेटाबेसमधील सर्व वापरकर्त्यांच्या खात्यांची यादी पाहण्याची आवश्यकता आहे. बऱ्याच वेळा, डेटाबेस सर्व्हरमध्ये सबलब्ध असलेल्या सर्व वापरकर्त्यांची लिस्ट प्रदर्शित करण्यासाठी डेटाबेस दाखवा, टेबल दाखवा इ. सारखीच एक SHOW USERS कमांड आहे असे आम्ही गृहीत धरतो. दुर्दैवाने, MySQL सर्व्हरमधील सर्व वापरकर्त्यांची लिस्ट प्रदर्शित करण्यासाठी MySQL डेटाबेसमध्ये SHOW USERS कमांड नाही. डेटाबेस सर्व्हरमधील सर्व वापरकर्त्यांची यादी पाहण्यासाठी आम्ही खालील केरी वापरू शकतो:

**mysql> mysql.user वरून युजर निवडा;**

वरील विधानाच्या यशस्वी अंमलबजावणीनंतर, आम्हाला MySQL डेटाबेस सर्व्हरच्या युजर टेबल वरून युजर डेटा मिळेल.

### डेटाबेस स्कीमा

mysql स्कीमा ही सिस्टिम स्कीमा आहे. यात टेबल आहेत जी MySQL सर्व्हर चालू असताना आवश्यक असलेली इनफार्मेशन साठवतात. विस्तृत वर्गीकरण असे आहे की mysql स्कीमामध्ये डेटा डिक्शनरी टेबल्स असतात ज्यात डेटाबेस ऑब्जेक्ट मेटाडेटा स्टोर केला जातो आणि इतर ऑपरेशनल उद्देशांसाठी वापरल्या जाणाऱ्या सिस्टिम टेबल्स असतात. पुढील चर्चा सिस्टिम टेबलच्या संचाला आणखी लहान वर्गामध्ये स्प्लिट करते.

- डेटा डिक्शनरी टेबल्स
- सिस्टिम टेबल्स द्या
- ऑब्जेक्ट इन्फार्मेशन सिस्टिम टेबल्स
- लॉग सिस्टिम टेबल्स
- सर्व्हर-साइड हेलप सिस्टिम टेबल्स
- टाइम झोन सिस्टिम टेबल्स
- प्रतिकृती सिस्टिम सारण्या
- ऑप्टिमायझर सिस्टिम टेबल्स
- विविध सिस्टिम सारण्या

या विभागाचा उर्वरित पार्ट अतिरिक्त इनफार्मेशन साठी क्रॉस संदर्भांसह, प्रत्येक रेन्ज तील सारण्यांची गणना करतो. डेटा डिक्शनरी टेबल्स आणि सिस्टिम टेबल्स अन्यथा सूचित केल्याशिवाय InnoDB स्टोरेज इंजिन वापरतात.

mysql सिस्टिम टेबल्स आणि डेटा डिक्शनरी टेबल्स MySQL डेटा डिरेक्टरीमध्ये mysql.ibd नावाच्या एका InnoDB टेबलस्पेस फाइलमध्ये राहतात. पूर्वी, या टेबल्स mysql डेटाबेस डिरेक्टरीमध्ये वैयक्तिक टेबलस्पेस फाइल्समध्ये तयार केल्या होत्या.

डेटा-अॅट-रेस्ट एन्क्रिप्शन mysql सिस्टिम स्कीमा टेबलस्पेससाठी सक्षम केले जाऊ शकते. अधिक इनफार्मेशन साठी, विपार्ट 15.13, "InnoDB Data-at-Rest Encryption" पहा.



**नॉर्मलायजेशन नियम, विविध डेटाटाइप, डेटा इंटिग्रिटी, डीडीएल, डीएमएल आणि डीसीएल स्टेटमेंट वापरून डेटाबेस डिझाइन करणे प्रायमरी की आणि फॉरेन की (Designing data-base using normalization rules, various datatypes, data integrity, DDL, DML&DCL Statements Enforcing Primary key and Foreign key)**

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- MySQL मध्ये डेटाबेस नॉर्मलायजेशन
- विविध डेटाटाइप
- डेटा इंटिग्रिटी
- MySQL मध्ये DDL, DML आणि DCL स्टेटमेंट्स डेटाबेस नॉर्मलायजेशन
- प्रायमरी की आणि फॉरेन की लागू करणे
- इंडिसेस जोडणे.

**MySQL मध्ये डेटाबेस नॉर्मलायजेशन**

डेटाबेस डिझाइन किंवा डेटा मॉडेलिंगमध्ये डेटाबेस नॉर्मलायजेशन हा सर्वात महत्वाचा एलिमेंट आहे. डेटाबेस नॉर्मलायझेशन ही डेटा रिडंडंसी दूर करण्यासाठी आणि डेटा मॅनेजमेंट सुलभ करण्यासाठी लॉजिकल रिट्या डेटा स्टोर करण्याची प्रक्रिया आहे.

डेटाबेस संबंध आणि कळा डेटाबेस नॉर्मलायजेशन प्रक्रियेत सबयुक्त आहेत. डेटाबेस नॉर्मलायजेशन E.F.Codd द्वारे विकसित केले गेले. डेटाबेस नॉर्मलायजेशन प्रक्रियेत, सामान्य फॉर्म नावाच्या नियमांची सीरीज आहे.

नॉर्मल फॉर्मचे प्रामुख्याने 6 टाइप आहेत: पहिला नॉर्मल फॉर्म (1NF), दुसरा नॉर्मल फॉर्म (2NF), तिसरा नॉर्मल फॉर्म (3NF), चौथा नॉर्मल फॉर्म (4NF), पाचवा नॉर्मल फॉर्म (5NF), आणि बॉयस कॉड नॉर्मल फॉर्म (BCNF). परंतु मुख्यतः आम्ही आमच्या डेटाबेस डिझाइनमध्ये तिसऱ्या सामान्य स्वरूपाचा वापर केला.

- 1 फर्स्ट नॉर्मल फॉर्म (1NF): पहिल्या सामान्य फॉर्ममध्ये, प्रत्येक कॉलममध्ये फक्त एक व्हॅल्यू असणे आवश्यक आहे आणि कोणत्याही टेबलमध्ये संबंधित डेटाचे पुनरावृत्ती होणारे ग्रुप स्टोर करू नये.
- 2 सेकंड नॉर्मल फॉर्म (2NF): दुसऱ्या नॉर्मल फॉर्ममध्ये, प्रथम डेटाबेस पहिल्या नॉर्मल फॉर्ममध्ये असणे आवश्यक आहे, त्याच टेबलमध्ये डुप्लिकेट रो स्टोर करू नये. आणि रो मध्ये डुप्लिकेट व्हॅल्यू असल्यास, ती त्यांच्या स्वतःच्या वेगळ्या टेबलमध्ये स्टोर केली पाहिजेत आणि फॉरेन की वापरून टेबलशी लिंक केली पाहिजेत. दुसऱ्या सामान्य स्वरूपात डेटाबेसचा आदर्श पाथ म्हणजे एक ते अनेक संबंध सारण्या तयार करणे.
- 3 थर्ड नॉर्मल फॉर्म (3NF): तिसऱ्या नॉर्मल फॉर्ममध्ये, डेटाबेस आधीपासूनच तिसऱ्या फॉर्ममध्ये असतो, जर तो दुसऱ्या नॉर्मल फॉर्ममध्ये असेल आणि प्रत्येक नॉन-की कॉलम परस्पर स्वतंत्र असेल. टेबलमधील कोणतेही कॉलम ओळखा जे एकमेकांवर अवलंबून आहेत आणि त्या स्तंभांना त्यांच्या स्वतःच्या वेगळ्या सारण्यांमध्ये स्लिट करा.

- 4 बॉयस कॉड नॉर्मल फॉर्म (BCNF): हा तिसऱ्या सामान्य स्वरूपाचा सर्वोच्च टाइप आहे जो 3NF द्वारे हाताळल्या जात नसलेल्या विविध टाइप च्या विसंगतींशी संबंधित आहे.

**विविध डेटा टाइप**

**इनफार्मेशन इंटिग्रिटी**

डेटा अखंडता म्हणजे कोणत्याही डेटाबेस किंवा वेअरहाऊसमध्ये साठवलेल्या डेटाची सातत्य, अचूकता आणि विश्वासार्हता. सर्व वैशिष्ट्ये अचूक असलेली संपूर्ण रचना असलेला डेटा अखंडतेसह डेटा असल्याचे म्हटले जाते.

डेटाच्या अखंडतेचे अनेक पैलू आहेत जसे की फिजिकल अखंडता, ज्यामध्ये डेटा स्टोर करणे आणि ते प्रमाणिकरित्या गोळा करणे समाविष्ट आहे, त्यानंतर लॉजिकल अखंडता येते, ज्यामध्ये इनफार्मेशन स्पेसिफिक संदर्भित संबंधित आणि अचूक आहे की नाही हे तपासणे समाविष्ट आहे किंवा एकूणच सर्व नियम जे डेटाची गुणवत्ता राखण्यासाठी डेटा इंटिग्रिटी इंटरनल येतो. अखंडता आणि अचूकतेशिवाय, तुमचा सर्व गोळा केलेला डेटा कंपनीसाठी निरुपयोगी आहे, त्यामुळे डेटा प्रोटेक्शन आणि त्याची कार्यक्षमता आणि स्थिरता वाढवण्यासाठी त्याची अचूकता सुनिश्चित करणे आवश्यक आहे. करप्टेड डेटा काहीवेळा तुमच्या व्यवसायाचे नुकसान देखील करू शकतो.

डेटा बदलणे सोपे नाही कारण डेटा स्थिर नाही. तुम्हाला मिळालेली इनफार्मेशन विविध ठिकाणांहून येते आणि अनेक गोष्टी तुमचा डेटा तयार केलेल्या साइटवरून बदलू शकतात. याव्यतिरिक्त, ते इतर उपकरणांवर हस्तांतरित केले जाऊ शकते, बदलले जाऊ शकते आणि कोणाकडून आणि जेव्हा आवश्यक असेल तेव्हा अपडेट केले जाऊ शकते.

आणि डेटा अखंडता सुनिश्चित करणे ही केवळ एक-चरण प्रक्रिया नाही. मॉडेलच्या डिझाइनपासून त्याच्या अंतिम आउटपुटपर्यंत प्रत्येक टप्प्यावर त्याची तपासणी आणि कन्फर्म करणे आवश्यक आहे.

**DDL, DML आणि DCL स्टेटमेंट:** स्ट्रक्चर्ड क्वेरी लॅंग्वेज (SQL) ही डेटाबेस भाषा आहे ज्याचा वापर करून आपण सध्याच्या डेटाबेसवर काही ऑपरेशन्स करू शकतो आणि डेटाबेस तयार करण्यासाठी देखील ही भाषा वापरू शकतो. आवश्यक फंक्शन पार पाडण्यासाठी एसक्यूएल क्रिएट, ड्रॉप, इन्सर्ट इत्यादी काही कमांड्स वापरते.

या SQL कमांड्सचे मुख्यत्वे चार प्रकारांमध्ये वर्गीकरण केले आहे:

- 1 DDL - डेटा व्याख्या भाषा
- 2 DML - डेटा मॅनिप्युलेशन भाषा
- 3 DCL - डेटा कंट्रोल भाषा

### DDL (डेटा डेफिनिशन लॅंग्वेज)

डीडीएल किंवा डेटा डेफिनिशन लॅंग्वेजमध्ये एसक्यूएल कमांड्स असतात ज्यांचा वापर डेटाबेस स्कीमा परिभाषित करण्यासाठी केला जाऊ शकतो. हे फक्त डेटाबेस स्कीमाच्या कॅरक्टर नाशी संबंधित आहे आणि डेटाबेसमधील डेटाबेस ऑब्जेक्ट्सची रचना तयार करण्यासाठी आणि सुधारण्यासाठी वापरले जाते. DDL हा डेटाबेस स्ट्रक्चर तयार करण्यासाठी, सुधारण्यासाठी आणि हटवण्यासाठी वापरल्या जाणाऱ्या SQL कमांडचा संच आहे परंतु डेटा नाही. या आज्ञा सामान्यतः सामान्य यूजर द्वारे वापरल्या जात नाहीत, जो अनुप्रयोगाद्वारे डेटाबेसमध्ये एक्सेस करत असावा.

### DDL कमांड ची यादी

- **क्रिएट :** ही कमांड डेटाबेस किंवा त्याच्या ऑब्जेक्ट्स (जसे की टेबल, इंडेक्स, फंक्शन, व्ह्यू, स्टोअर प्रक्रिया आणि ट्रिगर) तयार करण्यासाठी वापरली जाते.
- **ड्रॉप:** ही कमांड डेटाबेसमधून ऑब्जेक्ट्स हटवण्यासाठी वापरली जाते.
- **आल्टर:** हे डेटाबेसच्या संरचनेत बदल करण्यासाठी वापरले जाते.
- **ट्रंकेट:** याचा यूजेस टेबलमधून सर्व रेकॉर्ड काढून टाकण्यासाठी केला जातो, ज्यामध्ये रेकॉर्डसाठी वाटप केलेल्या सर्व जागा काढून टाकल्या जातात.
- **कमेंट :** डेटा शब्दकोशात कमेंट जोडण्यासाठी याचा वापर केला जातो
- **रिनेम:** हे डेटाबेसमध्ये अस्तित्वात असलेल्या ऑब्जेक्टचे नाव बदलण्यासाठी वापरले जाते.

### डीएमएल (डेटा मॅनिप्युलेशन भाषा)

डेटाबेसमध्ये सबस्थित असलेल्या डेटाच्या हाताळणीशी संबंधित SQL कमांड्स DML किंवा डेटा मॅनिप्युलेशन लॅंग्वेजशी संबंधित आहेत आणि यामध्ये बहुतांश SQL स्टेटमेंट समाविष्ट आहेत. एसक्यूएल स्टेटमेंटचा हा एलिमेंट आहे जो डेटा आणि डेटाबेसमध्ये एक्सेस कंट्रोल करतो. मुळात, डीसीएल स्टेटमेंट्स डीएमएल स्टेटमेंटसह ग्रुप बद्ध केली जातात.

### DML कमांड ची यादी

- **इन्सर्ट :** हे टेबलमध्ये डेटा घालण्यासाठी वापरले जाते.

- **अपडेट:** हे टेबलमधील विद्यमान डेटा अपडेट करण्यासाठी वापरले जाते.
- **डिलेट करा:** हे डेटाबेस टेबलमधून रेकॉर्ड हटवण्यासाठी वापरले जाते.
- **लॉक:** टेबल कंट्रोल कॉन्करन्सी.
- **कॉल करा:** PL/SQL किंवा JAVA सबप्रोग्रामवर कॉल करा.
- **योजना स्पष्ट करा:** हे डेटाच्या एक्सेस मार्गाचे डिस्क्रिप्शन करते.

### DCL (डेटा कंट्रोल लॅंग्वेज)

DCL मध्ये GRANT आणि REVOKE सारख्या कमांड चा समावेश आहे जे मुख्यतः डेटाबेस सिस्टमचे अधिकार, परवानग्या आणि इतर कंट्रोल हाताळतात.

### DCL कमांड ची यादी

- **ग्रान्ट:** ही आज्ञा वापरकर्त्यांना डेटाबेसमध्ये एक्सेस करण्याचे विशेषाधिकार देते.
- **मागे घेणे:** ही कमांड GRANT कमांड वापरून यूजर चे एक्सेस विशेषाधिकार काढून घेते.

### प्रायमरी की

प्रायमरी की मर्यादा टेबलवर प्रायमरी की तयार करून परिभाषित केली जाते. प्रायमरी की मधील मूल्याने टेबल तील प्रत्येक रो अद्वितीयपणे ओळखली पाहिजे. प्रायमरी की मर्यादा रो अखंडतेची अंमलबजावणी करते.

### प्रायमरी व्याख्या करताना खालील तथ्ये विचारात घ्या की:

- जेव्हा तुम्ही टेबलवर प्रायमरी की तयार करता, तेव्हा एक युनिक इंडेक्स (आणि म्हणूनच युनिक कंस्ट्रेंट) ऑटोमेटिक आणि अस्पष्टपणे तयार होते जे या कॉलममधील डेटाची स्पेसिफिक ता लागू करते.
- प्रायमरी की कॉलम त NULL व्हॅल्यू असू शकत नाहीत. डेटाबेस इंजिन प्रायमरी की कॉलमवर अस्पष्टपणे शून्य नसलेला प्रतिबंध तयार करते.
- टेबलवर फक्त एक प्रायमरी की परिभाषित केली जाऊ शकते.
- प्रायमरी की एका कॉलम द्वारे किंवा अनेक स्तंभांच्या (म्हणजे संयुक्त प्रायमरी की) बनवता येते.
- तुम्ही प्रायमरी की मध्ये डुप्लिकेट रेकॉर्ड टाकल्यास, तुम्हाला एरर मिळेल. प्रायमरी कीचे दोन टाइप आहेत: कृत्रिम प्रायमरी की आणि नैसर्गिक प्रायमरी की.फॉरेन की

MySQL फॉरेन की ला सपोर्ट करते, जे टेबलवर क्रॉसरेफरन्सिंग संबंधित डेटाला परवानगी देते आणि फॉरेन की कंस्ट्रेंट्स, जे संबंधित डेटा सुसंगत ठेवण्यास मदत करते.

फॉरेन की रिलेशनशिपमध्ये स्टार्टिंग कॉलम व्हॅल्यूज असणारी पॅरेंट टेबल आणि पॅरेंट कॉलम व्हॅल्यूजचा रेफरन्स देणारी कॉलम व्हॅल्यूज असलेली चाइल्ड टेबल समाविष्ट असते. चाइल्ड टेबलवर फॉरेन की मर्यादा परिभाषित केली आहे.

## केरी इन्सर्ट आणि डिलेट करा केरी अपडेट करा (Insert and delete queries Update queries)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- सिंपल सिलेक्ट केरी.

### इंडिसेस अड करा

#### ट्रांसक्शन ची संकल्पना विचारा

ट्रांसक्शन हा डेटाबेस मॅनिपुलेशन ऑपरेशन्सचा एक क्रमिक ग्रुप असतो, जो एकच कार्य युनिट असल्याप्रमाणे केला जातो. दुसऱ्या शब्दांत सांगायचे तर, समूहातील प्रत्येक वैयक्तिक ऑपरेशन यशस्वी झाल्याशिवाय व्यवहार कधीही पूर्ण होणार नाही. ट्रांसक्शन तील कोणतेही ऑपरेशन अयशस्वी झाल्यास, संपूर्ण व्यवहार अयशस्वी होईल.

व्यावहारिकदृष्ट्या, तुम्ही अनेक SQL केरी एका गटात एकत्र कराल आणि ट्रांसक्शन चा एक पार्ट म्हणून तुम्ही त्या सर्व एकत्र कराल.

#### ट्रांसक्शन चे प्रॉपर्टीज

ट्रांसक्शन मध्ये खालील चार स्टॅण्डर्ड प्रॉपर्टीज असतात, ज्याला सामान्यतः ACID या संक्षेपाने संदर्भित केले जाते:

- **ऑटोमिसिटी** -हे सुनिश्चित करते की कार्य युनिटमधील सर्व ऑपरेशन्स यशस्वीरित्या पूर्ण झाल्या आहेत; अन्यथा, व्यवहार अयशस्वी झाल्याच्या वेळी रद्द केला जातो आणि मागील ऑपरेशन्स त्यांच्या पूर्वीच्या स्थितीत परत आणल्या जातात.
- **कन्सिस्टेन्सी** -हे सुनिश्चित करते की डेटाबेस यशस्वीरित्या वचनबद्ध ट्रांसक्शन वर राज्ये योग्यरित्या बदलतो.
- **आइसोलेशन** -हे व्यवहार स्वतंत्रपणे आणि एकमेकांशी पारदर्शकपणे कार्य करण्यास सक्षम करते.
- **ड्युरेबिलिटी** -हे सुनिश्चित करते की सिस्टम अयशस्वी झाल्यास वचनबद्ध ट्रांसक्शन चा रिझल्ट किंवा रिझल्ट कायम राहतो.

नंबर , date आणि कॅरक्टर फंक्शन्स वापरून, सब क्वेरी करून ग्रुप करा (Using the Number, Date and Character functions, group by having, sub query)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- सब क्वेरी सांगा.

MySQL जॉईनिंग टेबल्स

JOIN क्लॉजचा वापर दोन किंवा अधिक सारण्यांमधील रो एकत्र करण्यासाठी केला जातो, त्यांच्यामधील संबंधित कॉलम वर आधारित.

चला "ऑर्डर्स" टेबल मधून निवड पाहू:

ऑर्डर आयडी	ग्राहक आयडी	मागणीची तारीख
10308	2	1996-09-19
10309	37	1996-09-19
10310	77	1996-09-20

त्यानंतर, "ग्राहक" टेबल मधून निवड पाहा: लक्षात घ्या की "ऑर्डर्स"

ग्राहक आयडी	ग्राहकाचे नाव	संपर्क नाव	देश
1	आल्फ्रेडस फ्युटरकिस्ते	मारिया अँडर्स	जर्मनी
2	अँना टुजिलो एम्पारेडाडोस येहेलाडोस	आना टुजिलो	मेक्सिको
3	अँटोनियो मोरेनो टाकेरिया	अँथनी मोरेनो	मेक्सिको

टेबल तील "CustomerID" कॉलम "ग्राहक" टेबल मधील "CustomerID" चा रेफरेन्स देतो. वरील दोन सारण्यांमधील संबंध "CustomerID" कॉलम आहे.

त्यानंतर, आम्ही खालील SQL स्टेटमेंट तयार करू शकतो (ज्यात INNER JOIN आहे), जे दोन्ही टेबल्समध्ये जुळणारी व्हॅल्यू असलेले रेकॉर्ड निवडते:

उदाहरण

ऑर्डर निवडा.ऑर्डरआयडी, ग्राहक.ग्राहकांचे नाव, ऑर्डर.ऑर्डर्समधून ऑर्डरची date

ग्राहकांना ऑर्डरवर आतील सहभागी व्हा.CustomerID=Customer.CustomerID;

ते स्वतः वापरून पाहा" आणि ते असे काहीतरी तयार करेल

ऑर्डरआयडी	ग्राहकाचे नाव	मागणीची तारीख
10308	आना टुजिलो सँडविच आणि आइस्क्रीम	9/18/1996
10365	अँटोनियो मोरेनो टॅकेरिया	11/27/1996
10383	हॉर्नच्या आसपास	12/16/1996
10355	हॉर्नच्या आसपास	11/15/1996
10278	बर्गलुंडचे सुपरमार्केट	8/12/1996

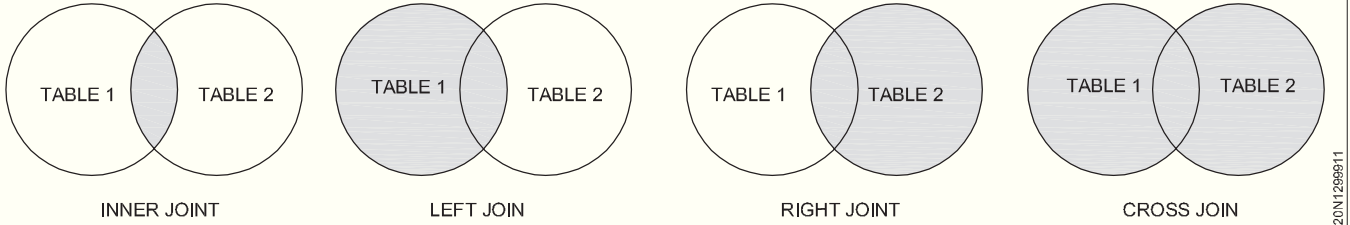
MySQL मध्ये सामील होण्याचे समर्थित टाइप

- इनर जॉईन: दोन्ही सारण्यांमध्ये जुळणारी व्हॅल्यू असलेले रेकॉर्ड परत करते
- डावीकडे सामील होणे: डाव्या तक्त्यातील सर्व नोंदी आणि उजव्या टेबल वरून जुळलेले रेकॉर्ड परत करते
- राईट जॉईन: उजव्या टेबलवरून सर्व रेकॉर्ड आणि डाव्या टेबलवरून जुळलेले रेकॉर्ड परत करते
- क्रॉस जॉईन: दोन्ही सारण्यांवरील सर्व रेकॉर्ड परत करते

सब क्वेरी

MySQL मधील सबक्वेरी ही एक क्वेरी आहे, जी दुसऱ्या SQL क्वेरीमध्ये नेस्ट केलेली असते आणि विविध ऑपरेटर्ससह SELECT, INSERT, UPDATE किंवा DELETE स्टेटमेंटसह एम्बेड केलेली असते. आपण दुसऱ्या सबक्वेरीसह सबक्वेरी नेस्ट देखील करू शकतो. सबक्वेरी आतील क्वेरी म्हणून ओळखली जाते आणि सबक्वेरी असलेली क्वेरी एक्सटर्नल क्वेरी म्हणून ओळखली जाते. अंमलात आणलेली इंटरनल क्वेरी प्रथम एक्सटर्नल क्वेरीला निकाल देते आणि नंतर मुख्य/एक्सटर्नल क्वेरी केली जाईल. MySQL आम्हाला सबक्वेरी कुठेही वापरण्याची परवानगी देते, परंतु ती कंसात बंद केली पाहिजे. SQL मानकाद्वारे समर्थित सर्व सबक्वेरी फॉर्म आणि ऑपरेटर्ससह MySQL मध्ये देखील समर्थित असतील.

Fig 1



CP20N129911

### सबक्केरी वापरण्याचे नियम खालीलप्रमाणे आहेत

- सबक्केरी नेहमी कंसात वापरल्या पाहिजेत.
- जर मुख्य क्केरीमध्ये सबक्केरीसाठी एकापेक्षा जास्त कॉलम नसतील, तर सबक्केरीमध्ये SELECT कमांडमध्ये फक्त एक कॉलम असू शकतो.
- आम्ही सबक्केरीसह विविध कम्प्यारीजन ऑपरेटर वापरू शकतो, जसे की >, <, =, IN, ANY, SOME, आणि ALL. जेव्हा सबक्केरी एकापेक्षा जास्त रोजे परत करते तेव्हा मल्टिपल - रोजे ऑपरेटर खूप सबयुक्त आहे.
- आम्ही सबक्केरीमध्ये ऑर्डर बाय क्लॉज वापरू शकत नाही, जरी ते मुख्य क्केरीमध्ये वापरले जाऊ शकते.
- जर आपण सेट फंक्शनमध्ये सबक्केरी वापरतो, तर ती लगेच सेट फंक्शनमध्ये बंद केली जाऊ शकत नाही.

### सबक्केरीज वापरण्याचे खालील फायदे आहेत

- सबक्केरीज संरचित स्वरूपात क्केरी बनवतात ज्यामुळे आम्हाला विधानाचा प्रत्येक पार्ट वेगळा करता येतो.
- सबक्केरीज टेबलमधील डेटाची क्केरी करण्याचे पर्यायी पाथ प्रदान करतात; अन्यथा, आम्हाला जटिल जोड आणि युनियन वापरण्याची आवश्यकता आहे.
- सबक्केरीज जटिल सामील किंवा युनियन विधानांपेक्षा अधिक वाचनीय (रिडेबल) आहेत.

### MySQL सबक्केरी सिंटॅक्स

MySQL मध्ये सबक्केरी वापरण्यासाठी खालील बेसिक वाक्यरचना आहे:

**SELECT** column\_list(s) from table\_name

**WHERE** कॉलम\_नाव OPERATOR

(SELECT column\_list(s) FROM table\_name [WHERE])

### क्केरीमध्ये वापरलेली फंक्शन

- 1 बेरीज;  
व्हॅल्यू च्या संचाच्या बेरजेची गणना करते  
syntax: SUM(expression)
- 2 average (avg):  
expression चे सरासरी व्हॅल्यू मिळवते  
syntax: AVG(expression)
- 3 Max:  
व्हॅल्यू च्या संचामध्ये कमाल व्हॅल्यू मिळवते  
syntax: MAX(expression)
- 4 MIN :  
व्हॅल्यू च्या संचामध्ये किमान व्हॅल्यू मिळवते  
सिंटॅक्स: MIN(expression)
- 5 Count:  
निवडलेल्या क्केरीद्वारे परत केलेल्या रेकॉर्डची नंबर मिळवते  
सिंटॅक्स: COUNT(expression)

## संगणकाला नेटवर्कशी कनेक्ट करा आणि डिव्हाइसेस उदा. प्रिंटर, फाइल्स, फोल्डर्स आणि ड्राइव्ह शेअर करा (Connect a computer to a network and share Devices i.e. Printers, files, folders and drives)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- लोकल नेटवर्क.
- लोकल नेटवर्कवर कम्प्युनिकेशन करणे:
- कम्प्युनिकेशन तत्त्वे.

### जोडलेल्या जगात कम्प्युनिकेशन

#### लोकल नेटवर्क.

लोकल एरिया नेटवर्क (LAN) हा संगणक आणि पेरीफेरल उपकरणांचा समूह आहे जो एका स्पेसिफिक भौगोलिक क्षेत्रामध्ये सर्व्हरशी एक सामान्य कम्प्युनिकेशन लाइन किंवा वायरलेस लिंक सामायिक करतो. लोकल एरिया नेटवर्क होम ऑफिसमध्ये दोन किंवा तीन वापरकर्ते किंवा कॉर्पोरेशनच्या सेंट्रल ऑफिसमध्ये हजारो वापरकर्ते देऊ शकतात.

#### लोकल नेटवर्कवर कम्प्युनिकेशन :

एक पीअर-टू-पीअर LAN थेट इथरनेट केबल वापरून दोन उपकरणांना सामान्यतः वर्कस्टेशन्स किंवा वैयक्तिक संगणकांना जोडतो. क्लायंट-सर्व्हर LAN मध्ये मल्टिपल एंडपॉइंट आणि सर्व्हर असतात जे LAN स्विचशी कनेक्ट केलेले असतात. स्विच मल्टिपल कनेक्ट केलेल्या उपकरणांमधील कम्प्युनिकेशन प्रवाह निर्देशित करतो.

#### कम्प्युनिकेशन ची तत्त्वे:

कम्प्युनिकेशन व्यावसायिक कधीकधी कम्प्युनिकेशन च्या 7 तत्त्वांचा

### इथरनेट नेटवर्क (Ethernet networks)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- इथरनेट परिभाषित करा
- इथरनेट कसे कार्य करते
- नेटवर्क बिल्ट परिभाषित करा.

इथरनेट हे मूळ नेटवर्किंग तंत्रज्ञानांपैकी एक आहे, ज्याचा सर्च जवळपास 50 वर्षांपूर्वी लागला होता. आणि तरीही, साधेपणामुळे ज्याद्वारे कम्प्युनिकेशन प्रोटोकॉल तैनात केला जाऊ शकतो आणि आधुनिक प्रगतीचा समावेश करण्याच्या क्षमतेमुळे, संगणक नेटवर्किंगसाठी इथरनेट हे वास्तविक स्टॅण्डर्ड म्हणून राज्य करत आहे.

इथरनेट हा एक प्रोटोकॉल आहे जो संगणकांना (सर्व्हरपासून लॅपटॉपपर्यंत) वायर्ड नेटवर्कवर एकमेकांशी बोलण्याची परवानगी देतो जे थेट रहदारीसाठी राउटर, स्विचेस आणि हब सारख्या उपकरणांचा वापर करतात. इथरनेट वायरलेस प्रोटोकॉलसह अखंडपणे कार्य करते.

जवळजवळ कोणत्याही वातावरणात काम करण्याच्या क्षमतेमुळे जगभरात त्याचा सार्वत्रिक अवलंब झाला आहे. हे स्पेशल तः खरे आहे कारण ते संस्थांना त्यांच्या लोकल एरिया नेटवर्क (LAN) आणि त्यांच्या वाइड-एरिया

रेफरेन्स घेतात; दुर्दैवाने, या प्रमाणित नाहीत (त्यामुळे कदाचित पातळी 200!), परंतु काही सामान्य थीम आहेत ज्या Microsoft 365 मॅज्युरिटी मॉडेलमध्ये कम्प्युनिकेशन सक्षमतेची इनफार्मेशन देण्यासाठी वापरल्या गेल्या आहेत. जर तुम्ही कम्प्युनिकेशन व्यावसायिक नसाल तर तुम्हाला कम्प्युनिकेशन धोरण, परिपक्वता असेसमेंट किंवा तंत्रज्ञान निवडीचा पार्ट म्हणून तत्त्वांचे हे स्पष्टीकरण सबयुक्त वाटू शकते.

- 1 प्रभावी
- 2 सर्वसमावेशक
- 3 स्पष्टता
- 4 लक्ष आणि स्टाइल
- 5 सुसंगतता
- 6 समयसूचकता आणि निकड
- 7 अभिप्रायाचे महत्त्व

नेटवर्क (WAN) मध्ये समान इथरनेट प्रोटोकॉल वापरण्याची परवानगी देते. याचा अर्थ असा की ते डेटा सेंटरमध्ये, प्राइवेट किंवा इंटरनल कंपनी नेटवर्कमध्ये, इंटरनेट ऍप्लिकेशन्ससाठी आणि जवळपास कोणत्याही गोष्टीसाठी चांगले कार्य करते. हे व्हर्च्युअल प्रायव्हेट नेटवर्क (VPN) आणि सॉफ्टवेअर-परिभाषित नेटवर्किंग सबयोजनांसारख्या नेटवर्किंगच्या सर्वात जटिल प्रकारांना देखील सपोर्ट देऊ शकते.

इथरनेटला व्हिडिओ स्ट्रीमिंग किंवा व्हॉइस ओव्हर आयपी ऍप्लिकेशन्स सारख्या बँडविड्थ-केंद्रित ऍप्लिकेशन्स हाताळण्यात कोणतीही समस्या नाही. आणि दुसऱ्या बाजूला, त्याची साधेपणा कोणत्याही स्पेशल कॉन्फिगरेशनशिवाय इंटरनेट ऑफ थिंग्ज (IoT) सारख्या अतिशय लहान, तुलनेने अत्याधुनिक उपकरणांसह कार्य करण्यास सक्षम करते.

## इंटरनेट कसे कार्य करते?

इंटरनेट पर्सनल कॉम्प्युटर सारख्या उपकरणांवर किंवा वरून पाठवल्या जाणाऱ्या इनफार्मेशन चे वेगवेगळ्या आकाराच्या इनफार्मेशन च्या छोट्या तुकड्यांमध्ये पार्टीशन करून कार्य करते ज्याला फ्रेम म्हणतात. त्या फ्रेममध्ये सोर्स आणि डेस्टिनेशन एड्रेस यासारखी प्रमाणित इनफार्मेशन असते जी फ्रेमला नेटवर्कमधून पाथ काढण्यास मदत करते.

आणि LAN वरील संगणकांनी सामान्यतः एकच कनेक्शन सामायिक केल्यामुळे, इंटरनेट हे CSMA/CD च्या प्रिन्सिपलभोवती तयार केले गेले होते, किंवा वाहक-सेन्स मल्टिपल ऍक्सेससह टक्कर सर्च जे. मुळात, प्रोटोकॉल हे सुनिश्चित करते की कोणतीही फ्रेम बाहेर पाठवण्यापूर्वी लाइन वापरात नाही. आज, ते पूर्वीपेक्षा खूपच कमी महत्त्वाचे आहे

नेटवर्किंगचे सुरुवातीचे दिवस कारण डिव्हाइसेसना सामान्यतः स्विक किंवा नोडद्वारे नेटवर्कशी त्यांचे स्वतःचे प्राइवेट कनेक्शन असते. आणि इंटरनेट आता पूर्ण डुप्लेक्स वापरून कार्यरत असल्यामुळे, पाठवणे आणि प्राप्त

करणारे चॅनेल देखील पूर्णपणे वेगळे आहेत, त्यामुळे त्यांच्या प्रवासाच्या त्या टप्प्यावर टक्कर प्रत्यक्षात येऊ शकत नाही.

टक्कर परिस्थितीचा सामना करताना, इंटरनेटमध्ये कोणतीही त्रुटी दुरुस्त केली जात नाही, म्हणून सर्व काही अचूकपणे प्रसारित केले जात आहे याची खात्री करण्यासाठी संप्रेषणांना एडवॉन्स प्रोटोकॉलवर अवलंबून राहणे आवश्यक आहे. तथापि, इंटरनेट अजूनही बऱ्याच इंटरनेट आणि डिजिटल संप्रेषणांसाठी आधार प्रदान करते, आणि बऱ्याच उच्च-स्तरीय प्रोटोकॉलसह सहजपणे समाकलित होते, त्यामुळे आजकाल ही समस्या जवळजवळ कधीच नाही.

## नेटवर्क कसे तयार केले जाते?

नोड्स आणि कनेक्शन (लिंक) वापरून संगणक नेटवर्क तयार केले जातात जे संधाना एकमेकांशी संवाद साधण्यास सक्षम करतात. प्रत्येक नोड नेटवर्क इंटरफेस कार्ड वापरतो जे नेटवर्कशी कसे संवाद साधते ते परिभाषित करते. प्रत्येक नेटवर्क कार्डचा स्वतःचा अंकीय एड्रेस असतो.

## एन्ड -युजर उपकरणे आणि लोकल नेटवर्क (End-User Devices and local networks)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कम्प्युनिकेशन नेटवर्क परिभाषित करा
- संगणक नेटवर्कमधील नोड उपकरणांचे टाइप
- एंड डिव्हाइसेसची फंक्शन.

लोक ज्या नेटवर्क उपकरणांशी सर्वात परिचित आहेत त्यांना एंड डिव्हाइसेस म्हणतात. नेटवर्कशी कनेक्ट केलेले सर्व संगणक जे नेटवर्क कम्प्युनिकेशनमध्ये थेट पार्ट घेतात ते होस्ट म्हणून वर्गीकृत केले जातात. ही उपकरणे वापरकर्ते आणि बेसिक कम्प्युनिकेशन नेटवर्क यांच्यातील इंटरफेस तयार करतात.

एन्ड युजर नोड

डेस्कटॉप संगणक, लॅपटॉप, प्रिंटर, टॅब्लेट आणि सर्व्हरसह अंतिम युजर उपकरणांची अनेक उदाहरणे आहेत. या एन्ड युजर उपकरणांसाठी सर्वात सामान्य संज्ञा म्हणजे होस्ट, आणि एंड यूजर नोड; आम्ही या संपूर्ण प्रकरणांमध्ये एंड यूजर नोड वापरू.

संगणक नेटवर्कमधील नोड उपकरणांचे टाइप :

एंड डिव्हाइसेस आणि मध्यस्थ (Intermediary) साधने

आम्हाला माहित आहे की संगणक नेटवर्क ही एक संज्ञा आहे जी सामान्यतः इनफार्मेशन च्या देवाणघेवाणीसाठी वापरली जाते. आणि संसाधन सामायिकरण. संगणक नेटवर्कमध्ये, एक नोड आहे

डेटा पाठवण्यास किंवा प्राप्त करण्यास सक्षम असलेले कोणतेही सबकरण, यांना

आणि इतर नोड्समधून निश्चित आणि इच्छित प्रवाह दर सुरक्षितपणे आणि विश्वासार्हपणे.

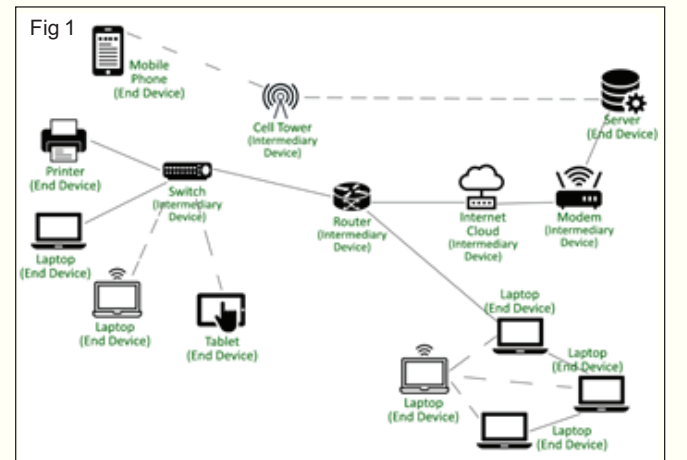
कार्यक्षमता आणि वापराच्या आधारावर, नोड उपकरणांचे विस्तृतपणे खालील प्रकारांमध्ये वर्गीकरण केले जाऊ शकते -

1 एन्ड उपकरणे

2 मध्यस्थ (Intermediary) साधने

एन्ड डिव्हाइसेस

एंड डिव्हाइसेस ही नोड उपकरणे आहेत जी संगणक नेटवर्कवर होणार्या कम्प्युनिकेशन मध्ये सोर्स पॉइंट किंवा डेस्टिनेशन पॉइंट म्हणून काम करतात. संगणक नेटवर्किंग सिस्टीममध्ये येणाऱ्या प्रगतीसह, आमच्याकडे नोड्स आहेत जे क्लायंट, सर्व्हर किंवा दोन्ही म्हणून काम करू शकतात. उर्वरित नेटवर्क त्यांच्या दरम्यान कम्प्युनिकेशन दुवे इंस्टाल करण्यासाठी या अंतिम उपकरणांभोवती तयार केले आहे. नोड डिव्हाइसेसवर इंस्टाल केलेले सॉफ्टवेअर संगणक नेटवर्कमध्ये प्ले करण्यासाठी ऑफर केलेली भूमिका निर्धारित करते. त्यांच्या वापरानुसार, शेवटची उपकरणे मोठ्या प्रमाणात खालील रेन्ज मध्ये विपार्ट ली जाऊ शकतात - (पिक्चर 1)



- **क्लायंट म्हणून एन्ड उपकरणे** - डेटाची विनंती करणे, प्राप्त केलेला डेटा प्रदर्शित करणे इत्यादी फंक्शन पूर्ण करा. सामान्यतः अंतिम क्लायंटच्या वापरासाठी.
- **सर्व्हर म्हणून एन्ड डिव्हाइसेस** - नेटवर्कवरील इतर नोड्स (किंवा होस्ट) यांना वेबपेजेस किंवा ई-मेल यांसारखी इनफार्मेशन आणि सर्विस प्रदान करणाऱ्या प्रोग्रामसह सुसज्ज.

### एंड डिव्हाइसेसची फंक्शन

- 1 ते नेटवर्कमधून वाहणाऱ्या डेटा किंवा इनफार्मेशन चे प्रवर्तक म्हणून काम करतात.
- 2 एन्ड यूजर (humans) आणि अनेक नोड उपकरणे असलेले कम्युनिकेशन नेटवर्क यांच्यातील इंटरफेस म्हणून कार्य करतात.

### एंड डिव्हाइसेसची उदाहरणे

वर्क स्टेशन, लॅपटॉप, डेस्कटॉप कॉम्प्युटर, प्रिंटर आणि स्कॅनर, सर्व्हर (फाइल सर्व्हर, वेब सर्व्हर), मोबाईल फोन, टॅब्लेट, स्मार्ट फोन इ.

### मध्यस्थ (Intermediary) डिव्हाइसेस

मध्यस्थ डिव्हाइसेस ही नोड उपकरणे आहेत जी संगणक नेटवर्कमध्ये एका बाजूकडून दुसऱ्या बाजूला डेटा फॉरवर्ड करण्यासाठी डिझाइन केलेली आहेत. ही मध्यस्थ डिव्हाइसेस इतर नोड्स आणि संपूर्ण संगणक नेटवर्कवर डेटा इच्छित प्रवाह दराने प्रभावीपणे प्रवाहित होतो याची खात्री करून बॅकग्राउंड त फंक्शन हाताळतात.

त्यांच्याद्वारे प्रवाहित होणाऱ्या डेटाच्या इन्स्टॉलेशन साठी मध्यस्थ डिव्हाइसेस नेटवर्क इंटरकनेक्शनसच्या इनफार्मेशन सह IP एड्रेस, MAC एड्रेस आणि पोर्ट क्रमांक (किंवा पोर्ट एड्रेस) यासारख्या विविध ॲट्रिब्युट्स सिस्टमचा वापर करतात. संगणक नेटवर्कमधील पुढील विविध टाइप चे स्विचिंग संदेश कम्युनिकेशन दरम्यान नेटवर्कद्वारे कोणता पाथ घेतात हे निर्धारित करतात.

### मध्यस्थ डिव्हाइसेस ची फंक्शन

- सिग्नल डॅम्पिंग ही एक सामान्य इवेंट्स आहे जी या उपकरणांद्वारे केलेल्या डेटा सिग्नलच्या पुनरुत्पादन (रिजनरेशन) आणि पुनर्प्रसारणाद्वारे (रिट्रांसमिशन) मात केली जाते.
- यशस्वी हस्तांतरण सुनिश्चित करण्यासाठी ही उपकरणे वापरल्या जात असलेल्या स्विचिंग तंत्राच्या आधारावर सोर्स एड्रेस, डेस्टिनेशन एड्रेस किंवा नेटवर्कद्वारे अस्तित्वात असलेल्या भिन्न मार्गांची इनफार्मेशन रेकॉर्ड ठेवतात.
- ते रिडंडंसी बिट्स इत्यादींचा प्रभावीपणे वापर करून दोष आणि त्रुटी सर्च तात आणि संगणक नेटवर्कमध्ये डेटा हस्तांतरित करताना दुरुस्त्या करून दोष सहिष्णुता सुनिश्चित करण्यासाठी उपकरणांना सूचित करतात.
- कम्युनिकेशन च्या संपूर्ण कोर्स दरम्यान निश्चित इच्छित प्रवाह कंट्रोल आणि प्रतिसाद कालएक्सटर्नल ता राखणे.
- या उपकरणांद्वारे संदेशांचे (किंवा पॅकेट्स) त्यांच्या प्राधान्यक्रमानुसार सेटअप, ग्रुप बद्धीकरण आणि दिशा दिली जाते.

- सुरक्षा सेटिंग्जवर आधारित, डेटाच्या प्रवाहाला परवानगी देऊन किंवा नाकारून नेटवर्क सुरक्षा प्रदान करा.

### मध्यस्थ उपकरणांची उदाहरणे

हब, स्विचेस, वायरलेस ऍक्सेस पॉइंट्स आणि नेटवर्कमध्ये एक्सेस करण्यासाठी वापरलेली इतर उपकरणे, फाइल सर्व्हर, वेब सर्व्हर, प्रिंट सर्व्हर, मॉडेम, इंटरनेटवर्किंगसाठी वापरणारी उपकरणे जसे की राउटर, ब्रिज, रिपीटर्स आणि सुरक्षा फायरवॉल इ.

### लोकल नेटवर्क

आकार, जटिलता, सुरक्षिततेची पातळी किंवा भौगोलिक रेन्ज यावर अवलंबून नेटवर्कचे वर्गीकरण केले जाऊ शकते. आम्ही भौगोलिक प्रसारावर आधारित काही लोकप्रिय टोपोलॉजीजवर चर्चा करू.

### पॅन (पिक्चर 2)

PAN हे पर्सनल एरिया नेटवर्कचे संक्षिप्त रूप आहे. PAN म्हणजे एखाद्या व्यक्तीच्या प्राइवेट जागेच्या मर्यादित, स्पेशल तः 10 मीटरच्या मर्यादित उपकरणांमधील परस्पर संबंध. तुम्ही तुमच्या लॅपटॉपवरून किंवा मोबाईलवरून इमेज किंवा गाणी ट्रान्सफर केली असल्यास ब्लूटूथ वापरून तुमच्या मित्राच्या मोबाईलवर मोबाइल, तुम्ही वैयक्तिक एरिया नेटवर्क सेट केले आहे आणि वापरले आहे.



एखादी व्यक्ती तिचा लॅपटॉप, स्मार्ट फोन, वैयक्तिक डिजिटल असिस्टंट आणि पोर्टेबल प्रिंटर घरबसल्या नेटवर्कमध्ये कनेक्ट करू शकते. हे नेटवर्क पूर्णपणे वाय-फाय किंवा वायर्ड आणि वायरलेसचे संयोजन असू शकते.

### LAN (पिक्चर 3)

LAN किंवा लोकल एरिया नेटवर्क हे ऑफिस, बिल्डिंग किंवा मॅन्युफॅक्चरिंग युनिटसारख्या एकाच साइटवर पसरलेले वायर्ड नेटवर्क आहे. जेव्हा कार्यसंघ सदस्यांना सॉफ्टवेअर आणि हार्डवेअर रिसोर्सेस एकमेकांशी सामायिक करण्याची आवश्यकता असते तेव्हा LAN सेट केले जाते परंतु एक्सटर्नल जगाशी नाही. ठराविक सॉफ्टवेअर संसाधनांमध्ये अधिकृत डॉक्युमेंट्स, युजर पुस्तिका, कर्मचारी हँडबुक इत्यादींचा समावेश होतो. नेटवर्कवर सहजपणे शेअर करता येणारी हार्डवेअर रिसोर्सेस प्रिंटर, फॅक्स मशीन, मोडेम, मेमरी स्पेस इ. यांचा समावेश होतो. यामुळे संस्थेसाठी पायाभूत सुविधांच्या खर्चात मोठी घट होते.

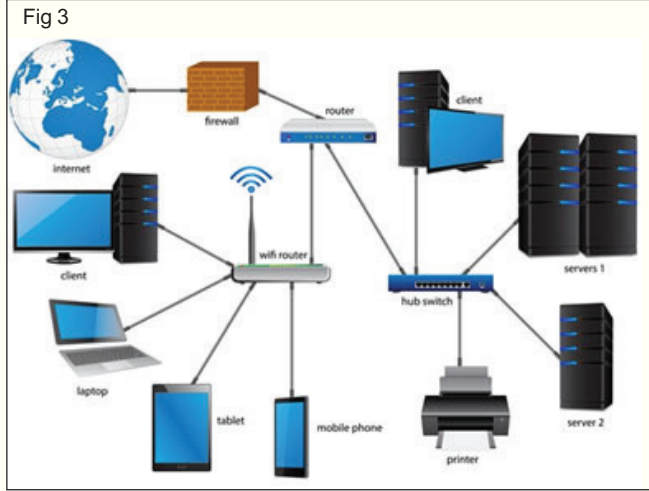
वायर्ड किंवा वायरलेस कनेक्शन वापरून LAN सेट केले जाऊ शकते. पूर्णपणे वायरलेस असलेल्या लॅनला वायरलेस लॅन किंवा डब्ल्यूएलएएन म्हणतात.



## MAN (पिक्चर 4)

MAN हे मेट्रोपॉलिटन एरिया नेटवर्कचे संक्षिप्त रूप आहे. हे शहर, कॉलेज कॅम्पस किंवा लहान प्रदेशात पसरलेले नेटवर्क आहे. MAN LAN पेक्षा मोठा आहे आणि सामान्यतः अनेक किलोमीटरवर पसरलेला आहे. MAN चे उद्दिष्ट हार्डवेअर आणि सॉफ्टवेअर रिसोर्सेस सामायिक करणे आहे, ज्यामुळे पायाभूत सुविधांच्या खर्चात घट होते. MAN अनेक LAN जोडून तयार केले जाऊ शकते.

MAN चे सर्वात सामान्य उदाहरण म्हणजे केबल टीव्ही नेटवर्क.



## WAN (पिक्चर 5)

WAN किंवा वाइड एरिया नेटवर्क एका देशात किंवा अनेक देशांमध्ये पसरलेले आहे. WAN हे सहसा अनेक LAN, MANs आणि WAN चे नेटवर्क असते. उपलब्धता आणि विश्वासार्हतेवर अवलंबून, वायर्ड किंवा वायरलेस कनेक्शन वापरून नेटवर्क सेट केले जाते.

WAN चे सर्वात सामान्य उदाहरण म्हणजे इंटरनेट.



## नेटवर्क कम्युनिकेशन

- नेटवर्क कम्युनिकेशन ची बेसिक इनफार्मेशन
- होम नेटवर्क परिभाषित करा
- वाय-फाय परिभाषित करा
- LAN चा परिचय

## नेटवर्क कम्युनिकेशनची बेसिक इनफार्मेशन

हा विपार्ट नेटवर्क कम्युनिकेशनला खूप उच्च लेयर वरून पाहतो आणि संपूर्ण इनफार्मेशन मध्ये वापरल्या जाणाऱ्या काही संज्ञा परिभाषित करतो. z/OS® नेटवर्क कम्युनिकेशन आणि TCP/IP सॉफ्टवेअर अधिक तपशीलवार इनफार्मेशन साठी, z/OS कम्युनिकेशन सर्व्हर: IP कॉन्फिगरेशन पाथ दर्शक आणि z/OS कम्युनिकेशन सर्व्हर: IP प्रोग्रामरचे पाथ दर्शक आणि रेफरेन्स पहा. IPv6 नेटवर्क कम्युनिकेशन आणि AF\_INET6 सॉफ्टवेअर अधिक तपशीलवार इनफार्मेशन साठी, z/OS कम्युनिकेशन सर्व्हर पहा: IPv6 नेटवर्क आणि ॲपल डिझाइन पाथ दर्शक. नेटवर्क कम्युनिकेशन, किंवा इंटरनेटवर्किंग, प्रोटोकॉलचा एक संच (म्हणजे नियम आणि मानके) परिभाषित करते जे एप्लिकेशन प्रोग्रामना ते जिथे चालवले जातात त्या हार्डवेअर आणि ऑपरेटिंग सिस्टमचा विचार न करता एकमेकांशी बोलू देतात. इंटरनेटवर्किंग ॲप्लिकेशन प्रोग्रामना त्यांच्या फिजिकल नेटवर्क कनेक्शनपासून स्वतंत्रपणे संवाद साधण्याची परवानगी देते.

TCP/IP नावाच्या इंटरनेटवर्किंग तंत्रज्ञानाचे नाव त्याच्या दोन मुख्य प्रोटोकॉलवर ठेवले आहे: ट्रान्समिशन कंट्रोल प्रोटोकॉल (TCP) आणि इंटरनेट प्रोटोकॉल (IP). TCP/IP समजून घेण्यासाठी, तुम्हाला खालील अटींशी परिचित असले पाहिजे:

### क्लायंट

नेटवर्कवरील सर्विस ची विनंती करणारी प्रक्रिया.

### सर्व्हर

क्लायंटच्या सेवेच्या विनंतीला प्रतिसाद देणारी प्रक्रिया.

### डेटाग्राम

इनफार्मेशन चे बेसिक एकक, ज्यामध्ये एक किंवा अधिक डेटा पॅकेट असतात, जे वाहतूक लेयर वर इंटरनेटवरून दिले जातात.

### पॅकेट

संगणक आणि त्याच्या नेटवर्कमधील डेटा ट्रांसमिशन चे युनिट किंवा ब्लॉक. पॅकेटमध्ये सहसा नेटवर्क हेडर, किमान एक उच्च-स्तरीय प्रोटोकॉल हेडर आणि डेटा ब्लॉक असतात. सामान्यतः, डेटा ब्लॉक्सचे फॉरमॅट पॅकेट कसे हाताळले जातात यावर रिझल्ट करत नाही. नेटवर्कद्वारे डेटा पाठवण्यासाठी इंटरनेटवर्क लेयर वर पॅकेट्स हे एक्सचेंज माध्यम आहे.

इंटरनेटवरील उपकरणे एकमेकांशी कशी संवाद साधतात?

इंटरनेट हा एक स्टॅण्डर्ड प्रोटोकॉल आहे जो कितीही संगणकांना एकमेकांशी संवाद साधण्याची परवानगी देतो. ही यंत्रे इंटरनेट केबल किंवा "रेन्ज 5" वायरिंगद्वारे एकमेकांशी जोडली जाणे आवश्यक आहे.

## होम नेटवर्किंग म्हणजे काय?

होम नेटवर्क हे उपकरणांचा एक समूह आहे - जसे की संगणक, गेम सिस्टम, प्रिंटर आणि मोबाइल उपकरणे - जे इंटरनेट आणि एकमेकांना जोडतात. होम नेटवर्क दोन प्रकारे जोडतात:

- वायर्ड नेटवर्क, जे प्रिंटर आणि स्कॅनर सारख्या उपकरणांना केबलसह जोडते
- एक वायरलेस नेटवर्क, जे टॅब्लेट आणि ई-रीडर्स सारख्या उपकरणांना केबलशिवाय जोडते

## होम नेटवर्क का सेट अप करावे?

होम नेटवर्क इंस्टाल करण्याची अनेक कारणे आहेत. होम नेटवर्किंग तुम्हाला करू देते अशा काही गोष्टी येथे आहेत:

- मल्टिपल संगणक, गेम सिस्टम, मोबाइल डिव्हाइस आणि बरेच काही वरून इंटरनेटशी कनेक्ट करा.
- नेटवर्कशी कनेक्ट केलेल्या सर्व डिव्हाइसेसवर फायली आणि फोल्डरमध्ये एक्सेस करा.
- एकाच प्रिंटरवर अनेक संगणकांवरून प्रिंटेड करा.
- एकाच ठिकाणी सर्व नेटवर्क उपकरणांसाठी सुरक्षा सेटिंग्ज मॅनेज करा.

तुम्ही होम नेटवर्किंग वापरून पाहण्यास तयार असल्यास, तुम्हाला स्टार्ट करण्यासाठी काय आवश्यक आहे हे सर्च प्यासाठी आणि तुमच्या नेटवर्कसाठी सबलब्ध पर्याय पाहण्यासाठी खाली वाचा.

## होम नेटवर्क सेट करण्यासाठी तुम्हाला काय हवे आहे

होम नेटवर्किंग सेट करण्यासाठी, तुम्हाला खालील गोष्टींची आवश्यकता असेल:

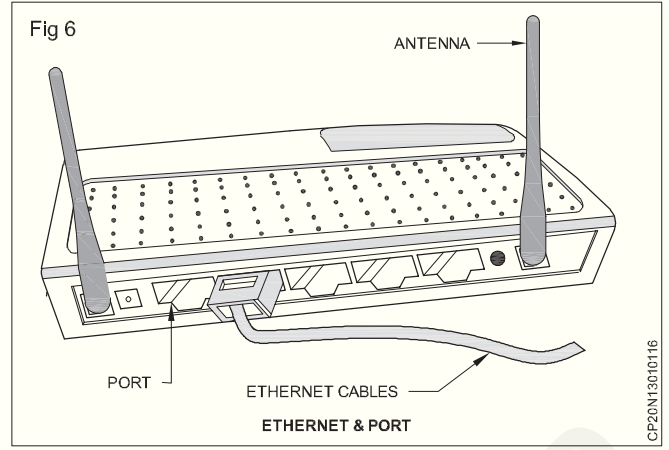
- Xfinity इंटरनेट सर्विस सबस्क्रिप्शन (किंवा दुसऱ्या इंटरनेट प्रदात्याची सदस्यता)
- एक मॉडेम, जो इंटरनेटशी कनेक्ट होतो आणि एक राउटर, जो तुमच्या मॉडेमद्वारे तुमची डिव्हाइस एकमेकांना आणि इंटरनेटशी जोडतो (किंवा गेटवे, जो मोडेम आणि राउटर दोन्ही म्हणून कार्य करतो)

- नेटवर्कशी जोडण्यासाठी संगणक किंवा इतर सबकरण अधिक इनफार्मेशन साठी वायरलेस गेटवे ऍक्टिव्ह करा पहा.

वायरलेस गेटवे 1 (मॉडेल क्रमांक TG852G, TG862G, SMCD3GNV, TC8305C) आणि वायरलेस गेटवे 2 (मॉडेल क्रमांक DPC3939) ऑल-इन वन मोडेम, राउटर आणि फोन सबकरण म्हणून कार्य करतात. ते वापरकर्त्यांना होम नेटवर्कसाठी सबलब्ध सर्वोत्तम सुरक्षा सेटिंग्ज ऑटोमॅटिक पणे प्रदान करतात. पहा वायरलेस गेटवे म्हणजे काय? अधिक इनफार्मेशन साठी.

**वायरलेस होम नेटवर्क:** एक वायरलेस नेटवर्क, ज्याला सहसा वाय-फाय म्हणतात, केबलस न वापरता डिव्हाइसेसना एकमेकांशी आणि इंटरनेटशी कनेक्ट करते. वायफाय म्हणजे काय आणि ते कसे मिळवायचे ते पहा? अधिक इनफार्मेशन साठी.

**वायर्ड होम नेटवर्क (पिक्चर 6):** वायर्ड होम नेटवर्क इथरनेट केबल्स वापरून उपकरणे एकमेकांना आणि इंटरनेटशी जोडते.



वायर्ड होम नेटवर्क असण्याचे अनेक फायदे आहेत:

- इंटरनेटशी जलद आणि अधिक विश्वासार्ह कनेक्शन
- वाढलेली सुरक्षितता, कारण कोणतेही बाहेरचे वापरकर्ते तुमच्या नेटवर्कमध्ये एक्सेस करू शकत नाहीत
- वायरलेसपेक्षा सोपे सेटअप आणि समस्यानिवारण कनेक्शन

**मिक्सड होम नेटवर्क:** अनेकांना असे आढळते की वायरलेस आणि वायर्ड नेटवर्किंगचे मिश्रण त्यांच्या गरजा उत्तम प्रकारे पूर्ण करते. उदाहरणार्थ, पिक्चर पट प्रवाहित करणारी उपकरणे जलद आणि अधिक स्थिर वायर्ड कनेक्शनचा फायदा घेतात. लॅपटॉप किंवा टॅब्लेट सारख्या उपकरणांना, तथापि, वायरलेस कनेक्शनसह सबलब्ध असलेल्या गतिशीलतेचा फायदा होतो.

वायरलेस गेटवे 1 आणि वायरलेस गेटवे 2 दोन्ही वायरलेस क्षमता आणि चार इथरनेट पोर्टसह येतात, ज्यामुळे तुम्हाला एकाच वेळी केबलसह आणि त्याशिवाय डिव्हाइसेस कनेक्ट करता येतात.

## होम नेटवर्किंग शब्दावली(Glossary)

खाली सामान्य होम नेटवर्किंग संज्ञांची लिस्ट आहे. या व्याख्यांशी परिचित होण्यासाठी थोडा वेळ घ्या, कारण त्यांचा उल्लेख इतर मदत लेखांमध्ये केला जाईल.

**वायरलेस नेटवर्क** -उपकरणांचा समूह (संगणक, गेम सिस्टम, ई-रीडर, इ.) एकमेकांशी आणि केबलशिवाय इंटरनेटशी कनेक्ट केलेले, ज्याला वाय-फाय देखील म्हणतात.

**मोडेम** -एक सबकरण जे केबलद्वारे राउटर किंवा संगणकांना इंटरनेटशी जोडते.

**राउटर** -एक सबकरण जे वायरलेस सिग्नल प्रसारित करते आणि मॉडेमद्वारे डिव्हाइसेसना एकमेकांशी आणि इंटरनेटशी कनेक्ट करते.

**गेटवे** -वायरलेस गेटवे 1 किंवा वायरलेस गेटवे 2 सारखे मॉडेम आणि राउटर दोन्ही म्हणून कार्य करणारे सबकरण.

**सिग्नल रेंज** -ज्या भागात वायरलेस-सक्षम डिव्हाइस वायरलेस सिग्नल उचलू शकते आणि वायरलेस नेटवर्कशी कनेक्ट करू शकते.

**Administration साइट** - वायरलेस गेटवे 1 आणि वायरलेस गेटवे 2 साठी अॅडमिन टूल नावाची राउटर आणि वायरलेस नेटवर्कची सेटिंग्ज पाहण्यासाठी आणि बदलण्यासाठी वापरण्यात येणारी ऑनलाइन साइट.

**राउटर युजर नाव आणि पासवर्ड** - एडमिनिस्ट्रेशन साइटवर एक्सेस करण्यासाठी वापरण्यात येणारी लॉगिन इनफार्मेशन .

**नेटवर्क नाव (SSID)** - वायरलेस नेटवर्कचे नाव.

**नेटवर्क पासवर्ड (की)** - वायरलेस नेटवर्कशी कनेक्ट करण्यासाठी वापरला जाणारा पासवर्ड.

### **Wi- वाय-फाय व्याख्या**

वाय-फाय हे एक वायरलेस तंत्रज्ञान आहे जे संगणक, टॅब्लेट, स्मार्टफोन आणि इतर उपकरणांना इंटरनेटशी जोडण्यासाठी वापरले जाते.

वाय-फाय हा वायरलेस राउटरवरून जवळच्या इन्फ्रारेड वर पाठवलेला रेडिओ सिग्नल आहे, जो सिग्नलला तुम्ही पाहू आणि वापरू शकता अशा डेटामध्ये अनुवादित करतो. डिव्हाइस राउटरवर रेडिओ सिग्नल पाठवते, जे वायर किंवा केबलद्वारे इंटरनेटशी कनेक्ट होते.

### **वाय-फाय नेटवर्क म्हणजे काय?**

वाय-फाय नेटवर्क हे फक्त एक इंटरनेट कनेक्शन आहे जे वायरलेस राउटरद्वारे घर किंवा व्यवसायातील मल्टिपल डिव्हाइसेससह सामायिक केले जाते. राउटर थेट तुमच्याशी कनेक्ट केलेले आहे

इंटरनेट मॉडेम आणि तुमच्या सर्व वाय-फाय सक्षम उपकरणांवर इंटरनेट सिग्नल प्रसारित करण्यासाठी हब म्हणून कार्य करते. हे तुम्हाला तुमच्या नेटवर्क कवरेज क्षेत्रामध्ये असेपर्यंत इंटरनेटशी कनेक्ट राहण्याची लवचिकता देते.

### **वाय-फाय म्हणजे काय?**

हा शब्द मार्केटिंग फर्मने तयार केला आहे कारण वायरलेस उद्योग IEEE 802.11 म्हणून ओळखल्या जाणाऱ्या काही युजर -अनुकूल तंत्रज्ञानाचा रेफरन्स देण्यासाठी युजर -अनुकूल नाव सर्च त होता आणि नाव अडकले. वाय-फाय, ज्याला वाय-फाय, वाय-फाय, वाय-फाय किंवा वाय-फाय असे संबोधले जाते, ते वायरलेस फिडेलिटीसाठी लहान मानले जाते आणि मार्केटिंग फर्मसाठी पैसे देणाऱ्या संस्थेला कधीकधी वायरलेस फिडेलिटी अलायन्स इंक म्हणून संबोधले जाते.

### **वाय-फाय कसे कार्य करते?**

तुमच्या वायरलेस राउटरवरून तुमच्या TV, स्मार्टफोन, टॅब्लेट आणि संगणकासारख्या वाय-फाय सक्षम डिव्हाइसवर डेटा प्रसारित करण्यासाठी वाय-फाय रेडिओ लहरी वापरते. ते एअरवेव्हद्वारे एकमेकांशी संवाद साधत असल्यामुळे, तुमची उपकरणे आणि वैयक्तिक इनफार्मेशन हॅकर्स, सायबर हल्ले आणि इतर धोक्यांसाठी असुरक्षित होऊ शकतात. जेव्हा तुम्ही कॉफी शॉप किंवा विमानतळासारख्या ठिकाणी सार्वजनिक वाय-फाय नेटवर्कशी कनेक्ट करता तेव्हा हे स्पेशल तः खरे आहे. जेव्हा शक्य असेल तेव्हा, पासवर्ड-संरक्षित किंवा वैयक्तिक हॉटस्पॉट असलेल्या वायरलेस नेटवर्कशी कनेक्ट करणे सर्वोत्तम आहे.

### **वाय-फाय कनेक्शनचे टाइप**

घरामध्ये वायरलेस मेथड्स ने कनेक्ट होण्याचे तुमचे पर्याय वाढत आहेत कारण मोबाइल नेटवर्क होम इंटरनेट क्षेत्रात एक्सपान्शन त आहेत.

इंटरनेट सेवेप्रमाणे, प्रत्येक टाइप च्या वायरलेस कनेक्शनचे फायदे आणि तोटे आहेत, जसे की वेग आणि सिग्नलची ताकद. त्यापैकी काही आम्ही येथे मांडल्या आहेत.

### **वायरलाइन/राउटर**

आजकाल बहुतेक घरे इंटरनेट वापरण्यासाठी वायरलेस राउटर वापरतात. साधकामध्ये सेटअपची सोय, वाय-फाय ऍक्सेस पॉईंट (राउटर) च्या मर्यादित गतिशीलता आणि मल्टिपल उपकरणे कनेक्ट करण्याची क्षमता समाविष्ट आहे. बाधकः मर्यादित बँडविड्थ आणि अधिक उपकरणे समान वाय-फाय नेटवर्कशी जोडलेली असल्यामुळे कमी गती, तसेच घरातील इतर इलेक्ट्रोमॅग्नेटिक उपकरणांचा संभाव्य हस्तक्षेप.

### **मोबाइल हॉटस्पॉट किंवा जेटपॅक**

मोबाइल आणि समर्पित हॉटस्पॉट हे जाता जाता सुरक्षितपणे कनेक्ट होण्याचा अधिक लोकप्रिय पाथ बनत आहेत. तुमचा स्मार्टफोन आणि एक जेटपॅक ही दोन सामान्य हॉटस्पॉट उपकरणे आहेत. आज बहुतेक कोणताही स्मार्टफोन किंवा टॅब्लेट तात्पुरते हॉटस्पॉट म्हणून वापरला जाऊ शकतो आणि तुम्हाला अधूनमधून गरज पडल्यास हा एक उत्तम पर्याय आहे. हे वापरण्यास सोपे आहे आणि त्यासाठी अतिरिक्त उपकरणे खरेदी करण्याची आवश्यकता नाही, परंतु ते तुमच्या बॅटरीचे आयुष्य आणि डेटा खूप जलद झेप करू शकते. दुसरीकडे, जेटपॅक एक समर्पित मोबाइल हॉटस्पॉट म्हणून काम करतो जो तुमच्या स्मार्टफोनप्रमाणेच तुमच्या क्षेत्रातील सेल टॉवरमधून सिग्नल घेतो. अधिक उपकरणे त्यास कनेक्ट करू शकतात आणि ते वाय-फाय ची मोठी रेन्ज देते. आणि ते एक वेगळे सबकरण असल्यामुळे तुमच्या स्मार्टफोनची बॅटरी शक्ती स्पर्श करत नाही. पतन म्हणजे जेटपॅक आणि वेगळा प्लॅन खरेदी करावा लागत आहे.

### **LTE होम इंटरनेट**

जर तुम्ही ग्रामीण भागात राहता जेथे तुमचे इंटरनेट पर्याय मर्यादित आहेत, तर 4G LTE होम इंटरनेट विचारात घेण्यासारखे आहे. हे सेल फोन टॉवर्स आणि मोबाइल नेटवर्कवर 25 Mbps च्या सरासरी डाउनलोड गतीसह उच्च गतीची इंटरनेट सर्विस देते. सबग्रहापेक्षा LTE चे फायदे म्हणजे तुमच्या वाहकावर अवलंबून चांगला वेग आणि विश्वासार्हता.

**5G होम इंटरनेट:** 5G होम इंटरनेट (फिक्स्ड वायरलेस ऍक्सेस) आता देशभरात अधिक ठिकाणी आहे.\* मल्टिपल डिव्हाइस कनेक्ट केलेले असले तरीही, ते आपल्या संपूर्ण घराला उर्जा देण्यासाठी पुरेसे विश्वसनीय आणि जलद आहे. फिक्स्ड वायरलेस ऍक्सेसमध्ये एक साधा प्लग अँड प्ले सेटअप आहे, याचा अर्थ कोणत्याही गोंधळलेल्या वायर्स नाहीत आणि सेटअपसाठी तुमच्या घरी तंत्रज्ञ येण्याची वाट पाहण्याची गरज नाही.

### **घरी वाय-फाय कसे मिळवायचे**

नमूद केल्याप्रमाणे, तुमच्या घरी वायरलेस सर्विस मिळवण्याचे अनेक पाथ आहेत आणि बहुतेक ते भौगोलिक पोजिशन आणि सबलब्धतेवर अवलंबून असतात. (तुमच्या परिसरात कोणते वाय-फाय होम इंटरनेट सबलब्ध आहे ते पहा.) बहुतांश शहरी आणि सबनगरी भागात 5G होम इंटरनेट अगदी कोपन्यात यापैकी बहुतांश सर्विस देतात. ग्रामीण पार्ट बहुधा सबग्रह आणि 4G LTE होम इंटरनेट ऑफर करेल. तुमच्याकडे वायर्ड इंटरनेट सर्विस असल्यास, तुम्ही घरी तुमचे स्वतःचे वाय-फाय नेटवर्क सेट करू शकाल. राउटरला तुमच्या मॉडेमशी कनेक्ट करून, तुम्ही तुमचे इंटरनेट कनेक्शन

तुमच्या सर्व वाय-फाय सक्षम डिव्हाइसेससह शेअर करू शकता. तुमच्या घरामध्ये दोन स्तर, कॉन्क्रीटच्या भिंती किंवा यादृच्छिक डेड झोन असल्यास, या भागात वायरलेस सिग्नल रिले करणारा वाय-फाय एक्सपान्शन क जोडल्यास मोठा फरक पडू शकतो.

लक्षात ठेवा की जसजशी तुमच्या मोबाईल उपकरणांची नंबर वाढते, तशीच बँडविड्थची मागणीही वाढते. तुमची डिव्हाइस टॉप स्पीडवर चालू ठेवण्यासाठी, तुम्हाला तुमच्या इंटरनेट स्पीड प्लॅन अपग्रेड करण्याची आवश्यकता असू शकते. Verizon तुमच्या स्थानावर अवलंबून, DSL आणि फायबरपासून ते 5G होम इंटरनेटपर्यंत अनेक इंटरनेट सर्विस देते. चांगल्या विश्वासाहतेसाठी आणि Verizon च्या जलद डाउनलोड गतीसाठी, Fios इंटरनेट किंवा 5G होम इंटरनेट तुमच्या पत्त्यावर सबलब्ध आहे का ते तपासा.

तुमच्या परिसरात कोणते वाय-फाय इंटरनेट सबलब्ध आहे ते पहा.

### LAN चा परिचय

लोकल एरिया नेटवर्क (LAN) मध्ये कॉम्प्युटरची सीरीज असते जी एकमेकांशी जोडलेली असते आणि एका परिमित ठिकाणी नेटवर्क बनवते. LAN मधील संगणक TCP/IP इन्टरनेट किंवा वाय-फाय द्वारे एकमेकांना जोडतात. LAN साधारणपणे शाळा, कार्यालय, असोसिएशन किंवा चर्च यांसारख्या संस्थेसाठीच असतो.

### LAN डिव्हाइस म्हणजे काय?

लोकल एरिया नेटवर्क (LAN) हा संगणक आणि पेरीफेरल उपकरणांचा समूह आहे जो एका स्पेसिफिक भौगोलिक क्षेत्रामध्ये सर्व्हरशी एक सामान्य कम्प्युनिकेशन लाइन किंवा वायरलेस लिंक सामायिक करतो. लोकल एरिया

नेटवर्क होम ऑफिसमध्ये दोन किंवा तीन वापरकर्ते किंवा कॉर्पोरेशनच्या सेंट्रल ऑफिसमध्ये हजारो वापरकर्ते देऊ शकतात.

### नेटवर्क उपकरणांचे टाइप

- हब.
- स्विच.
- राउटर.
- पूल.
- गेटवे.
- मोडेम.
- रिपीटर.
- एक्सेस पॉइंट .

### लोकल एरिया नेटवर्कचे टाइप (LAN)

- पीअर टू पीअर (P2P) LAN. ...
- टोकन रिंग LAN. ...
- टोकन बस LAN. ...
- वायर्ड LAN. ...
- क्लाउड-मॅनेज LAN. ...
- सार्वजनिक इंटरनेट. ...
- वायर्ड एंड-यूजर डिव्हाइसेस. ...
- मोबाइल एंड-यूजर डिव्हाइसेस.

### LAN वि. MAN वि. WAN (पिक्चर 7)

	LAN	MAN	WAN
GEOGRAPHICAL AREA	Small area, 1 to 5 km	City limits, 50 to 60 km	Global, up to 1000s of km
TYPICAL CUSTOMERS	Schools and colleges, offices, small industries and SMBs	Mid-market to large enterprises, city governments, business chains and financial institutions	Large nationwide or global enterprises
TECHNICAL ASPECTS	Ethernet and Ethernet switches; some L3 switching and routing	Ethernet switching on the LAN; Metro Ethernet on the MAN	MPLS, VPLS, SONET and satellite technologies

### इंटरनेट संकल्पना

- www परिभाषित करा
- WWW आर्किटेक्चर
- युजर इंटरफेस आणि एप्लिकेशन

WWW म्हणजे वर्ल्ड वाइड वेब. वर्ल्ड वाइड वेबची तांत्रिक व्याख्या अशी आहे: इंटरनेटवरील सर्व रिसोर्सेस आणि वापरकर्ते जे हायपरटेक्स्ट ट्रान्सफर प्रोटोकॉल (HTTP) वापरत आहेत.

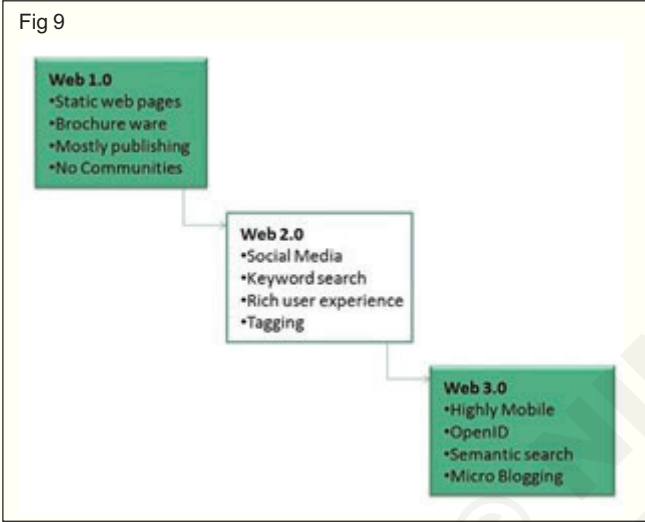
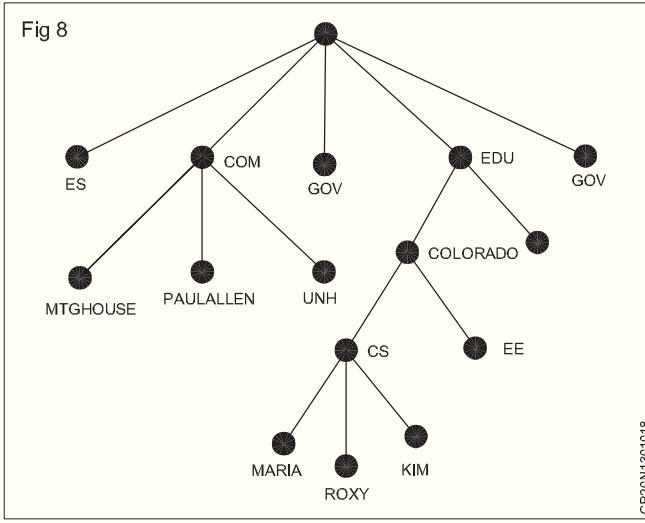
वर्ल्ड वाइड वेब हे नेटवर्क-अॅक्सेसिबल इनफार्मेशन चे विश्व आहे, मानवी ज्ञानाचे मूर्त फॉर्मेट आहे.

सोप्या भाषेत सांगायचे तर, वर्ल्ड वाइड वेब हा इंटरनेटवरील संगणकांमधील

इनफार्मेशन ची देवाणघेवाण करण्याचा एक पाथ आहे, त्यांना परस्परसंवादी मल्टीमीडिया संसाधनांच्या विशाल संग्रहामध्ये एकत्र बांधणे.

इंटरनेट आणि वेब एकच गोष्ट नाही: वेब इनफार्मेशन पार करण्यासाठी इंटरनेट वापरते. (पिक्चर 8)

उत्क्रांती: वर्ल्ड वाइड वेबची निर्मिती टिमोथी बर्नर्स ली यांनी 1989 मध्ये जिनिव्हा येथील CERN येथे केली होती. CERN मध्ये रिसर्च कांना एकत्रितपणे आणि कार्यक्षमतेने काम करता यावे यासाठी वर्ल्ड वाइड वेब त्यांच्या प्रस्तावानुसार अस्तित्वात आले. कालांतराने ते वर्ल्ड वाइड वेब बनले. खालील आकृती वर्ल्ड वाइड वेबच्या उत्क्रांतीची थोडक्यात व्याख्या करते: (पिक्चर 9)



### WWW आर्किटेक्चर

WWW आर्किटेक्चर खालील आकृतीमध्ये दर्शविल्याप्रमाणे अनेक स्तरांमध्ये विभागलेले आहे:

### ओळखणारे (Identifiers) आणि कॅरक्टर संच

युनिफॉर्म रिसोर्स आयडेंटिफायर (URI) चा वापर वेबवरील संसाधनांची अनन्यपणे ओळख करण्यासाठी केला जातो आणि UNICODE मानवी भाषांमध्ये वाचता आणि लिहिता येणारी वेब पेजेज तयार करणे शक्य करते.

### सिन्टेक्स

XML (एक्सटेंसिबल मार्कअप लॅंग्वेज) सिमेंटिक वेबमध्ये सामान्य वाक्यरचना परिभाषित करण्यात मदत करते.

### डेटा इंटरचेंज

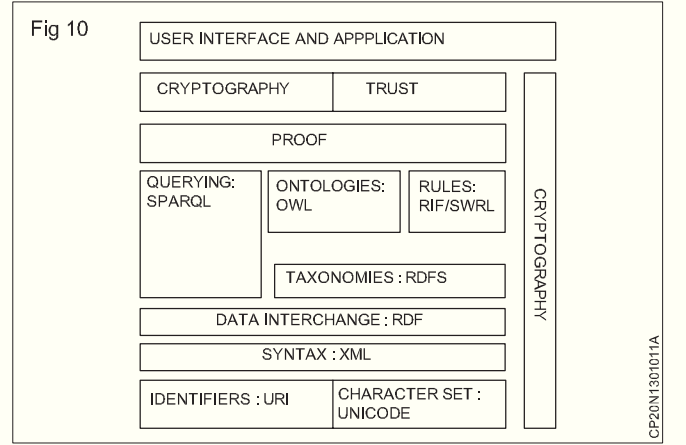
संसाधन डिस्क्रिप्शन फ्रेमवर्क (RDF) फ्रेमवर्क वेबसाठी डेटाचे मुख्य प्रतिनिधित्व परिभाषित करण्यात मदत करते. आरडीएफ आलेख स्वरूपात संसाधनाविषयी डेटा दर्शवते.

### वर्गीकरण (टॅक्सॉनॉमी)

RDF स्कीमा (RDFS) वर्गीकरण आणि इतर ऑन्टोलॉजिकल रचनांचे अधिक प्रमाणित डिस्क्रिप्शन करण्यास अनुमती देते.

### ऑन्टोलॉजीज (पिक्चर 10)

वेब ऑन्टोलॉजी लॅंग्वेज (OWL) RDFS वर अधिक रचना ऑफर करते. हे खालील तीन आवृत्त्यांमध्ये येते:



- वर्गीकरण आणि साध्या मर्यादांसाठी ओडब्ल्यूएल लाइट.
- संपूर्ण डिस्क्रिप्शन लॉजिक सपोर्टसाठी OWL DL.
- RDF च्या अधिक सिमेंटिक स्वातंत्र्यासाठी OWL

### नियम (Rules)

RIF आणि SWRL RDF आणि OWL कडून सबलब्ध असलेल्या बांधकामांच्या पलीकडे नियम ऑफर करतात. सिंपल प्रोटोकॉल आणि आरडीएफ केरी लॅंग्वेज (SPARQL) ही एसक्यूएल सारखी भाषा आहे जी आरडीएफ डेटा आणि ओडब्ल्यूएल ऑन्टोलॉजीजसाठी केरी करण्यासाठी वापरली जाते.

### पुरावा (Proof)

सर्व सिमेंटिक आणि नियम जे पुराव्याच्या खालील स्तरांवर अंमलात आणले जातात आणि त्यांचे रिझल्ट वजावट सिद्ध करण्यासाठी वापरले जातात.

### क्रिप्टोग्राफी

क्रिप्टोग्राफी म्हणजे स्त्रोतांच्या उत्पत्तीच्या पडताळणीसाठी डिजिटल स्वाक्षरीचा वापर केला जातो.

### युजर इंटरफेस आणि ऍप्लिकेशन

लेयरच्या शीर्षस्थानी युजर इंटरफेस आणि ऍप्लिकेशन्स स्तर यूजर च्या परस्परसंवादासाठी तयार केला आहे.

### WWW ऑपरेशन

WWW क्लायंट-सर्व्हर दृष्टिकोनावर कार्य करते. वेब कसे कार्य करते हे खालील चरण स्पष्ट करतात:

- 1 युजर वेब ब्राउझरच्या अॅड्रेस बारमध्ये वेब पेजची URL (म्हणजे <http://www.tutorialspoint.com>) प्रविष्ट करतो.
- 2 नंतर ब्राउझर [www.tutorialspoint.com](http://www.tutorialspoint.com) शी संबंधित IP पत्त्यासाठी डोमेन नेम सर्व्हरला विनंती करतो.
- 3 IP एड्रेस प्राप्त केल्यानंतर, ब्राउझर HTTP प्रोटोकॉल वापरून वेब सर्व्हरला वेब पेजसाठी विनंती पाठवते जे ब्राउझर आणि वेब सर्व्हरचे कम्युनिकेशन करण्याचा पाथ स्पेसिफाइड करते.
- 4 नंतर वेब सर्व्हर HTTP प्रोटोकॉल वापरून विनंती प्राप्त करतो आणि विनंती केलेल्या वेब पेजसाठी त्याचा सर्व्हर तपासतो. आढळल्यास ते वेब ब्राउझरवर परत करते आणि HTTP कनेक्शन बंद करा.
- 5 आता वेब ब्राउझर वेब पृष्ठ प्राप्त करतो, ते त्याचा अर्थ लावतो आणि वेब ब्राउझरच्या विंडोमध्ये वेब पेजची कंटेंट प्रदर्शित करतो.

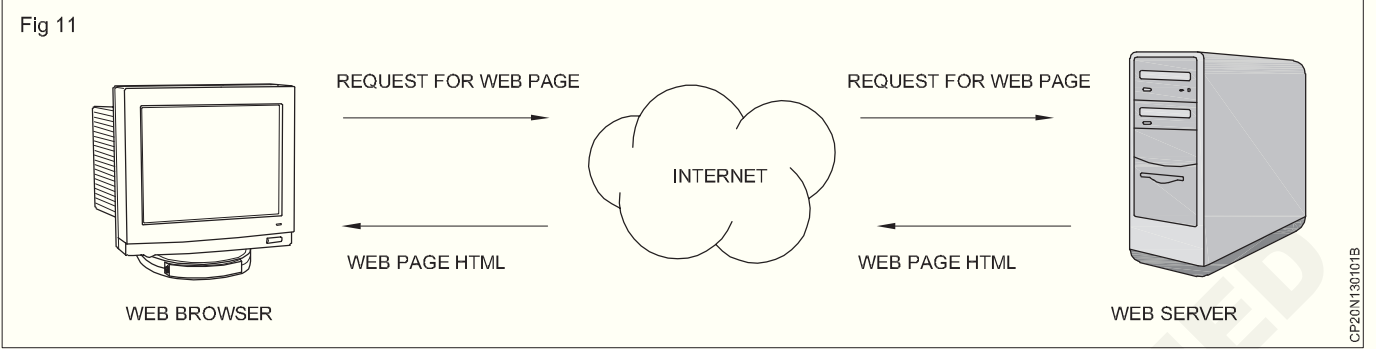
## भविष्य(Future)

वेब क्षेत्रात झपाट्याने विकास झाला होता. शिक्षण, रिसर्च न, तंत्रज्ञान, वाणिज्य, विपणन इत्यादी जवळपास प्रत्येक क्षेत्रात त्याचा इफेक्ट आहे. त्यामुळे वेबचे भविष्य जवळजवळ अप्रत्याशित आहे.

WWW च्या क्षेत्रातील मोठ्या विकासाव्यतिरिक्त, काही तांत्रिक समस्या देखील आहेत ज्यांना W3 कंसोर्टियमला सामोरे जावे लागते.

## यूजर इंटरफेस (पिक्चर 11)

3-डी इनफार्मेशन चे उच्च दर्जाचे प्रेसेंटेशन करण्याचे काम विकासाधीन आहे. W3 Consortium देखील सर्व प्रादेशिक भाषा आणि लेखन प्रणाल्यांचा समावेश असलेल्या ग्लोबल समुदायांच्या आवश्यकता पूर्ण करण्यासाठी वेब वाढवण्यास उत्सुक आहे.



## तंत्रज्ञान

प्रायव्हसी आणि सुरक्षिततेवर काम सुरू आहे. यामध्ये इनफार्मेशन लपवणे, लेखा, एक्सेस कंट्रोल, सचोटी आणि रिस्क मॅनेजमेंट यांचा समावेश असेल.

## आर्किटेक्चर

वेबच्या क्षेत्रात प्रचंड वाढ झाली आहे ज्यामुळे इंटरनेट ओव्हरलोड होऊ शकते आणि त्याचे कार्यप्रदर्शन खराब होऊ शकते. त्यामुळे अधिक चांगले प्रोटोकॉल विकसित करणे आवश्यक आहे.

ई-मेल (इलेक्ट्रॉनिक मेल) म्हणजे काय?

ई-मेल (इलेक्ट्रॉनिक मेल) ही संगणकाची देवाणघेवाण आहे - इंटरनेटद्वारे एका यूजर कडून एक किंवा अधिक प्राप्तकर्त्याकडे स्टोर संदेश. ई-मेल हा व्यवसाय किंवा वैयक्तिक वापरासाठी संवाद साधण्याचा जलद, स्वस्त आणि एक्सेस जोगी पाथ आहे.

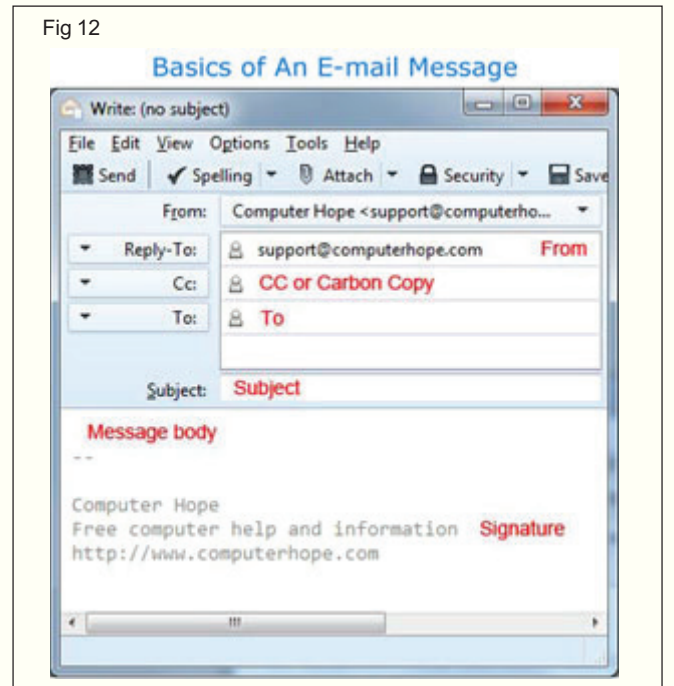
## ई-मेल लिहिणे (पिक्चर 12)

ई-मेल संदेश लिहिताना, तो खालील उदाहरण विंडोसारखा दिसला पाहिजे. जसे आपण पाहू शकता, ई-मेल पाठवताना अनेक फील्ड आवश्यक आहेत:

- To फील्ड हे ई-मेल प्राप्त करणाऱ्या व्यक्तीचा ई-मेल एड्रेस ठेवला जातो.
- From फील्डमध्ये तुमचा ई-मेल एड्रेस असावा.
- तुम्ही संदेशाला उत्तर देत असल्यास, प्रति: आणि प्रेषक: फील्ड ऑटोमॅटिक भरले जातात. जर तो नवीन संदेश असेल, तर तुम्हाला प्राप्तकर्त्यांना तुमच्या संपर्क लिस्ट मधून निवडून किंवा ई-मेल एड्रेस टाईप करून To: फील्डमध्ये स्पेसिफाइड करणे आवश्यक आहे. तुम्ही एकापेक्षा जास्त प्राप्तकर्ता (उदा. ग्रुप ई-मेल) एंटर केल्यास, एड्रेस स्वल्पविराम आणि स्पेसने किंवा टॅब की दाबून वेगळे केले जावेत.
- विषयामध्ये ई-मेलच्या मजकुराचे डिस्क्रिप्शन करणारे काही शब्द असावेत. पूर्ण ई-मेल न उघडता आणि वाचल्याशिवाय, विषय प्राप्तकर्त्याला ई-मेल कशाबद्दल आहे ते पाहू देतो. हे फील्ड ऐच्छिक आहे.

- CC ("कार्बन कॉपी") फील्ड तुम्हाला प्राप्तकर्ते स्पेसिफाइड करू देते जे थेट एड्रेस नाहीत ("टू" फील्डमध्ये लिस्ट बद्ध). उदाहरणार्थ, तुम्ही जेफ आणि सीसी लिंडा आणि स्टीव्हन यांना ई-मेल पाठवू शकता. ई-मेल जेफला उद्देशून असला तरी, लिंडा आणि स्टीव्हन यांनाही एक प्रत मिळते आणि प्रत्येकजण पाहू शकतो की ई-मेल कोणाला प्राप्त झाला आहे. हे फील्ड ऐच्छिक आहे.
- BCC ("अंध कार्बन कॉपी") फील्ड CC प्रमाणेच आहे, प्राप्तकर्ते गुप्त असल्याशिवाय. प्रत्येक BCC प्राप्तकर्त्याला ई-मेल प्राप्त होतो, परंतु त्याची प्रत कोणाला मिळाली हे पाहत नाही. एड्रेस ("टू" फील्डमध्ये लिस्ट बद्ध केलेले कोणीही) सर्व प्राप्तकर्त्यांना व्हा मान राहतात. हे फील्ड ऐच्छिक आहे.
- शेवटी, मेसेज बॉडी हे पोजिशन आहे जिथे तुम्ही तुमचा मुख्य संदेश टाईप करता. त्यात अनेकदा तळाशी तुमची स्वाक्षरी असते; हस्तलिखित पत्रासारखे.

Fig 12



## वेब डिझायनिंग (Web designing)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- वेब साइट्स आणि वेब पेजेज स्पष्ट करा
- स्टॅटिक आणि डायनॅमिक वेब पेजेज स्पष्ट करा
- HTML, DHTML आणि XML स्पष्ट करा
- वेब होस्टिंग, वेब सर्व्हर, ऍप्लिकेशन सर्व्हर आणि डेटाबेस सर्व्हरची संकल्पना स्पष्ट करा

### परिचय

वर्ल्ड वाइड वेब (WWW) 1990 मध्ये CERN फिजिकल शास्त्रज्ञ टिम बर्नर्स ली यांनी तयार केले होते. 30 एप्रिल 1993 रोजी, CERN ने घोषणा केली की वर्ल्ड वाइड वेब कोणासाठीही वापरण्यास विनाव्हॅल्यु असेल. एचटीएमएल आणि एचटीटीपीच्या परिचयापूर्वी, फाइल ट्रान्सफर प्रोटोकॉल आणि गोफर प्रोटोकॉल सारख्या इतर प्रोटोकॉलचा वापर सर्व्हरवरून वैयक्तिक फाइल्स पुनर्प्राप्त करण्यासाठी केला जात असे. हे प्रोटोकॉल एक साधी डिरेक्ट्री स्ट्रक्चर ऑफर करतात जी युजर नेव्हिगेट करतो आणि डाउनलोड करण्यासाठी फाइल्स निवडतो. डॉक्युमेंट्स बहुतेक वेळा फॉर्मॅटिंगशिवाय प्लेन टेक्स्ट फाइल्स म्हणून सादर केले जातात किंवा वर्ड प्रोसेसर फॉर्मॅटमध्ये एन्कोड केलेले होते.

### वेबसाइट्स

वेबसाइट, वेब साइट, वेब साइट किंवा फक्त साइट म्हणून देखील लिहिलेली, लिखित भाषा, प्रतिमा, व्हिडिओ, ध्वनी इत्यादी कंटेंट (मीडिया) असलेल्या संबंधित वेब पृष्ठाचा संच आहे. वेबसाइट किमान एका वेबवर होस्ट केली जाते. सर्व्हर, इंटरनेट सारख्या नेटवर्कद्वारे किंवा युनिफॉर्म रिसोर्स लोकेटर म्हणून ओळखल्या जाणाऱ्या इंटरनेट पत्त्याद्वारे प्राइवेट लोकल एरिया नेटवर्कद्वारे एक्सेस करण्यायोग्य. सर्व सार्वजनिकरित्या एक्सेस करण्यायोग्य वेबसाइट एकत्रितपणे वर्ल्ड वाइड वेब बनवतात.

वेबपेज हे एक डॉक्युमेंट्स आहे, जे सामान्यतः हायपरटेक्स्ट मार्कअप लॅंग्वेज (HTML) XHTML च्या फॉर्मॅटिंग सूचनांसह जोडलेले साध्या टेक्स्ट मध्ये लिहिलेले असते. वेबपेज योग्य HTML अँकरसह इतर वेबसाइटवरील एलिमेंट समाविष्ट करू शकते.

हायपरटेक्स्ट ट्रान्सफर प्रोटोकॉल (HTTP) सह वेब पेजवर एक्सेस आणि वाहतूक केली जाते, जे वेबपेज कंटेंट च्या यूजर साठी सुरक्षितता आणि प्रायव्हसी प्रदान करण्यासाठी वैकल्पिकरित्या एन्क्रिप्शन सुरक्षित, HTTPS वापरू शकते. यूजर चा ऍप्लिकेशन, अनेकदा वेब ब्राउझर, त्याच्या HTML मार्कअप सूचनांनुसार पृष्ठ कंटेंट संगणक मॉनिटरवर रेंडर करतो.

वेब अँड्रेस नावाच्या साध्या युनिफॉर्म रिसोर्स लोकेटर (URL) वरून वेबसाइटची पेजेज सामान्यतः एक्सेस केली जाऊ शकतात. पेजच्या URL त्यांना एका पदानुक्रमात मॅनेज करतात, जरी त्यांच्यामधील हायपरलिंक वाचकाला समजलेला साइटमॅप दर्शविते

आणि साइटच्या वाचकांच्या नेव्हिगेशनसाठी पाथ दर्शन करते ज्यात

सामान्यतः साइटच्या वेब कंटेंट च्या बहुतेक लिंक्ससह मुख्यपृष्ठ आणि पृष्ठ, संपर्क पृष्ठ आणि लिंक पृष्ठाबद्दल पूरक.

काही वेबसाइटना त्यांच्या काही किंवा सर्व कंटेंट मध्ये एक्सेस करण्यासाठी सदस्यता आवश्यक आहे. सबस्क्रिप्शन वेबसाइट्सच्या उदाहरणांमध्ये अनेक व्यावसायिक साइट्स, न्यूज वेबसाइट्सचे पार्ट, शैक्षणिक जर्नल वेबसाइट्स, गेमिंग वेबसाइट्स, फाइल-शेअरिंग वेबसाइट्स, इंटरनेट फोरम, वेब-आधारित ईमेल, सोशल नेटवर्किंग वेबसाइट्स, रिअल-टाइम स्टॉक मार्केट डेटा प्रदान करणाऱ्या वेबसाइट्स आणि वेबसाइट्सचा समावेश होतो. इतर विविध सर्विस (उदा., प्रतिमा, फाइल्स आणि इतर गोष्टींचे संचयन आणि/किंवा शेअरिंग ऑफर करणाऱ्या वेबसाइट्स).

वेबसाइट असू शकते

- वैयक्तिक वेबसाइट
- व्यावसायिक वेबसाइट
- ई-सरकार
- ना-नफा संस्था वेबसाइट.

### स्टॅटिक वेबसाइट

स्टॅटिक वेबसाइट अशी आहे की ज्यामध्ये क्लायंट वेब ब्राउझरला पाठवल्या जाणाऱ्या फॉर्मॅटमध्ये सर्व्हरवर वेब पेजेज स्टोर केली जातात. हे प्रामुख्याने हायपरटेक्स्ट मार्कअप लॅंग्वेज (HTML) मध्ये कोड केलेले आहे.

वेबसाइट्सचे साधे फॉर्मॅट किंवा विपणन उदाहरणे, जसे की क्लासिक वेबसाइट, पाच पेजची वेबसाइट किंवा ब्रोशर वेबसाइट या बऱ्याचदा स्थिर वेबसाइट असतात, कारण त्या यूजर ला पूर्व-परिभाषित, स्थिर इनफार्मेशन सादर करतात. यात टेक्स्ट, फोटो, अॅनिमेशन, ऑडिओ/व्हिडिओ आणि परस्परसंवादी मेनू आणि नेव्हिगेशनद्वारे कंपनी आणि तिची उत्पादने आणि सर्विस बद्दल इनफार्मेशन समाविष्ट असू शकते.

या टाइप ची वेबसाइट सहसा सर्व अभ्यागतांना समान इनफार्मेशन प्रदर्शित करते. ग्राहकांना किंवा क्लायंटला छापिलेले इनफार्मेशन पत्रक देण्यासारखेच, स्थिर वेबसाइट सामान्यतः एक्सटेंडेड कालावधीसाठी सातत्यपूर्ण, स्टॅण्डर्ड इनफार्मेशन प्रदान करते. वेबसाइट मालक वेळोवेळी अपडेट करत असला तरी, टेक्स्ट, फोटो आणि इतर कंटेंट एडिट करण्यासाठी ही एक मॅन्युअल प्रक्रिया आहे आणि त्यासाठी बेसिक वेबसाइट डिझाइन कौशल्ये आणि सॉफ्टवेअरची आवश्यकता असू शकते.

## फायदे

- स्थिर वेबसाइट्स विकसित करणे सोपे आहे
- त्वरीत विकसित केले जाऊ शकते
- ते सर्च इंजिनद्वारे सहजपणे अनुक्रमित केले जातात कारण सर्व वेब पेजेज प्रत्यक्षात सर्व्हरवर अस्तित्वात आहेत, जे डायनॅमिक वेबसाइट्सच्या बाबतीत नाही.

## तोटे

- स्थिर वेबसाइट अनेक ऑनलाइन सर्विस साठी आवश्यक असलेली जटिल फंक्शन करू शकत नाहीत.
- संपूर्ण साइट अपडेट करणे अवघड आणि वेळखाऊ असू शकते.
- स्थिर वेबसाइट्समध्ये डेटा आणि डिझाइनचे अलगाव प्रदान केले जात नाही

## डायनॅमिक वेबसाइट

डायनॅमिक वेबसाइट अशी आहे जी स्पेसिफिक निकषांच्या आधारे स्वतःमध्ये वारंवार आणि ऑटोमेटिक बदलते किंवा कस्टमाइझ करते.

डायनॅमिक वेबसाइट्समध्ये दोन टाइप चे डायनॅमिक क्रियाकलाप असू शकतात: कोड आणि कंटेंट . डायनॅमिक कोड अव्ह किंवा पडद्यामागे आहे आणि डायनॅमिक कंटेंट व्ह मान किंवा पूर्णपणे प्रदर्शित आहे

## फायदे

- ऑनलाइन सर्विस साठी आवश्यक असलेले अधिक जटिल कार्य करू शकते.
- ते अपडेट करणे सोपे आहे.
- डेटा आणि डिझाइनचे पृथक्करण प्रोग्रामर आणि कंटेंट लेखकांना स्वतंत्रपणे कार्य करण्यास अनुमती देते.

## तोटे

- बांधण्यासाठी अधिक वेळ लागू शकतो.
- बांधणे कठीण होऊ शकते.
- डायनॅमिक वेबसाइट्स सर्च इंजिनद्वारे सहजपणे अनुक्रमित केल्या जात नाहीत, कारण त्यांच्याकडे वेब सर्व्हरवर वास्तविक वेब पेजेज नसतात. सर्च इंजिन तंत्रज्ञानामध्ये सतत होत असलेल्या सुधारणांमुळे, ही समस्या आता खूप दूर झाली आहे आणि आजकाल अनेक डायनॅमिक वेबसाइट्स सर्च इंजिनद्वारे खूप चांगल्या प्रकारे अनुक्रमित केल्या गेल्या आहेत.

अशी काही वर्गीकरणे आहेत:

- संलग्न: सक्षम पोर्टल जे केवळ त्याचे कस्टम CMSच नाही तर इतर कंटेंट प्रदात्यांकडील सिंडिकेटेड कंटेंट देखील मान्य शुल्कासाठी प्रस्तुत करते. सहसा तीन संबंध स्तर असतात. संलग्न एजन्सी (उदा., कमिशन जंक्शन), जाहिरातदार (उदा., eBay) आणि ग्राहक (उदा. Yahoo!).

- संग्रहण साइट: नामशेष होण्याच्या मार्गावर असलेली मौल्यवान इलेक्ट्रॉनिक कंटेंट सेव्ह करण्यासाठी वापरली जाते. उदाहरणार्थ: इंटरनेट आर्काइव्ह, ज्याने 1996 पासून अब्जावधी जुनी आणि नवीन वेब पेजेज सेव्ह केली आहेत; आणि Google Groups, ज्यांनी 2005 च्या सुरुवातीला युजनेट बातम्या/चर्चा गटांवर पोस्ट केलेले 845,000,000 संदेश सेव्ह केले होते.
- ब्लॉग साइट: सामान्यतः ऑनलाइन डायरी, कमेंट किंवा दृश्ये पोस्ट करण्यासाठी वापरल्या जाणाऱ्या साइट ज्यामध्ये चर्चा मंच (उदा. ब्लॉगर, Xanga) समाविष्ट असू शकतात.
- कंटेंट साइट: या साइट अंतिम यूजर ला मूळ कंटेंट तयार करतात आणि विकतात. (उदा., स्लेट, About.com).
- कॉर्पोरेट वेबसाइट: व्यवसाय, संस्था किंवा सेवेशी संबंधित इनफार्मेशन प्रदान करण्यासाठी वापरली जाते.
- कॉमर्स साइट (किंवा ईकॉमर्स साइट): या साइट्स Amazon.com, CSN स्टोअर्स आणि Overstock.com सारख्या वस्तू खरेदी किंवा विक्रीसाठी डिझाइन केल्या आहेत.
- समुदाय साइट: अशा साइट जिथे समान रूची असलेल्या व्यक्ती चॅटिंग आणि मेसेजिंगद्वारे किंवा मायस्पेस किंवा Facebook सारख्या सामाजिक संदेश बोर्डद्वारे एकमेकांशी संवाद साधतात.
- शहराची साइट: एक साइट जी स्पेसिफिक शहर किंवा गाव आणि त्या गावात घडणाऱ्या घटनांची इनफार्मेशन दर्शवते. सहसा नगर परिषदेने तयार केले. उदाहरणार्थ, Richmond.com हे रिचमंड, व्हर्जिनियासाठी जिओडोमेन आहे.
- इनफार्मेशन साइट: अभ्यागतांना इनफार्मेशन देण्याच्या उद्देशाने असलेली कंटेंट आहे, परंतु व्यावसायिक हेतूसाठी आवश्यक नाही, जसे की: RateMyProfessors.com, मोफत इंटरनेट लेक्सिकॉन आणि एनसायक्लोपीडिया. बहुतेक सरकारी, शैक्षणिक आणि ना-नफा संस्थांकडे इनफार्मेशन ची साइट असते.
- मिरर साइट: वेबसाइटचे संपूर्ण पुनरुत्पादन.
- बातम्या साइट: इनफार्मेशन साइट प्रमाणेच, परंतु बातम्या आणि भाष्य देण्यासाठी समर्पित.
- वैयक्तिक मुख्यपृष्ठ: एखाद्या व्यक्तीद्वारे किंवा एखाद्या लहान गटाद्वारे चालवले जाते जसे की एक कुटुंब ज्यामध्ये इनफार्मेशन किंवा व्यक्ती समाविष्ट करू इच्छित असलेली कोणतीही कंटेंट असते. हे सहसा Geocities सारख्या वेब होस्टिंग सेवेचा वापर करून अपलोड केले जातात.
- फिश साइट: इलेक्ट्रॉनिक कम्युनिकेशन मध्ये विश्वासार्ह व्यक्ती किंवा व्यवसाय (जसे की सामाजिक सुरक्षा एडमिनिस्ट्रेशन , PayPal) म्हणून वेष करून पासवर्ड आणि क्रेडिट कार्ड तपशील यासारखी सेन्सिटिव्ह इनफार्मेशन फसवणूक करण्यासाठी तयार केलेली वेबसाइट.
- राजकीय साइट: एक साइट ज्यावर लोक राजकीय विचार व्यक्त करू शकतात.



- रेटिंग साइट: एक साइट ज्यावर लोक वैशिष्ट्यीकृत केलेल्या गोष्टींची प्रशंसा किंवा अपमान करू शकतात. • रिव्यु साइट: एक साइट ज्यावर लोक उत्पादने किंवा सर्विस साठी पुनरावलोकने पोस्ट करू शकतात.
- शाळेची साइट: एक साइट ज्यावर शिक्षक, विद्यार्थी किंवा एडमिनिस्ट्रेटर त्यांच्या शाळेतील करंट घडामोडींची इनफार्मेशन पोस्ट करू शकतात.
- व्हिडिओ शेअरिंग: एक साइट जी वापरकर्त्यांना व्हिडिओ अपलोड करण्यास सक्षम करते, जसे की YouTube आणि Google Video.
- सर्च इंजिन साइट: एक साइट जी सामान्य इनफार्मेशन प्रदान करते आणि इतर साइट्स पुनर्प्राप्त करण्यासाठी गेटवे म्हणून अभिप्रेत आहे. Google, Yahoo आणि MSN ही सर्वात मोठ्या प्रमाणावर ज्ञात सर्च इंजिने आहेत.
- शॉक साइट: बहुतेक दर्शकांसाठी (उदा. rotten.com) आक्षेपाई बनवण्याच्या हेतूने असलेल्या प्रतिमा किंवा इतर कंटेंट समाविष्ट करते.
- Warez: वापरकर्त्यांना कॉपीराइट केलेली कंटेंट बेकायदेशीरपणे डाउनलोड करू देण्यासाठी आणि होस्ट करण्यासाठी डिझाइन केलेली साइट.
- वेब पोर्टल: साइट हे वाहन आहे जे इंटरनेट किंवा इंटरनेटवरील इतर संसाधनांसाठी एक्सेस द्वारे प्रदान करते.

## वेब पेजेज

वेबपेज किंवा वेबपेज हे एक डॉक्युमेंट्स किंवा इनफार्मेशन संसाधन आहे जे जगभरातील वेबसाठी योग्य आहे आणि वेब ब्राउझरद्वारे एक्सेस केला जाऊ शकतो आणि संगणक प्रदर्शन किंवा मोबाइल डिव्हाइसवर प्रदर्शित केला जाऊ शकतो. ही इनफार्मेशन सहसा एचटीएमएल किंवा एक्सएचटीएमएल फॉर्मॅटमध्ये असते आणि ती हायपर टेक्स्ट हायपर लिंकद्वारे इतर वेब पेजेसवर नेव्हिगेशन बार देऊ शकते. वेब पेजेज वारंवार इतर रिसोर्सेस जसे की कॅस्केडिंग स्टाइल पत्रक, क्लायंट-साइड-स्क्रिप्टिंग आणि प्रतिमा त्यांच्या अंतिम सादरीकरणामध्ये समाविष्ट करतात.

वेब पेजेज लोकल संगणकावरून किंवा दूरस्थ वेब सर्व्हरवरून पुनर्प्राप्त केली जाऊ शकतात. वेब सर्व्हर केवळ प्राइवेट नेटवर्कवर एक्सेस प्रतिबंधित करू शकतो, उदा. कॉर्पोरेट इंटरनेट किंवा ते वर्ल्ड वाइड वेबवर पेजेज प्रकाशित करू शकते. वेब पेजची विनंती केली जाते आणि वेबवरून सर्विस दिली जाते. हायपरटेक्स्ट ट्रान्सफर प्रोटोकॉल (HTTP) वापरून वेब सर्व्हर.

वेब पेजमध्ये स्थिर टेक्स्ट च्या फायली आणि वेब सर्व्हरच्या फाइल सिस्टममध्ये (स्टॅटिक वेब पृष्ठ) स्टोर केलेल्या इतर वेब कंटेंट चा समावेश असू शकतो किंवा जेव्हा त्यांना विनंती केली जाते तेव्हा सर्व्हर-साइड स्क्रिप्टिंगद्वारे तयार केले जाऊ शकते (डायनॅमिक वेब पृष्ठ). क्लायंट-साइड स्क्रिप्टिंग क्लायंट ब्राउझरवर एकदा यूजर च्या इनपुटसाठी वेब पेजना अधिक प्रतिसाद देऊ शकते.

## वेब ब्राउझर

वेब ब्राउझरमध्ये ग्राफिकल यूजर इंटरफेस असू शकतो, जसे की इंटरनेट एक्सप्लोरर, मोझिला फायरफॉक्स, गूगल क्रोम आणि

ऑपेरा (वेब ब्राउझर), किंवा कमांड लाइन इंटरफेस असू शकतो, जसे की लिंक्स (वेब ब्राउझर) (पिक्चर 1) किंवा लिंक्स (वेब ब्राउझर). अपंग असलेले वेब वापरकर्ते सहसा वेब ऍक्सेसिबिलिटी वेब पेजेससाठी सहाय्यक तंत्रज्ञान आणि अनुकूली धोरणे वापरतात. वापरकर्ते कलर ब्लाइंड असू शकतात, कदाचित पुनरावृत्ती झालेल्या तणावाच्या दुखापतीमुळे किंवा मोटरन्युरोनच्या समस्यांमुळे माउस वापरू इच्छित नसू शकतात, बहिरा असू शकतात आणि कॅप्शन देण्यासाठी ऑडिओ आवश्यक असू शकतात, अंध असू शकतात आणि स्क्रीन रीडर किंवा डिस्प्ले वापरत आहेत, स्क्रीन मॅग्निफिकेशनची आवश्यकता असू शकते, इ. अक्षम आणि सक्षम वापरकर्ते वेळ, नेटवर्क बँडविड्थ वाचवण्यासाठी किंवा फक्त त्यांचा ब्राउझिंग अनुभव सुलभ करण्यासाठी प्रतिमा आणि इतर मीडिया डाउनलोड आणि पाहणे अक्षम करू शकतात.

मोबाईल डिव्हाइसेसच्या वापरकर्त्यांकडे अनेकदा प्रतिबंधित डिस्प्ले आणि बँडविड्थ असतात. कोणीही फॉन्ट, फॉन्ट आकार, स्टाइल आणि रंगसंगती वापरण्यास प्राधान्य देऊ शकत नाही.

वेब पृष्ठ डिझाइनर आणि पेजवर त्यांची स्वतःची CSS स्टाइल लागू करू शकतात. वर्ल्ड वाइड वेब कन्सोर्टियम (W3C) आणि वेब ऍक्सेसिबिलिटी इनिशिएटिव्ह (WAI) शिफारस करतात की सर्व वेब पेजेज हे सर्व पर्याय लक्षात घेऊन डिझाइन केले जावे.

## इंटरनेटवरून सॉफ्टवेअर डाउनलोड करणे

- 1 माय कॉम्प्युटर उघडून एक तात्पुरती फाइल्स फोल्डर तयार करा, तुमच्या हार्ड ड्राइव्हर (सामान्यतः C: ड्राइव्ह) डबल क्लिक करा, त्यानंतर आकृती 2 प्रमाणे फाइल/नवीन/फोल्डर निवडा.
- 2 "तात्पुरती फाइल" टाईप करा आणि पिक्चर 3 प्रमाणे नाव द्या.
- 3 इंटरनेटवरील फ्रीवेअर सॉफ्टवेअर वेबसाइटवरून "माय SQL 5.1 फ्री डाउनलोड" टाईप करा

**टीप:** उदाहरण म्हणून "http:// dev.mysql.com/downloads/" निवडा "माय SQL कम्प्युनिटी सर्व्हर" उघडलेल्या साइटवरून डाउनलोड करताना पिक्चर 4 नुसार

- 4 आकृती 5 प्रमाणे मागील आवृत्ती पर्याय सर्व करण्यापासून "MySQL समुदाय सर्व्हर 5.1" वर क्लिक करा.
- 5 "Windows (x86, 32-bit), MSI Installer Essentials - Recommended" निवडा आणि डाउनलोड वर क्लिक करा. (पिक्चर 6)

**टीप:** "माय SQL 5.1.63" तयार केलेल्या फोल्डरच्या नावात "तात्पुरती फोल्डर" सेव्ह करा.

- 6 इन्स्टॉलेशनसाठी CD ROM मध्ये डाउनलोड केलेले "My SQL 5.1.63" बर्न करा.

Fig 1

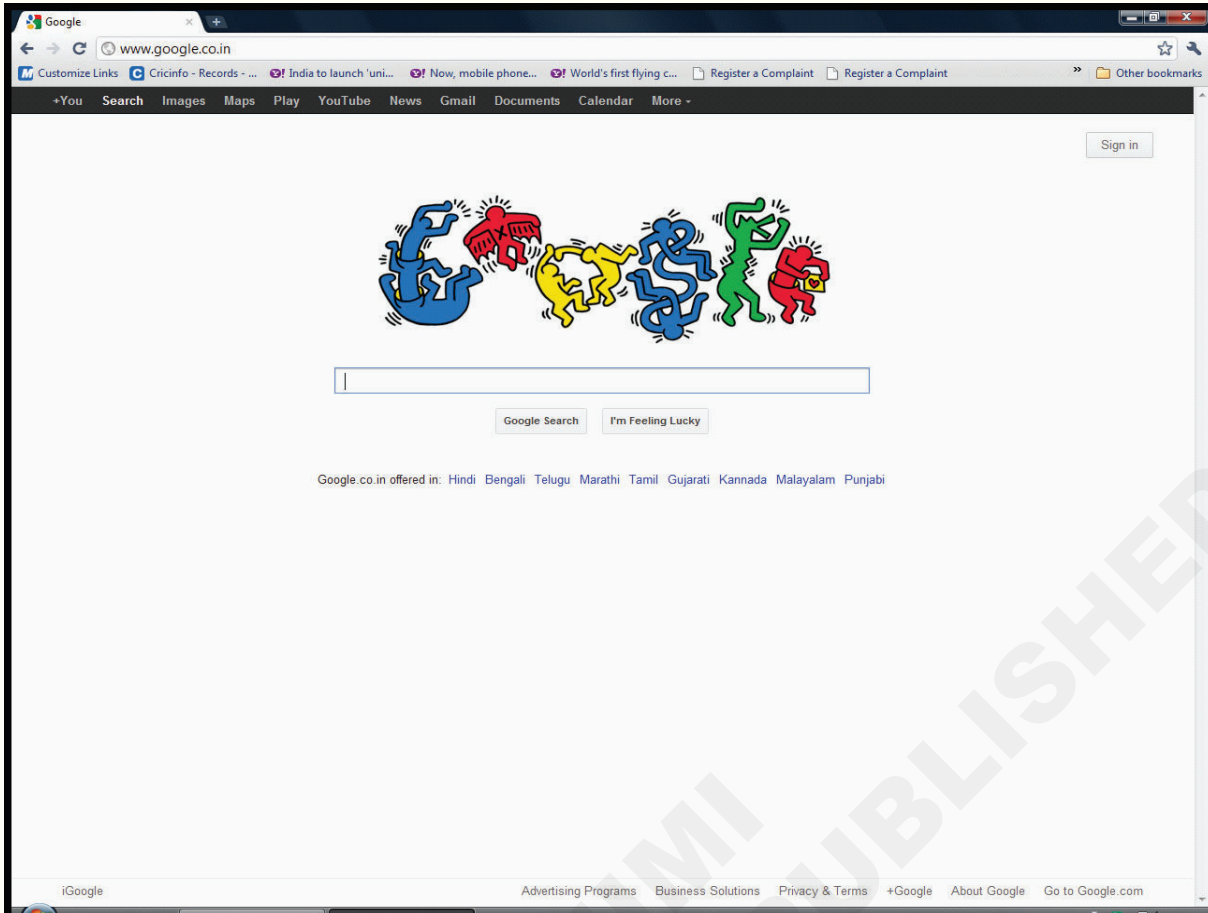


Fig 2

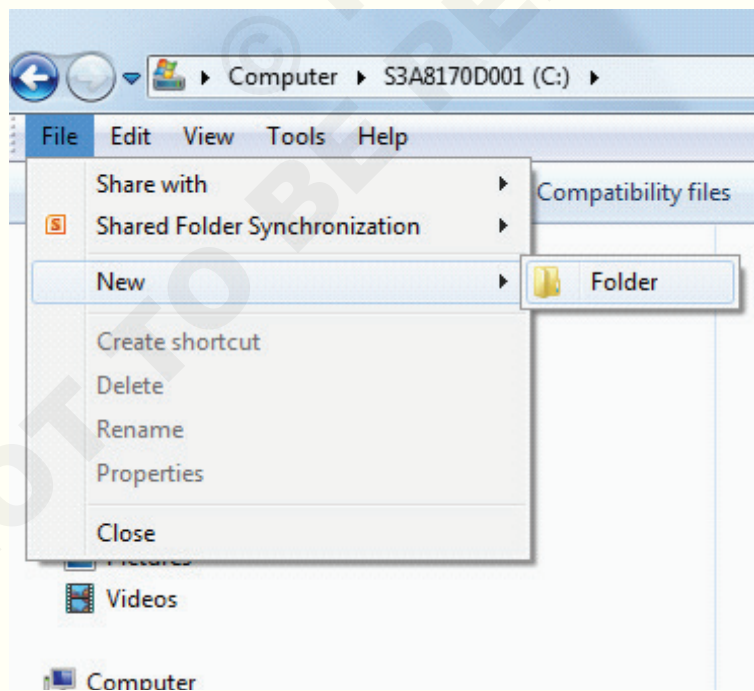


Fig 3



Fig 4

My Sql Community



Fig 5

## Download MySQL Community Server

MySQL Community Edition is a freely downloadable version of the world's most popular open source database that is supported by an active community of open source developers and enthusiasts.

MySQL Cluster Community Edition is available as a separate download. The reason for this change is so that MySQL Cluster can provide more frequent updates and support using the latest sources of MySQL Cluster Carrier Grade Edition.

### Important Platform Support Updates

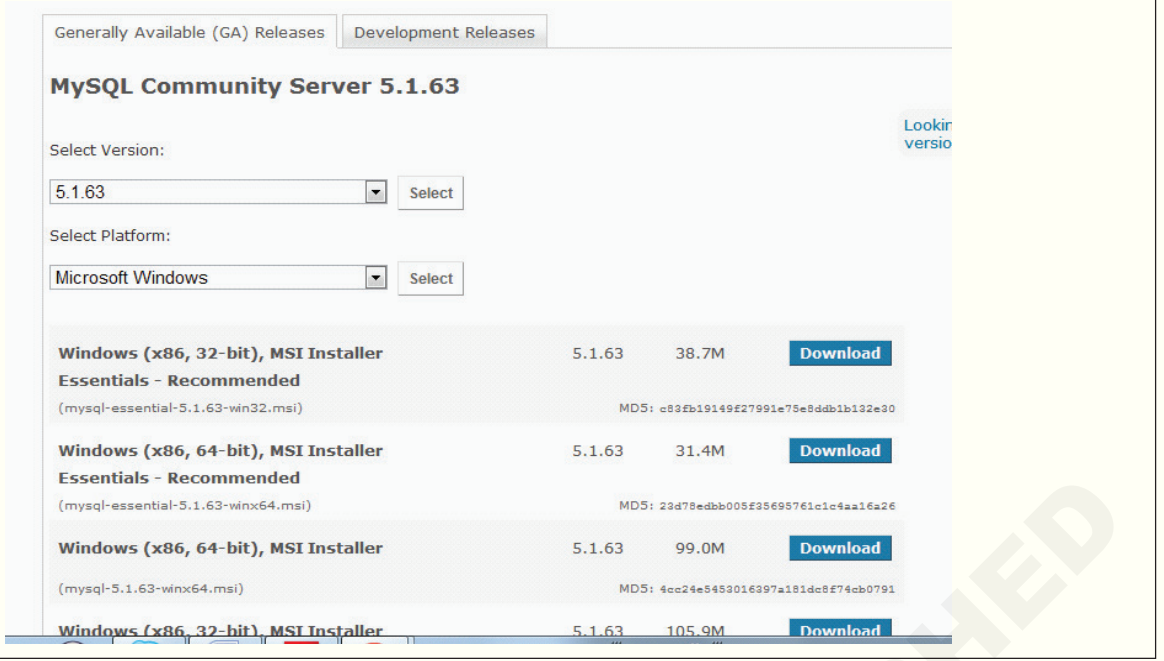
#### Online Documentation

- [MySQL 5.5 Documentation and Change History for the Generally Available \(GA\) Release](#)
- [MySQL 5.1 Documentation and Change History for the previous Generally Available \(GA\) Release](#)

#### Looking for previous GA versions?

- [MySQL Community Server 5.1 »](#)
- [MySQL Community Server 5.0 »](#)

Fig 6



## वेब लैंग्वेज

वेब भाषांना मार्कअप भाषा म्हणतात कारण टेक्स्ट ची प्रक्रिया, व्याख्या आणि सादरीकरणासाठी डिझाइन केलेले आहेत. भाषा टेक्स्ट फाइलमध्ये फॉर्मॅट नासाठी कोड स्पेसिफाइड करते, लेआउट आणि स्टाइल दोन्ही. फॉर्मॅटिंग स्पेसिफाइड करण्यासाठी वापरल्या जाणाऱ्या कोडला टॅग म्हणतात

मार्कअप भाषांचे चार टाइप

1 BML

2 HTML

3 DHTML

4 XML

**BML (बेटर मार्कअप लैंग्वेज):** BML ही मूलतः साधी मॅक्रो भाषा आहे. मॅक्रोना BML मध्ये ब्लॉक म्हणतात. ब्लॉक लूक फाइल्समध्ये परिभाषित केले जातात आणि बीएमएल फाइल्समध्ये मागवले जातात. ब्लॉक्स पॅरामीटर्स स्वीकारतात आणि पॅरामीटर्स कसे प्रसारित केले जातात आणि ब्लॉकची व्याख्या त्यांचा वापर करण्यास सक्षम आहे त्यानुसार अनेक प्रकारांमध्ये विपार्ट ले जातात.

**HTML (हायपर टेक्स्ट मार्कअप लैंग्वेज):** एचटीएमएल किंवा हायपरटेक्स्ट मार्कअप लैंग्वेज ही वेबची भाषा आहे. सर्व वेब पेजेज HTML मध्ये लिहिलेली आहेत. HTML वेब ब्राउझरमध्ये प्रतिमा, मल्टीमीडिया आणि टेक्स्ट प्रदर्शित करण्याचा पाथ परिभाषित करते. यात डॉक्युमेंट्स (हायपरटेक्स्ट) कनेक्ट करण्यासाठी आणि वेब डॉक्युमेंट्स परस्परसंवादी बनवण्यासाठी एलिमेंट समाविष्ट आहेत (जसे की फॉर्मसह).

HTML ही एक परिभाषित स्टॅण्डर्ड मार्कअप लैंग्वेज आहे. ते स्टॅण्डर्ड वर्ल्ड वाइड वेब कन्सोर्टियम (W3C) ने विकसित केले होते. हे SGML (स्टॅण्डर्ड जनरलाइज्ड मार्कअप लैंग्वेज) वर आधारित आहे. ही एक लैंग्वेज आहे जी

तुमच्या मजकुराची रचना परिभाषित करण्यासाठी टॅग वापरते. एलिमेंट आणि टॅग < आणि > कॅरक्टर द्वारे परिभाषित केले जातात.

**DHTML:** डायनॅमिक एचटीएमएल हे एचटीएमएलचे नवीन तपशील नाही, तर स्टॅण्डर्ड एचटीएमएल कोड आणि कमांड्स पाहण्याचा आणि कंट्रोल करण्याचा एक नवीन पाथ आहे.

डायनॅमिक एचटीएमएलचा विचार करताना, आम्हाला स्टॅण्डर्ड एचटीएमएलचे गुण लक्षात ठेवणे आवश्यक आहे, स्पेशल तः एकदा सर्व्हरवरून पृष्ठ लोड झाल्यानंतर, सर्व्हरवर दुसरी विनंती येईपर्यंत ते बदलणार नाही. डायनॅमिक एचटीएमएल एचटीएमएल एलिमेंटवर अधिक कंट्रोल देते आणि वेब सर्व्हरवर परत न येता त्यांना कधीही बदलण्याची परवानगी देते.

DHTML चे चार पार्ट आहेत:

- डॉक्युमेंट्स ऑब्जेक्ट मॉडेल (DOM) (definition)
- स्क्रिप्ट
- कॅस्केडिंग स्टाइल शीट्स (CSS)
- XHTML

**DOM:** DOM ला वेब पेज चा कोणताही पार्ट DHTML सह बदलण्यासाठी एक्सेस करण्याची परवानगी आहे. वेब पेज चा प्रत्येक पार्ट DOM द्वारे स्पेसिफाइड केला जातो आणि त्याच्या सुसंगत नामकरण पद्धतीचा वापर करून ते एक्सेस करू शकतात आणि त्यांचे प्रॉपर्टीज बदलू शकतात.

**स्क्रिप्ट:** JavaScript किंवा ActiveX मध्ये लिहिलेल्या स्क्रिप्ट या दोन सर्वात सामान्य स्क्रिप्टिंग भाषा आहेत ज्या DHTML ऍक्टिव्ह करण्यासाठी वापरल्या जातात. DOM मध्ये स्पेसिफाइड केलेल्या वस्तू कंट्रोल करण्यासाठी तुम्ही स्क्रिप्टिंग भाषा वापरता.

**कॅस्केडिंग स्टाइल पत्रके:** वेब पेज चे फॉर्मॅट आणि अनुभव कंट्रोल करण्यासाठी DHTML मध्ये CSS चा वापर केला जातो. स्टाइल पत्रके

टेक्स्ट चे रंग आणि फॉन्ट, बॅकग्राउंड रंग आणि प्रतिमा आणि पेज वरील ऑब्जेक्ट्सचे पोजिशन परिभाषित करतात. स्क्रिप्टिंग आणि DOM वापरून, आम्ही विविध एलिमेंटची स्टाइल बदलू शकतो.

**XHTML:** XHTML किंवा HTML 4.x चा वापर स्वतः पेज तयार करण्यासाठी आणि CSS आणि DOM वर कार्य करण्यासाठी एलिमेंट तयार करण्यासाठी केला जातो. डीएचटीएमएलसाठी एक्सएचटीएमएल बदल काही स्पेशल नाही - परंतु वैध एक्सएचटीएमएल असणे अधिक महत्वाचे आहे, कारण ब्राउझरपेक्षा त्यामधून बरेच काही कार्य करते.

DHTML ची वैशिष्ट्ये

DHTML ची चार प्राथमरी वैशिष्ट्ये आहेत:

- 1 टॅग आणि प्रॉपर्टीज बदलणे
- 2 रिअल-टाइम स्थिती
- 3 डायनॅमिक फॉन्ट (नेटस्केप कम्युनिकेटर)
- 4 डेटा बंधनकारक (इंटरनेट एक्सप्लोरर)

### टॅग आणि प्रॉपर्टीज बदलणे

हा DHTML च्या सर्वात सामान्य वापरांपैकी एक आहे. हे ब्राउझरच्या बाहेरील इव्हेंट (जसे की माउस क्लिक, वेळ किंवा date इत्यादी) वर अवलंबून HTML टॅगचे गुण बदलू देते. आम्ही याचा वापर पेजवर इनफार्मेशन प्रीलोड करण्यासाठी करू शकतो आणि जोपर्यंत वाचक स्पेसिफिक दुव्यावर क्लिक करत नाही तोपर्यंत ते प्रदर्शित करू शकत नाही.

### रिअल-टाइम पोस्टिंग

वेब पृष्ठाभोवती फिरत असलेल्या वस्तू, प्रतिमा आणि टेक्स्ट . हे आम्हाला वाचकांसह परस्परसंवादी गेम खेळू शकतो किंवा स्क्रीनचे पार्ट ॲनिमेट करू शकतो.

### डायनॅमिक फॉन्ट

हे फक्त नेटस्केप वैशिष्ट्य आहे. नेटस्केपने डिझायनर्सना वाचकांच्या सिस्टीमवर कोणते फॉन्ट असतील हे माहित नसलेल्या समस्या सोडवण्यासाठी हे विकसित केले. डायनॅमिक फॉन्टसह, फॉन्ट एन्कोड केले जातात आणि पृष्ठासह डाउनलोड केले जातात, जेणेकरून पेज नेहमी डिझाइनरचा हेतू कसा होता ते दिसते.

डेटा बॉन्डिंग: हे फक्त IE वैशिष्ट्य आहे. मायक्रोसॉफ्टने हे विकसित केले आहे जेणेकरून वेब साइट्सवरील डेटाबेसमध्ये सहज एक्सेस मिळू शकेल. हे डेटाबेसमध्ये एक्सेस करण्यासाठी CGI वापरण्यासारखे आहे, परंतु कार्य करण्यासाठी ActiveX कंट्रोल वापरते.

**XML:** एक्स्टेंसिबल मार्कअप लॅंग्वेज (XML) ही एक मार्कअप भाषा आहे जी मानव-वाचनीय (रिडेबल) आणि मशीन वाचण्यायोग्य अशा फॉर्मॅटमध्ये डॉक्युमेंट्स एन्कोडिंगसाठी नियमांचा संच परिभाषित करते. हे W3C द्वारे निर्मित XML 1.0 स्पेसिफिकेशन आणि इतर अनेक संबंधित वैशिष्ट्यांमध्ये, सर्व विनाव्हॅल्यु खुल्या मानकांमध्ये परिभाषित केले आहे. XML ची डिझाइन उद्दिष्टे साधेपणा, सामान्यता आणि इंटरनेटवरील सबयोगिता यावर जोर देतात. जगातील भाषांसाठी युनिकोड द्वारे भक्कम सपोर्ट असलेले हे टेक्स्ट डेटा फॉर्मॅट आहे. जरी XML ची रचना डॉक्युमेंट्स वर लक्ष केंद्रित करते, तरीही ते अकंट्रोल डेटा संरचनांचे प्रतिनिधित्व करण्यासाठी मोठ्या प्रमाणावर वापरले जाते, उदाहरणार्थ वेब सर्विस मध्ये

**HTML डॉक्युमेंट्स तयार करणे:** वेब पेज लिहिण्यासाठी कोड लिहिण्यास स्टार्ट करण्यापूर्वी, वेब पृष्ठाच्या देखाव्याची योजना करणे हा एक चांगला सराव आहे. HTML डॉक्युमेंट्स त दोन एलिमेंट असतात:

- 1 डॉक्युमेंट्स कंटेंट
- 2 टॅग

डॉक्युमेंट्स कंटेंट ही वेब पृष्ठावरील इनफार्मेशन आहे जी यूजर ला दिसेल. ती इनफार्मेशन टेक्स्ट किंवा ग्राफिक्स असू शकते.

टॅग हे HTML कोड आहेत जे डॉक्युमेंट्स कंटेंट कशी दिसेल हे कंट्रोल करतात. टॅग, दुसऱ्या शब्दांत, टेक्स्ट ठळक, काळा किंवा निळा किंवा टाइम न्यू रोमन किंवा एअरल फॉन्ट प्रकाराचा असेल हे निर्धारित करतील.

नोटपॅड सुरू करा

नोटपॅड सुरू करण्यासाठी येथे जा:

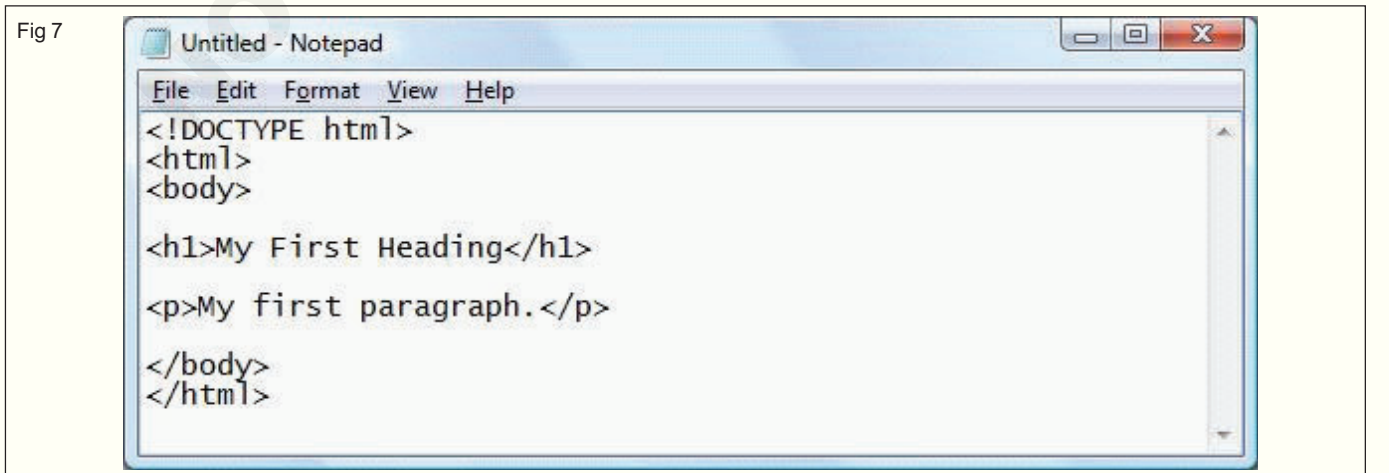
सुरू करा

सर्व प्रोग्रॅम

ॲक्सेसरीज

नोटपॅड

नोटपॅडसह तुमचा HTML एडिट करा (पिक्चर 7)



तुमचा HTML कोड तुमच्या Notepad मध्ये टाइप करा:

तुमचे HTML सेव्ह करा

नोटपॅडच्या फाइल मेनूमध्ये म्हणून सेव्ह करा निवडा.

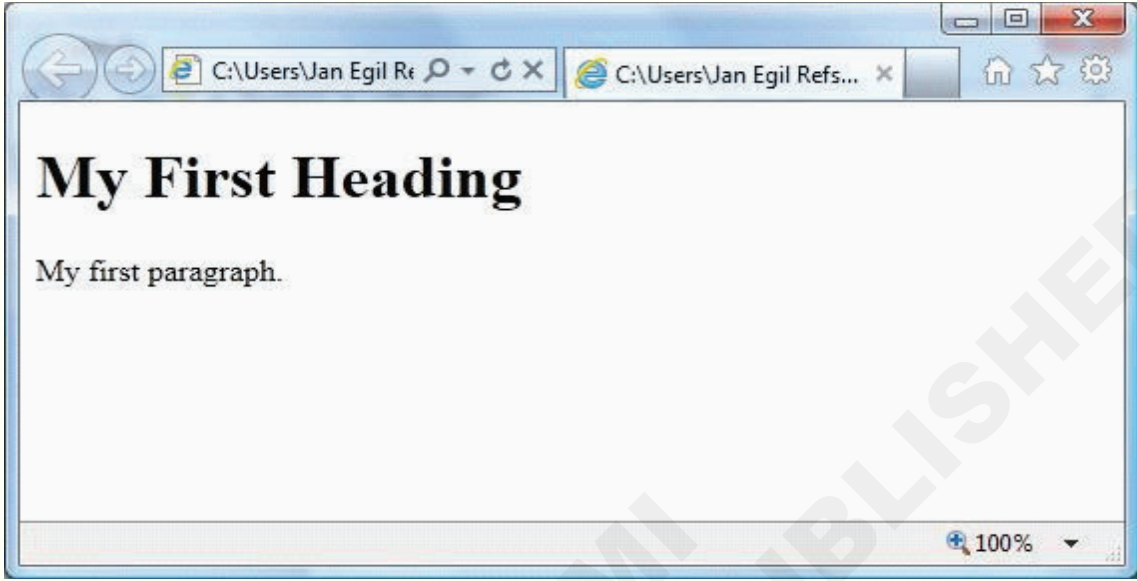
जेव्हा तुम्ही HTML फाइल सेव्ह करता, तेव्हा तुम्ही .htm किंवा .html फाइल एक्सपान्शन वापरू शकता. फाइल लक्षात ठेवण्यास सोपी असलेल्या फोल्डरमध्ये सेव्ह करा

तुमच्या ब्राउझरमध्ये HTML चालवा

तुमचा वेब ब्राउझर सुरू करा आणि फाइल, ओपन मेनूमधून तुमची html फाइल उघडा किंवा फक्त फोल्डर ब्राउझ करा आणि तुमच्या HTML फाइलवर डबल क्लिक करा.

रिझल्ट यासारखे दिसले पाहिजे: (पिक्चर 8)

Fig 8



### मार्कअप भाषेची स्ट्रक्चर

HTML डॉक्युमेंट्स त दोन\* मुख्य पार्ट असतात:

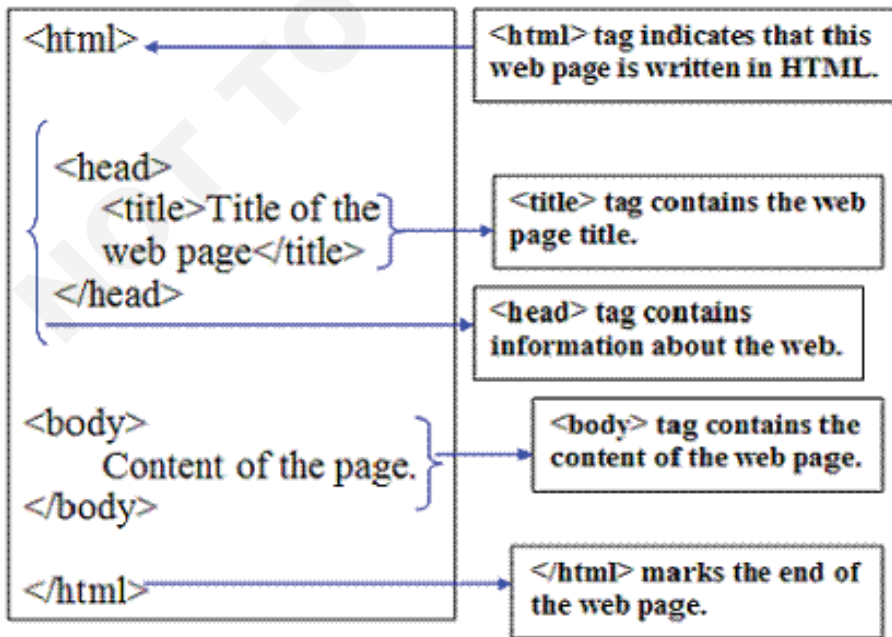
- 1 हेड. हेड एलिमेंटमध्ये वेब डॉक्युमेंटचे टाईटल आणि मेटा डेटा असतो.
- 2 बॉडी. बॉडी एलिमेंटमध्ये तुम्हाला वेब पेजवर दाखवायची असलेली इनफार्मेशन असते.

तुमची वेब पेज एचटीएमएल 4 शी सुसंगत बनवण्यासाठी, तुम्हाला

एचटीएमएल एलिमेंटपूर्वी डॉक्युमेंट्स टाइप घोषणा (डीटीडी) जोडणे आवश्यक आहे. अनेक वेब ऑथरिंग सॉफ्टवेअर तुम्ही नवीन वेब पेज तयार करता तेव्हा डीटीडी आणि बेसिक टॅग ऑटोमैटिक जोडतात.

वेब पेजवर, पहिला टॅग (स्पेशल तः, <html>) डॉक्युमेंट्स साठी वापरल्या जाणाऱ्या मार्कअप भाषा दर्शवतो. <head> टॅगमध्ये वेब पृष्ठाबद्दल इनफार्मेशन असते. शेवटी, कंटेंट <body> टॅगमध्ये दिसते. (पिक्चर 9)

Fig 9



<!DOCTYPE> Declaration

वेबवर अनेक भिन्न डॉक्युमेंट्स आहेत. ब्राउझर केवळ डॉक्युमेंट्स योग्यरित्या प्रदर्शित करू शकतो, जर त्याला माहित असेल की ते कोणत्या टाइप चे डॉक्युमेंट्स आहे.

HTML च्या बऱ्याच भिन्न आवृत्त्या देखील आहेत आणि ब्राउझर केवळ HTML पृष्ठ 100% योग्यरित्या प्रदर्शित करू शकतो जर त्याला पृष्ठमध्ये वापरलेली अचूक HTML आवृत्ती माहित असेल. यासाठी <!DOCTYPE> वापरले जाते.

<!DOCTYPE> हा HTML टॅग नाही. HTML कोणत्या आवृत्तीमध्ये लिहिले आहे याबद्दल ब्राउझरला इनफार्मेशन (घोषणा) आहे.

HTML <head> Element

<head> Element सर्व प्रमुख एलिमेंटसाठी एक कंटेनर आहे. <head> मधील एलिमेंटमध्ये स्क्रिप्टचा समावेश असू शकतो, ब्राउझरला स्टायल शीट कुठे शोधायचे, मेटा इनफार्मेशन प्रदान करू शकतात आणि बरेच काही करू शकतात.

खालील टॅग हेड विभागात जोडले जाऊ शकतात: <title>, <base>, <link>, <meta>, <script>, आणि <style>.

The HTML <title>

टॅग डॉक्युमेंट्स चे टाईटल परिभाषित करतो.

सर्व HTML/XHTML डॉक्युमेंट्स मध्ये टाईटल एलिमेंट आवश्यक आहे.

टाईटल एलिमेंट :

- ब्राउझर टूलबारमध्ये टाईटल परिभाषित करते.
- आवडीमध्ये जोडल्यावर पेजसाठी टाईटल प्रदान करते.
- सर्च -इंजिन परिणामांमध्ये पेजसाठी टाईटल प्रदर्शित करते.

### HTML एलिमेंट सिंटॅक्स

- HTML एलिमेंट स्टार्ट टॅग / ओपनिंग टॅगसह सुरू होतो
- HTML एलिमेंट एंड टॅग / क्लोजिंग टॅगसह समाप्त होतो
- एलिमेंट कंटेंट स्टार्ट आणि शेवट टॅग दरम्यान सर्वकाही आहे • काही HTML एलिमेंटमध्ये रिक्त कंटेंट आहे
- स्टार्ट टॅगमध्ये रिक्त एलिमेंट बंद आहेत
- बहुतेक HTML एलिमेंटमध्ये स्पेशल असू शकतात

HTML Headings

HTML हेडिंग <h1> ते <h6> टॅगसह परिभाषित केले आहेत.

उदाहरणे

<html>

<body>

<h1>This is heading 1</h1>

<h2>This is heading 2</h2>

<h3>This is heading 3</h3>

<h4>This is heading 4</h4>

<h5>This is heading 5</h5>

<h6>This is heading 6</h6>

</body>

</html>

### रिझल्ट

हे heading 1 आहे

हे heading 2 आहे

हे heading 3 आहे

हे heading 4 आहे

हे heading 5 आहे

हे heading 6 आहे

HTML प्याराग्राफ

HTML प्याराग्राफ <p> टॅगसह परिभाषित केले आहेत.

<html>

<body>

<p>This is a paragraph.</p>

<p>This is a paragraph.</p>

<p>This is a paragraph.</p>

</body>

</html>

### उदाहरणे

This is a paragraph.

This is a paragraph.

This is a paragraph

HTML Links

HTML links are defined with the <a> tag.

<html>

<body>

<a href="http://www.facebook.com">

This is a link</a>

</body>

</html>

Result

This is a link

By clicking the link it shows the facebook login page

HTML Images

HTML images are defined with the <img> tag.

<html>

<body>



</body>

</html>

रिझल्ट (पिक्चर 10)



HTML Attributes

- HTML एलिमेंटमध्ये स्पेशल असू शकतात
- स्पेशल ता एखाद्या एलिमेंटबद्दल अतिरिक्त इनफार्मेशन प्रदान करतात

- स्पेशल ता नेहमी स्टार्ट टॅगमध्ये नमूद केल्या जातात
- स्पेशल ता नाव/व्हॅल्यू जोड्यांमध्ये येतात जसे की: name="value"

### Attributes उदाहरण

HTML दुवे <a> टॅगसह परिभाषित केले आहेत. लिंक एट्रिब्युट href स्पेशल ता मध्ये स्पेसिफाइड केला आहे:

<html>

<body>

<a href="http://www.yahoo.com">

This is a link</a>

</body>

</html>

Result

This is the link

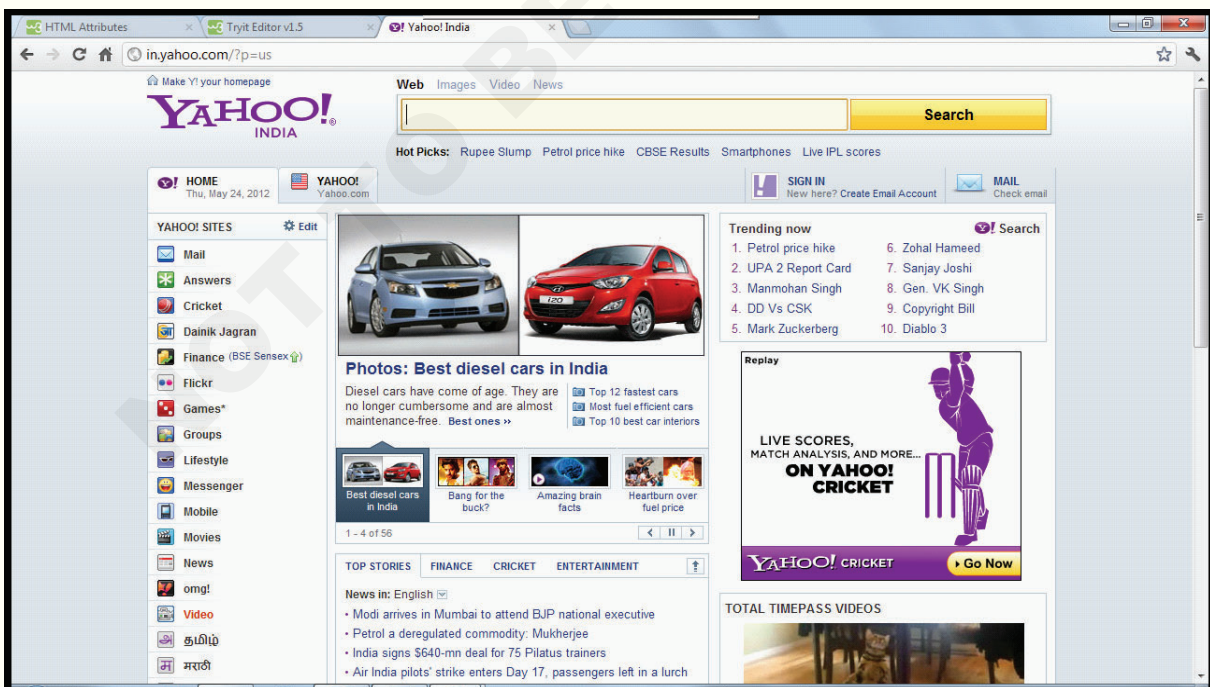
By clicking the link yahoo home page appears (Fig 11).

Formatting

Create Line Breaks - The <br /> Element

जेव्हा जेव्हा <br /> एलिमेंट , तेव्हा त्याच्या खालील काहीही पुढील ओळीवर सुरू होते. हा टॅग रिकाम्या एलिमेंटचे उदाहरण आहे, जिथे तुम्हाला टॅग उघडणे आणि बंद करणे आवश्यक नाही, कारण त्यांच्यामध्ये जाण्यासाठी काहीही नाही.

Fig 11





**Example:**

Hello<br/>

You come most carefully upon your hour.<br/>

Thanks<br/>

Mahnaz

**Result**

Hello

You come most carefully upon your hour.

Thanks

Mahnaz

To Become

Centring Content - The <center> Element:

You can use <center> tag to put any content in the center of the page or any table cell.

**Example**

<p>This is not in the center.</p>

<center>

<p>This is in the center.</p>

</center>

This will produce following result:

This is not in the center.

This is in the center

**सॉफ्ट हायफन**

अधूनमधून, परिच्छेदाचे औचित्य सिद्ध करण्यासाठी तुम्ही ब्राउझरला लांब शब्द हायफन करण्याची परवानगी देऊ इच्छित असाल. उदाहरणार्थ, खालील कोड आणि त्याचे परिणामी आउटपुट विचारात घ्या.

<p style="text-align: justify;"> The morbid fear of the number 13, or triskaidekaphobia, has plagued some important historic figures like Mahamiya and Nanao.</p>

**हे पुढील रिझल्ट देखील**

सॉफ्ट हायफनचे उदाहरण - क्रमांक 13 ची भयंकर भीती, किंवा त्रिस्कायडेकाफोबिया, महामिया आणि नानाओ सारख्या काही महत्वाच्या ऐतिहासिक व्यक्तींना त्रास देतात.

फॉरमॅटिंग सेव्ह करा - <pre> एलिमेंट

काहीवेळा तुमचा टेक्स्ट HTML डॉक्युमेंट्स त कसा लिहिला जातो याचे अचूक फॉरमॅट फॉलो करावे असे तुम्हाला वाटते. अशा प्रकरणांमध्ये, तुम्ही प्रीफॉर्मेट केलेला टॅग (<pre>) वापरू शकता.

ओपनिंग <pre> टॅग आणि क्लोजिंग </pre> टॅगमधील कोणताही टेक्स्ट सोर्स डॉक्युमेंट्स चे फॉरमॅटिंग सेव्ह करेल.

```
<pre>
```

```
function testFunction( strText ){
```

```
  alert (strText)
```

```
}
```

```
</pre>
```

This will produce following result:

```
function testFunction( strText ){
```

```
  alert (strText)
```

```
}
```

**हॉरिजॉन्टल नियम - <hr /> एलिमेंट**

डॉक्युमेंट्स चे विपार्ट व्ह मानपणे तोडण्यासाठी क्षैतिज नियम वापरले जातात. <hr> टॅग डॉक्युमेंट्स तील करंट स्थितीपासून उजव्या समासापर्यंत एक ओळ तयार करतो आणि त्यानुसार ओळ खंडित करतो.

उदाहरणार्थ, आपण दोन परिच्छेदांमध्ये खालीलप्रमाणे एक ओळ देऊ शकता:

<p>This is paragraph one and should be on top</p>

<hr />

<p>This is paragraph two and should be at bottom</p>

p>

हे खालील रिझल्ट देखील:

हा प्याराग्राफ एक आहे आणि वर असावा

हा प्याराग्राफ दोन आहे आणि तळाशी असावा

पुन्हा <hr /> टॅग हे रिकाम्या एलिमेंटचे उदाहरण आहे, जिथे तुम्हाला ओपनिंग आणि क्लोजिंग टॅगची आवश्यकता नाही, कारण त्यांच्यामध्ये जाण्यासारखे काहीही नाही.

टीप: <hr /> एलिमेंटमध्ये hr आणि फॉरवर्ड स्लॅश या कॅरक्टर मध्ये जागा आहे. तुम्ही ही जागा वगळल्यास, जुन्या ब्राउझरला लाइन ब्रेक रेंडर करण्यात अडचण येईल, जर तुम्ही फॉरवर्ड स्लॅश कॅरक्टर चुकवला आणि फक्त <hr> वापरला तर ते वैध XHTML नाही.

## प्रेझेंटेशनल टॅग

तुम्ही वर्ड प्रोसेसर वापरत असल्यास, तुम्ही टेक्स्ट ठळक, इटालिक किंवा अधोरेखित करण्याच्या क्षमतेची परिचित आहात; HTML आणि XHTML मध्ये टेक्स्ट कसा दिसू शकतो हे सूचित करण्यासाठी सबलब्ध दहा पर्यायांपैकी हे फक्त तीन आहेत.

### Bold Text - The <b> Element

<b>...</b> एलिमेंटमध्ये दिसणारी कोणतीही गोष्ट ठळक मध्ये प्रदर्शित केली जाते, जसे की येथे ठळक शब्द:

<p>The following word uses a <b>bold</b> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द ठळक टाइपफेस वापरतो.

### Italic Text - The <i> Element

<i>...</i> एलिमेंटमध्ये दिसणारी कोणतीही गोष्ट तिर्यकीकृत मध्ये प्रदर्शित केली जाते, जसे की येथे italicized शब्द:

<p>The following word uses a <i>italicized</i> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द इटॅलिक टाइपफेस वापरतो.

### Underlined Text - The <u> Element:

:<u>...</u> एलिमेंटमध्ये दिसणारी कोणतीही गोष्ट अधोरेखित करून दाखवली जाते, जसे की येथे अधोरेखित केलेला शब्द:

<p>The following word uses a <u>underlined</u> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द अधोरेखित टाइपफेस वापरतो.

### Strike Text - The <strike> Element:

< Strike >...</ Strike > एलिमेंटमध्ये दिसणारी कोणतीही गोष्ट स्ट्राइकथ्रू प्रदर्शित केली जाते, जी टेक्स्ट द्वारे एक पातळ रेषा आहे:

<p>The following word uses a <strike>strikethrough</strike> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द स्ट्राइकथ्रू टाइपफेस वापरतो.

### Monospaced font - The <tt> Element:

<tt> एलिमेंटची कंटेंट मोनोस्पेस फॉन्टमध्ये लिहिली जाते. बहुतेक फॉन्ट व्हेरिअबल-रुंदीचे फॉन्ट म्हणून ओळखले जातात कारण भिन्न अक्षरे

58 IT & ITES : COPA (NSQF - उजळणी 2022) - एक्सरसाईस साठी संबंधित थिअरी 1.31.113

वेगवेगळ्या रुंदीची असतात (उदाहरणार्थ, अक्षर m हे अक्षर i पेक्षा मोठे आहे). मोनोस्पेस फॉन्टमध्ये, तथापि, प्रत्येक अक्षराची रुंदी समान असते.

<p>The following word uses a <tt>monospaced</tt> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द मोनोस्पेस टाइपफेस वापरतो.

### Superscript Text - The <sup> Element

<sup> एलिमेंटची कंटेंट सुपरस्क्रिप्टमध्ये लिहिली जाते; वापरलेला फॉन्ट आकार त्याच्या सभोवतालच्या कॅरक्टर इतकाच आहे परंतु इतर कॅरक्टर पेक्षा अर्धा अक्षरांची उंची दर्शविली जाते.

<p>The following word uses a <sup>superscript</sup> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द सुपरस्क्रिप्ट टाइपफेस वापरतो.

### Subscript Text - The <sub> Element:

<sub> एलिमेंटची कंटेंट सबस्क्रिप्टमध्ये लिहिली जाते; वापरलेला फॉन्ट आकार त्याच्या सभोवतालच्या कॅरक्टर सारखाच असतो, परंतु इतर कॅरक्टर च्या खाली अर्धा अक्षरांची उंची दर्शविली जाते.

<p>The following word uses a <sub>subscript</sub> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द सबस्क्रिप्ट टाइपफेस वापरतो.

### Larger Text - The <big> Element:

<big> एलिमेंटची कंटेंट त्याच्या सभोवतालच्या उर्वरित टेक्स्ट पेक्षा एक फॉन्ट आकाराने मोठ्या प्रमाणात प्रदर्शित केली जाते.

<p>The following word uses a <big>big</big> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द मोठा टाइपफेस वापरतो.

### Smaller Text - The <small> Element:

<small> एलिमेंटची कंटेंट त्याच्या सभोवतालच्या उर्वरित मजकुरापेक्षा एक फॉन्ट आकार लहान प्रदर्शित केली जाते.

<p>The following word uses a <small>small</small> typeface.</p>

हे खालील रिझल्ट देखील:

खालील शब्द लहान टाइपफेस वापरतो.

## CSS सह HTML ची स्टाइल करणे

एचटीएमएल एलिमेंटना स्टाइल करण्याचा एक चांगला पाथ प्रदान करण्यासाठी HTML 4 सह CSS सादर करण्यात आले.

HTML मध्ये CSS खालील प्रकारे जोडले जाऊ शकते:

- इनलाईन - HTML एलिमेंटमध्ये स्टाइल स्पेशल ता वापरणे
- इंटरनल - <head> विभागातील <style> एलिमेंट वापरणे
- एक्सटर्नल - एक्सटर्नल CSS फाइल वापरणे

```
<html>
<body style="background-color:PowderBlue;">
<h1>Look! Styles and colors</h1>
<p style="font-family:verdana;color:red;">
This text is in Verdana and red</p>
<p style="font-family:times;color:green;">
This text is in Times and green</p>
<p style="font-size:30px;">This text is 30 pixels high</
p>
</body>
</html>
```

रिझल्ट :

दिसत! स्टाइल आणि रंग

हा टेक्स्ट वरदाना आणि लाल रंगात आहे

हा टेक्स्ट टाइम्स आणि ग्रीन मध्ये आहे

हा टेक्स्ट 30 पिक्सेल उंच आहे

## एचटीएमएल हायपरलिंक्स (लिंक)

हायपरलिंक (किंवा लिंक) हा एक शब्द, शब्दांचा समूह किंवा प्रतिमा आहे ज्यावर तुम्ही नवीन डॉक्युमेंट्स किंवा करंट डॉक्युमेंट्स तील नवीन विभागात जाण्यासाठी क्लिक करू शकता.

जेव्हा तुम्ही वेब पेजवरील लिंकवर कर्सर हलवता तेव्हा बाण थोडे हातात बदलेल.

लिंक्स HTML मध्ये <a> टॅग वापरून स्पेसिफाइड केल्या आहेत.

<a> टॅग दोन प्रकारे वापरला जाऊ शकतो:

- 1 दुसऱ्या डॉक्युमेंटची लिंक तयार करण्यासाठी, href स्पेशल ता वापरून
- 2 डॉक्युमेंट्स त बुकमार्क तयार करण्यासाठी, नाव स्पेशल ता वापरून लिंकसाठी HTML कोड सोपे आहे.

हे असे दिसते: <a href="url">लिंक टेक्स्ट </a>

href स्पेशल ता लिंकचे डेस्टिनेशन पोजिशन स्पेसिफाइड करते.

<a href="http://www.yahoo.com/">yahoo ला भेट द्या</a> जे असे दिसले: yahoo.com ला भेट द्या

या हायपरलिंकवर क्लिक केल्याने युजर Yahoo च्या होमपेजवर पाठवेल.

"लिंक टेक्स्ट " हा टेक्स्ट असणे आवश्यक नाही. ती प्रतिमा किंवा इतर HTML एलिमेंट असू शकते.

HTML लिंक्स - टार्गेट प्रॉपर्टीज

टार्गेट अॅट्रिब्यूट हे लिंक केलेले डॉक्युमेंट कुठे उघडायचे ते स्पेसिफाइड करते.

खालील उदाहरण लिंक केलेला डॉक्युमेंट्स नवीन ब्राउझर विंडोमध्ये किंवा नवीन टॅबमध्ये उघडेल:

## उदाहरण

```
<a href="http://www.yahoo.com/" target="_blank">Visit
yahoo !</a>
<html>
<body>
<a href="http://www.yahoo.com" target="_blank">Visit
yahoo.com!</a>
<p>If you set the target attribute to "_blank", the link will
open in a new browser window/tab.</p>
</body>
</html>
```

## रिझल्ट

yahoo.com ला भेट द्या!

तुम्ही टार्गेट स्पेशल ता "\_blank" वर सेट केल्यास, लिंक नवीन ब्राउझर विंडो/टॅबमध्ये उघडेल.

HTML प्रतिमा - <img> टॅग आणि Src स्पेशल ता

HTML मध्ये, प्रतिमा <img> टॅगसह परिभाषित केल्या जातात.

<img> टॅग रिकामा आहे, याचा अर्थ त्यात फक्त स्पेशल ता समाविष्ट आहेत आणि क्लोजिंग टॅग नाही.

पेजवर प्रतिमा प्रदर्शित करण्यासाठी, तुम्हाला src स्पेशल ता वापरण्याची आवश्यकता आहे. Src चा अर्थ "सोर्स " आहे. src स्पेशल ताचे व्हॅल्यू आपण प्रदर्शित करू इच्छित प्रतिमेची URL आहे.

प्रतिमा परिभाषित करण्यासाठी वाक्यरचना:

```

```

URL प्रतिमा संचयित केलेल्या स्थानाकडे निर्देश करते. "www.w3schools.com" वरील "images" डिरेक्ट्री असलेल्या "bamboo.gif" नावाच्या प्रतिमेची URL आहे: http:// www.backgroundlabs.com/index.php?search=bamboo.

डॉक्युमेंट्स त जेथे <img> टॅग आढळतो तेथे ब्राउझर प्रतिमा प्रदर्शित करतो. तुम्ही दोन परिच्छेदांमध्ये इमेज टॅग लावल्यास, ब्राउझर पहिला प्याराग्राफ , नंतर इमेज आणि नंतर दुसरा प्याराग्राफ दाखवतो.

### HTML प्रतिमा - Alt Attribute

आवश्यक alt स्पेशल ता प्रतिमेसाठी पर्यायी टेक्स्ट स्पेसिफाइड करते, जर प्रतिमा प्रदर्शित केली जाऊ शकत नाही.

alt स्पेशल ताचे व्हॅल्यू लेखक-परिभाषित टेक्स्ट आहे:

```

```

जर युजर काही कारणास्तव इमेज पाहू शकत नसेल (स्लो कनेक्शनमुळे, src स्पेशल तामध्ये त्रुटी किंवा युजर स्क्रीन रीडर वापरत असेल तर) alt स्पेशल ता इमेजसाठी पर्यायी इनफार्मेशन प्रदान करते.

HTML प्रतिमा - प्रतिमेची उंची आणि रुंदी सेट करा

उंची आणि रुंदीचे प्रॉपर्टीज प्रतिमेची उंची आणि रुंदी स्पेसिफाइड करण्यासाठी वापरले जातात. स्पेशल ता व्हॅल्यू डीफॉल्टनुसार पिक्सेलमध्ये स्पेसिफाइड केली जातात:

```

```

टीप: प्रतिमेसाठी उंची आणि रुंदी या दोन्ही प्रॉपर्टीज स्पेसिफाइड करणे ही एक चांगली सराव आहे. या स्पेशल ता सेट केल्या असल्यास, पृष्ठ लोड झाल्यावर प्रतिमेसाठी आवश्यक जागा आरक्षित केली जाते. तथापि, या गुणधर्मांशिवाय, ब्राउझरला प्रतिमेचा आकार माहित नाही. रिझल्ट असा होईल की लोडिंग दरम्यान पृष्ठ लेआउट बदलेल (प्रतिमा लोड होत असताना).

HTML फाइलमध्ये दहा प्रतिमा असल्यास - पृष्ठ उजवीकडे प्रदर्शित करण्यासाठी अकरा फाइल्स आवश्यक आहेत. प्रतिमा लोड होण्यास वेळ लागतो, म्हणून माझा सर्वोत्तम सल्ला आहे: प्रतिमा काळजीपूर्वक वापरा.

जेव्हा एखादे वेब पृष्ठ लोड केले जाते, तेव्हा तो ब्राउझर असतो, त्या क्षणी, तो प्रत्यक्षात वेब सर्व्हरवरून प्रतिमा मिळवतो आणि पृष्ठामध्ये समाविष्ट करतो. (पिक्चर 12) म्हणून, वेब पृष्ठाच्या संबंधात प्रतिमा प्रत्यक्षात त्याच ठिकाणी राहतील याची खात्री करा, अन्यथा तुमच्या अभ्यागतांना तुटलेली लिंक आयकॉन मिळेल. ब्राउझर प्रतिमा शोधू शकत नसल्यास तुटलेली लिंक चिन्ह दर्शविले जाते.

```
<html>
```

```
<body>
```

```
<h2>Friendship Card</h2>
```

```

```

```
</body>
```

```
</html>
```

### HTML टेबल

सारण्या <table> टॅगसह परिभाषित केल्या आहेत.

टेबल रो मध्ये विपार्ट ली आहे (<tr> टॅगसह), आणि प्रत्येक रो डेटा सेलमध्ये विपार्ट ली आहे (<td> टॅगसह). td चा अर्थ "टेबल डेटा" आहे आणि डेटा सेलची कंटेंट धारण करते. <td> टॅगमध्ये टेक्स्ट , दुवे, प्रतिमा, लिस्ट , फॉर्म, इतर सारण्या इत्यादी असू शकतात.

### टेबल उदाहरण

```
<table border="1">
```

```
<tr>
```

```
<td>row 1, cell 1</td>
```

```
<td>row 1, cell 2</td>
```

```
</tr>
```

```
<tr>
```

```
<td>row 2, cell 1</td>
```

```
<td>row 2, cell 2</td>
```

```
</tr>
```

```
</table>
```

How the HTML code above looks in a browser:

row 1, cell 1 row 1, cell 2

row 2, cell 1 row 2, cell 2

### HTML सारण्या आणि बॉर्डर प्रॉपर्टीज

तुम्ही बॉर्डर स्पेशल ता स्पेसिफाइड न केल्यास, टेबल सीमांशिवाय प्रदर्शित केले जाईल. कधीकधी हे सबयुक्त ठरू शकते, परंतु बहुतेक वेळा, आम्ही बॉर्डर दर्शवू इच्छितो.

बॉर्डरसह टेबल प्रदर्शित करण्यासाठी, बॉर्डर स्पेशल ता स्पेसिफाइड करा:

```
<table border="1">
```

```
<tr>
```

```
<td>Row 1, cell 1</td>
```

```
<td>Row 1, cell 2</td>
```

```
</tr>
```

```
</table>
```

### HTML टेबल हेडर

टेबलमधील हेडर इनफार्मेशन <th> टॅगसह परिभाषित केली जाते.

Fig 12



सर्व प्रमुख ब्राउझर <th> एलिमेंटतील टेक्स्ट ठळक आणि मध्यभागी प्रदर्शित करतात.

```
<table border="1">
```

```
<tr>
```

```
<th>Header 1</th>
```

```
<th>Header 2</th>
```

```
</tr>
```

```
<tr>
```

```
<td>row 1, cell 1</td>
```

```
<td>row 1, cell 2</td>
```

```
</tr>
```

```
<tr>
```

```
<td>row 2, cell 1</td>
```

```
<td>row 2, cell 2</td>
```

```
</tr>
```

```
</table>
```

वरील HTML कोड तुमच्या ब्राउझरमध्ये कसा दिसतो:

हेडर १	हेडर २
रो १, सेल १	रो १, सेल २
रो २, सेल १	रो २, सेल २

#### HTML Unordered Lists

अक्रमित यादी <ul> टॅगने सुरू होते. प्रत्येक लिस्ट आयटम <li> टॅगने सुरू होतो. लिस्ट आयटम बुलेट (सामान्यतः लहान काळी वर्तुळे) सह चिन्हांकित आहेत.

```
<ul>
```

```
<li>Coffee</li>
```

```
<li>Milk</li>
```

```
</ul>
```

ब्राउझरमध्ये वरील HTML कोड कसा दिसतो:

1 कॉफी 2 दूध

#### HTML Ordered Lists

ऑर्डर केलेली यादी <ol> टॅगने सुरू होते. प्रत्येक लिस्ट आयटम <li> टॅगने सुरू होतो. लिस्ट आयटम नंबर सह चिन्हांकित आहेत.

```
<ol>
```

```
<li>Coffee</li>
```

```
<li>Milk</li>
```

</ol>

ब्राउझरमध्ये वरील HTML कोड कसा दिसतो:

1 कॉफी 2 दूध

### HTML व्याख्या याद्या

परिभाषा लिस्ट म्हणजे प्रत्येक आयटमच्या कॅरक्टर नासह आयटमची लिस्ट .

<dl> टॅग परिभाषा लिस्ट परिभाषित करते.

<dl> टॅगचा वापर <dt> (यादीतील आयटम परिभाषित करते) आणि <dd> (लिस्ट मधील आयटमचे डिस्क्रिप्शन करते) सह संयोगाने केला जातो:

<dl>

<dt>Coffee</dt>

<dd>- black hot drink</dd>

<dt>Milk</dt>

<dd>- white cold drink</dd>

</dl>

ब्राउझरमध्ये वरील HTML कोड कसा दिसतो:

कॉफी - काळा गरम पेय

दूध- पांढरे थंड पेय

**टीप :** लिस्ट आयटममध्ये तुम्ही टेक्स्ट , ओळ खंडित, प्रतिमा, दुवे, इतर लिस्ट इत्यादी टाकू शकता.

### HTML लिस्ट टॅग

टॅग करा	डिस्क्रिप्शन
<ol>	ऑर्डर केलेली लिस्ट परिभाषित करते
<ul>	अक्रमित लिस्ट परिभाषित करते
<ते>	लिस्ट आयटम परिभाषित करते
<dl>	व्याख्या लिस्ट परिभाषित करते
<dt>	परिभाषा लिस्ट मधील आयटम परिभाषित करते
<dd>	परिभाषा लिस्ट मधील आयटमचे डिस्क्रिप्शन परिभाषित करते

HTML एलिमेंटना <div> आणि <span> सह एकत्रित केले जाऊ शकते

### HTML ब्लॉक एलिमेंट

बहुतेक एचटीएमएल एलिमेंट ब्लॉक स्तर एलिमेंट किंवा इनलाइन एलिमेंट म्हणून परिभाषित केले जातात.

ब्लॉक लेव्हल एलिमेंट सामान्यतः ब्राउझरमध्ये प्रदर्शित झाल्यावर नवीन ओळीने सुरू होतात (आणि समाप्त होतात).

उदाहरणे: <h1>, <p>, <ul>, <table>

HTML इनलाइन एलिमेंट

इनलाइन एलिमेंट सामान्यतः नवीन ओळ सुरू न करता प्रदर्शित केले जातात. उदाहरणे: <b>, <td>, <a>, <img>

HTML <div> एलिमेंट

HTML <div> एलिमेंट हा एक ब्लॉक लेव्हल एलिमेंट आहे जो इतर HTML एलिमेंटना ग्रुप बद्ध करण्यासाठी कंटेनर म्हणून वापरला जाऊ शकतो.

<div> एलिमेंटला स्पेशल अर्थ नाही. ते वगळता, तो ब्लॉक लेव्हल एलिमेंट असल्यामुळे, ब्राउझर त्याच्या आधी आणि नंतर एक लाइन ब्रेक प्रदर्शित करेल.

CSS सह एकत्र वापरल्यास, <div> एलिमेंट कंटेंट च्या मोठ्या ब्लॉक्सवर स्टाइल प्रॉपर्टीज सेट करण्यासाठी वापरला जाऊ शकतो.

<div> एलिमेंटचा आणखी एक सामान्य वापर, डॉक्युमेंट्स सिन्टेक्स साठी आहे. हे टेबल वापरून लेआउट परिभाषित करण्याच्या "जुन्या मेथड्स" ची जागा घेते. सारण्या वापरणे हा <table> एलिमेंटचा योग्य वापर नाही. <table> एलिमेंटचा उद्देश टेबल डेटा प्रदर्शित करणे आहे.

HTML <span> एलिमेंट

HTML <span> एलिमेंट हा एक इनलाइन एलिमेंट आहे जो टेक्स्ट साठी कंटेनर म्हणून वापरला जाऊ शकतो.

<span> एलिमेंटला स्पेशल अर्थ नाही.

CSS सह एकत्र वापरल्यास, <span> एलिमेंट टेक्स्ट च्या भागांमध्ये स्टाइल प्रॉपर्टीज सेट करण्यासाठी वापरला जाऊ शकतो.

एचटीएमएल ग्रुपिंग टॅग

टॅग करा	डिस्क्रिप्शन
<div>	div परिभाषित करते
<span>	स्पॅन परिभाषित करते

### HTML लेआउट

वेबसाइट लेआउट

बऱ्याच वेबसाइट्सनी त्यांची कंटेंट मल्टिपल स्तंभांमध्ये ठेवली आहे (नियतकालिक किंवा करंट पत्रासारखे फॉर्मॅटिंग).

<div> किंवा <table> एलिमेंट वापरून अनेक कॉलम तयार केले जातात. CSS चा वापर एलिमेंटना पोजिशन देण्यासाठी किंवा पेजसाठी बॅकग्राउंड किंवा रंगीत देखावा तयार करण्यासाठी केला जातो.

HTML सिन्टेक्स - <div> एलिमेंट वापरणे

div एलिमेंट हा ब्लॉक लेव्हल एलिमेंट आहे जो एचटीएमएल एलिमेंटना ग्रुप बद्ध करण्यासाठी वापरला जातो.

खालील उदाहरणात मागील उदाहरणाप्रमाणेच रिझल्ट तयार करून, मल्टिपल कॉलम लेआउट तयार करण्यासाठी पाच div एलिमेंट वापरतात:

उदाहरण

```
<div>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<div id="container" style="width:500px">
```

```
<div id="header" style="background-color:#FFA500;">
```

```
<h1 style="margin-bottom:0;">Main Title of Web Page</
```

```
h1></div>
```

```
<div id="menu" style="background-color:#FFD700;height:200px;width:100px;float:left;">
```

```
<b>Menu</b><br />
```

```
HTML<br />
```

```
CSS<br />
```

```
JavaScript</div>
```

```
<div id="content" style="background-color:#EEEEEE;height:200px;width:400px;float:left;">
```

```
Content goes here</div>
```

```
<div id="footer" style="background-color:#FFA500;clear:both;text-align:center;">
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

रिझल्ट (पिक्चर 13)

Fig 13

Main Title of Web Page

Menu

HTML

CSS

JavaScript

Content goes here

## HTML फॉर्म

HTML फॉर्मचा वापर सर्व्हरला डेटा पास करण्यासाठी केला जातो.

फॉर्ममध्ये टेक्स्ट फील्ड, चेकबॉक्स, रेडिओ-बटन्स, सबमिट बटणे आणि बरेच काही यासारखे इनपुट एलिमेंट असू शकतात. फॉर्ममध्ये इलेक्ट्रिक याद्या, टेक्स्ट, फील्डसेट, आख्यायिका आणि लेबल एलिमेंट देखील असू शकतात.

HTML फॉर्म तयार करण्यासाठी <form> टॅग वापरला जातो:

```
<फॉर्म>
```

इनपुट एलिमेंट .

```
</form>
```

HTML फॉर्म - इनपुट एलिमेंट

सर्वात महत्वाचा फॉर्म एलिमेंट इनपुट एलिमेंट आहे.

इनपुट एलिमेंट युजर इनफार्मेशन निवडण्यासाठी वापरला जातो.

इनपुट एलिमेंट टाइप गुणधर्मावर अवलंबून, अनेक प्रकारे बदलू शकतो. इनपुट एलिमेंट टेक्स्ट फील्ड, चेकबॉक्स, पासवर्ड, रेडिओ बटण, सबमिट बटण आणि बरेच काही असू शकतात.

सर्वात जास्त वापरलेले इनपुट टाइप खाली डिस्क्रिप्शन केले आहेत.

टेक्स्ट फील्ड

<input type="text" /> defines a one-line input field that a user can enter text into:

```
<form>
```

First name: <input type="text" name="firstname" /><br />

Last name: <input type="text" name="lastname" />

```
</form>
```

How the HTML code above looks in a browser:

First name:

Last name:

टीप: फॉर्म स्वतःच व्ह मान नाही. हे देखील लक्षात ठेवा की टेक्स्ट फील्डची डीफॉल्ट रुंदी 20 कॅरक्टर आहे.

### पासवर्ड फील्ड

`<input type="password" />` defines a password field:

`<form>`

Password: `<input type="password" name="pwd" />`

`</form>`

How the HTML code above looks in a browser:

Password:

टीप: पासवर्ड फील्डमधील कॅरक्टर मास केलेले आहेत (तारक किंवा मंडळे म्हणून दर्शविलेले)

### रेडिओ बटणे

`<input type="radio" />` defines a radio button. Radio buttons let a user select ONLY ONE of a limited number of

choices:

`<form>`

`<input type="radio" name="sex" value="male" />` Male  
`<br/>`

`<input type="radio" name="sex" value="female" />` Female

`</form>`

How the HTML code above looks in a browser:

Male

Female

### चेकबॉक्सेस

`<input type="checkbox" />` defines a checkbox.

Checkboxes let a user select ONE or MORE options of a limited number of choices.

`<form>`

`<input type="checkbox" name="vehicle" value="Bike" />`

I have a bike  
`<br />`

`<input type="checkbox" name="vehicle" value="Car" />` I

have a car

`</form>`

How the HTML code above looks in a browser:

I have a bike

I have a car

### सबमिट बटण

`<input type="submit" />` सबमिट बटण परिभाषित करते.

सर्व्हरला फॉर्म डेटा पाठवण्यासाठी सबमिट बटण वापरले जाते. डेटा फॉर्मच्या क्रिया गुणधर्मांमध्ये स्पेसिफाइड केलेल्या पेजवर पाठविला जातो. क्रिया स्पेशल ता मध्ये परिभाषित केलेली फाइल सामान्यतः प्राप्त इनपुटसह काहीतरी करते.

`<form name="input" action="html_form_action.asp" method="get">`

Username: `<input type="text" name="user" />`

`<input type="submit" value="Submit" />`

`</form>`

How the HTML code above looks in a browser:

Username:  Submit

तुम्ही वरील टेक्स्ट फील्डमध्ये काही अक्षरे टाईप केल्यास आणि "सबमिट" बटणावर क्लिक केल्यास, ब्राउझर तुमचे इनपुट "html\_form\_action.asp" नावाच्या पेजवर पाठवेल. पृष्ठ आपल्याला प्राप्त इनपुट दर्शवेल.

HTML Iframes

### iframe जोडण्यासाठी सिंटॅक्स:

`<iframe src="URL"></iframe>`

URL वेगळ्या पृष्ठाच्या स्थानाकडे निर्देश करते.

Iframe - उंची आणि रुंदी सेट करा

iframe ची उंची आणि रुंदी स्पेसिफाइड करण्यासाठी उंची आणि रुंदीचे प्रॉपर्टीज वापरले जातात.

स्पेशल ता व्हॅल्यू डीफॉल्टनुसार पिक्सेलमध्ये स्पेसिफाइड केली जातात, परंतु ती टक्केवारीत देखील असू शकतात (जसे "80%").

### उदाहरण

`<!DOCTYPE html>`

`<html>`

`<body>`

`<iframe src="demo_iframe.htm" width="200"`

`height="200"></iframe>`

`</body>`



</html>

ते पिक्चर 14 मध्ये दाखवल्याप्रमाणे दिसेल.

Iframe - बॉर्डर काढा

फ्रेमबॉर्डर स्पेशल ता iframe भोवती बॉर्डर प्रदर्शित करायची की नाही हे स्पेसिफाइड करते.

बॉर्डर काढण्यासाठी स्पेशल ता व्हॅल्यू "0" वर सेट करा:

उदाहरण

```
<iframe src="demo_iframe.htm" frameborder="0"></iframe>
```

दुव्यासाठी टारगेट म्हणून iframe वापरा

लिंकसाठी टारगेट फ्रेम म्हणून iframe वापरले जाऊ शकते.

लिंकच्या टारगेट गुणधर्माने iframe च्या नावाच्या स्पेशल ताचा रेफरेन्स घेणे आवश्यक आहे:

Example

```
<iframe src="demo_iframe.htm" frameborder="0"></
```

iframe>

Use iframe as a Target for a Link

An iframe can be used as the target frame for a link.

The target attribute of a link must refer to the name attribute of the iframe:

**Example**

```
<iframe src="demo_iframe.htm" name="iframe_a"></
```

iframe>

```
<p><a href="http://www.yahoo.com"
```

```
target="iframe_a">yahoo.com</a></p>
```

HTML iframe टॅग

टॅग करा	डिस्क्रिप्शन
<iframe>	इनलाइन सब विंडो (फ्रेम) परिभाषित करते

**HTML रंग**

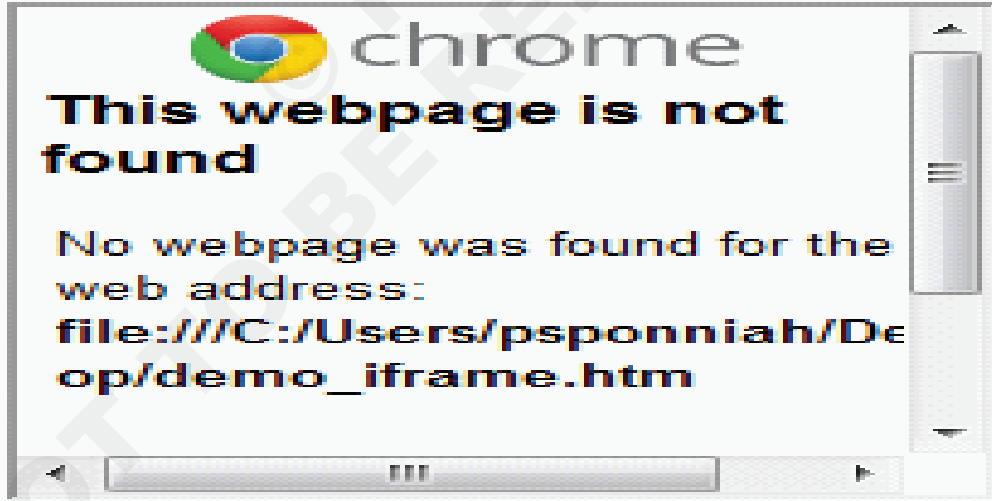
रंग व्हॅल्यू

लाल, हिरवा आणि निळा रंग व्हॅल्यू (RGB) च्या संयोजनासाठी हेक्साडेसिमल नोटेशन (HEX) वापरून HTML रंग परिभाषित केले जातात.

प्रकाश स्रोतांपैकी एकाला दिले जाणारे सर्वात कमी व्हॅल्यू 0 (HEX: 00 मध्ये) आहे. सर्वोच्च व्हॅल्यू 255 (HEX: FF मध्ये) आहे.

HEX व्हॅल्यू # चिन्हाने सुरू होणारी, दोन-अंकी नंबर च्या 3 जोड्या म्हणून स्पेसिफाइड केल्या आहेत.

Fig 14



**रंग व्हॅल्यू (पिक्चर 15)**

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p style="background-color:#FFFF00">
```

Color set by using hex value

```
</p>
```

```
<p style="background-color:rgb(255,255,0)">
```

Color set by using rgb value

```
</p>
```

```
<p style="background-color:yellow">
```










Color set by using color name

```
</p>
```

```
</body>
```

```
</html>
```

Fig 15

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

## रिझल्ट

हेक्स व्हॅल्यु वापरून रंग सेट

rgb व्हॅल्यु वापरून रंग सेट

रंग नाव वापरून रंग सेट

## DHTML

HTML स्क्रिप्ट एलिमेंट

<script> टॅग क्लायंट-साइड स्क्रिप्ट परिभाषित करण्यासाठी वापरला जातो, जसे की JavaScript.

स्क्रिप्ट एलिमेंटमध्ये एकतर स्क्रिप्टिंग स्टेटमेंट असतात किंवा ती src स्पेशल ताद्वारे एक्सटर्नल स्क्रिप्ट फाइलकडे निर्देश करते.

आवश्यक टाइप स्पेशल ता स्क्रिप्टचा MIME टाइप स्पेसिफाइड करते.

JavaScript चे सामान्य यूजेस म्हणजे इमेज मॅनिपुलेशन, फॉर्म व्हॅलिडेशन आणि कंटेंट चे डायनॅमिक बदल.

खालील स्क्रिप्ट हॅलो वर्ल्ड लिहिते! HTML आउटपुटवर:

## उदाहरण

```
<script type="text/javascript">
```

```
document.write("Hello World!")
```

```
</script>
```

HTML noscript एलिमेंट

<noscript> टॅग वापरकर्त्यासाठी पर्यायी कंटेंट प्रदान करण्यासाठी वापरला जातो ज्यांनी त्यांच्या ब्राउझरमध्ये स्क्रिप्ट अक्षम केल्या आहेत किंवा ब्राउझर आहे जो क्लायंट-साइड स्क्रिप्टिंगला सपोर्ट देत नाही.

noscript एलिमेंटमध्ये ते सर्व एलिमेंट असू शकतात जे तुम्हाला सामान्य HTML पृष्ठाच्या मुख्य भागामध्ये सापडतात.

स्क्रिप्ट समर्थित नसल्यास किंवा यूजर च्या ब्राउझरमध्ये अक्षम असल्यासच noscript एलिमेंटतील कंटेंट प्रदर्शित केली जाईल:

## उदाहरण

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<script type="text/javascript">
```

```
document.write("Hello World!")
```

```
</script>
```

```
<noscript>Sorry, your browser does not support
```

```
JavaScript!</noscript>
```

```
<p>A browser without support for JavaScript will show the text in the noscript element.</p>
```

```
</body>
```

```
</html>
```

## रिझल्ट

हॅलो वर्ल्ड!

JavaScript साठी सपोर्ट नसलेला ब्राउझर noscript एलिमेंटतील टेक्स्ट दर्शविल.

## HTML स्क्रिप्ट टॅग

टॅग करा	डिस्क्रिप्शन
<script>	क्लायंट-साइड स्क्रिप्ट परिभाषित करते
<noscript>	क्लायंट-साइड स्क्रिप्टला सपोर्ट न करणाऱ्या वापरकर्त्यासाठी पर्यायी कंटेंट परिभाषित करते

## HTML Entities

काही कॅरक्टर HTML मध्ये राखीव आहेत.

तुमच्या टेक्स्ट मध्ये (<) पेक्षा कमी किंवा (>) पेक्षा जास्त सिम्बॉल्स वापरणे शक्य नाही, कारण ब्राउझर त्यांना टॅगसह मिसळेल.

आरक्षित कॅरक्टर प्रदर्शित करण्यासाठी, आम्ही HTML सोर्स कोडमध्ये कॅरक्टर एलिमेंट वापरणे आवश्यक आहे.

कॅरक्टर अस्तित्व असे दिसते:

&#entity\_name;

किंवा

&#entity\_number;

Non-breaking Space

HTML मध्ये वापरलेली एक सामान्य कॅरक्टर संस्था म्हणजे नॉनब्रेकिंग स्पेस (&nbsp;).

ब्राउझर नेहमी एचटीएमएल पेजेसमधील स्पेस ट्रंक करतात. तुम्ही तुमच्या टेक्स्ट मध्ये 10 स्पेसेस लिहिल्यास, पृष्ठ प्रदर्शित करण्यापूर्वी ब्राउझर त्यापैकी 9 काढून टाकेल. तुमच्या मजकुरामध्ये स्पेस जोडण्यासाठी, तुम्ही &nbsp; कॅरक्टर अस्तित्व.

## HTML Useful Character Entities

टीप: एलिमेंटची नावे केस सेन्सिटिव्ह आहेत!

HTML एकसमान संसाधन लोकेटर

URL वेब पत्त्यासाठी दुसरा शब्द आहे.

रिझल्ट	डिस्क्रिप्शन	अस्तित्वाचे नाव	अस्तित्व क्रमांक
	न तोडणारा जागा	&nbsp;	&#160;
<	च्या पेक्षा कमी	&lt;	&#60;
>	या पेक्षा मोठे	&gt;	&#62;
&	अँपरसँड	&amp;	&#38;
¢	टक्के	&सेंट;	&#162;
£	पौंड	&पाउंड;	&#163;
¥	तर	&तर;	&#165;
•	युरो	&युरो;	&#8364;
§	विपार्ट	&पंथ;	&#167;
©	कॉपीराइट	&कॉपी;	&#169;
®	नोंदणीकृत ट्रेडमार्क	&reg;	&#174;
™	ट्रेडमार्क	&व्यापार;	&#8482;

URL शब्दांनी बनलेली असू शकते, जसे की "w3schools.com", किंवा इंटरनेट प्रोटोकॉल (IP) एड्रेस : 192.68.20.50. सर्किंग करताना बहुतेक लोक वेबसाइटचे नाव टाकतात, कारण नंबर पेक्षा नावे लक्षात ठेवणे सोपे असते.

## URL - युनिफॉर्म रिसोर्स लोकेटर

जेव्हा तुम्ही HTML पृष्ठावरील लिंकवर क्लिक करता, तेव्हा बेसिक <a> टॅग वलड वाइड वेबवरील पत्त्याकडे निर्देश करतो.

वलड वाइड वेबवरील डॉक्युमेंट्स (किंवा इतर डेटा) संबोधित करण्यासाठी युनिफॉर्म रिसोर्स लोकेटर (URL) वापरला जातो.

## स्पष्टीकरण

- योजना -इंटरनेट सेवेचा टाइप परिभाषित करते. सर्वात सामान्य टाइप म्हणजे http
- होस्ट -डोमेन होस्ट परिभाषित करते (http साठी डीफॉल्ट होस्ट www आहे)
- डोमेन -इंटरनेट डोमेन नाव परिभाषित करते, जसे की w3schools.com
- पोर्ट -होस्टवरील पोर्ट क्रमांक परिभाषित करते (http साठी डीफॉल्ट पोर्ट क्रमांक 80 आहे)
- पाथ -सर्व्हरवर पाथ परिभाषित करते (वगळल्यास, डॉक्युमेंट्स वेब साइटच्या मूळ डिरेक्ट्री स्टोर करणे आवश्यक आहे)
- फाईलचे नाव -डॉक्युमेंट्स /संसाधनाचे नाव परिभाषित करते

## Common URL Schemes

खालील तक्त्यामध्ये काही सामान्य योजना आहेत:

योजना	साठी थोडक्यात....	स्कीम कोणत्या पेजसाठी वापरली जाईल...
http	हायपरटेक्स्ट ट्रान्सफर प्रोटोकॉल	सामान्य वेब पेजेज http:// ने सुरू होतात. एनक्रिप्ट केलेले नाही
https	सुरक्षित हायपरटेक्स्ट	ट्रान्सफर प्रोटोकॉल वेब पेजेज सुरक्षित करा. देवाणघेवाण केलेली सर्व इनफार्मेशन एनक्रिप्टेड आहे
एफटीपी	फाइल ट्रान्सफर प्रोटोकॉल	वेबसाइटवर फाइल्स डाउनलोड किंवा अपलोड करण्यासाठी. डोमेन देखरेखीसाठी सबयुक्त
फाइल		तुमच्या संगणकावरील फाइल

## URL एन्कोडिंग

URL फक्त ASCII अक्षर-संच वापरून इंटरनेटवर पाठवल्या जाऊ शकतात.

URL मध्ये बऱ्याचदा ASCII संचाच्या बाहेरील कॅरक्टर असतात, URL ला वैध ASCII फॉर्मॅटमध्ये रूपांतरित करावे लागते.

URL एन्कोडिंग ASCII नसलेल्या कॅरक्टर ना "%" आणि त्यानंतर दोन हेक्साडेसिमल अंकांसह बदलते.

URL मध्ये स्पेस असू शकत नाही. URL एन्कोडिंग साधारणपणे + चिन्हाने स्पेस बदलते.

## XML रचना

डॉक्युमेंट्स च्या भागांसह XML रचना, प्रस्तावना आणि एक साधे XML उदाहरण डॉक्युमेंट्स प्रदान करते.

### डॉक्युमेंट्स पार्ट

- प्रोलॉग
- डॉक्युमेंट्स एलिमेंट (मूळ एलिमेंट)

### प्रोलॉग

HTML मधील शीर्षलेखाच्या समतुल्य प्रस्तावनामध्ये पुढील गोष्टींचा समावेश असू शकतो: • XML घोषणा (पर्यायी) जसे की: `<?xml version="1.0"?>`

- DTD किंवा एकाचा रेफरन्स (पर्यायी). एक्सटेर्नल डीटीडी फाइलचा एक उदाहरण रेफरन्स :

```
<!DOCTYPE LANGLIST SYSTEM "langlist.dtd">
```

- प्रक्रिया सूचना - एक उदाहरण प्रक्रिया निर्देश ज्यामुळे स्टाइल पत्रकाद्वारे निर्धारित केली जाते:

```
<?xml-stylesheet type="text/css"
```

```
href="xmlstyle.css"?>
```

XML उदाहरण

म्हणून संपूर्णपणे तयार केलेला XML डॉक्युमेंट्स असे दिसू शकतो:

```
<?xml version="1.0"?>
```

```
<LAND>
```

```
<FOREST>
```

```
<TREE>Oak</TREE>
```

```
<TREE>Pine</TREE>
```

```
<TREE>Maple</TREE>
```

```
</FOREST>
```

```
<MEADOW>
```

```
<GRASS>Bluegrass</GRASS>
```

```
<GRASS>Fescue</GRASS>
```

```
<GRASS>Rye</GRASS>
```

```
</MEADOW>
```

```
</LAND>
```

वरील लैंड एलिमेंट मूळ एलिमेंट आहे.

### रिझल्ट

ओक पाइन मॅपल ब्लूग्रास फेस्क्यू राय

### वेब एलिमेंट

वेब पेज, इनफार्मेशन संच म्हणून, असंख्य टाइप ची इनफार्मेशन असू शकते, जी अंतिम यूजर द्वारे पाहण्यास, ऐकण्यास किंवा संवाद साधण्यास सक्षम आहे.

### वेब होस्टिंग

वेब होस्टिंग सर्विस ही एक टाइप ची इंटरनेट होस्टिंग सर्विस आहे जी व्यक्ती आणि संस्थांना त्यांच्या वेबसाइटवर वर्ल्ड वाइड वेबद्वारे एक्सेस योग्य बनविण्यास अनुमती देते. वेब होस्ट या कंपनी आहेत ज्या सर्व्हर (संगणन) मालकीच्या किंवा क्लायंटच्या वापरासाठी भाडेतत्वावर जागा प्रदान करतात, तसेच इंटरनेट कनेक्टिव्हिटी प्रदान करतात, स्पेशल तः डेटा सेंटरमध्ये. वेब होस्ट त्यांच्या डेटा सेंटरमध्ये कोलोकेशन नावाच्या इतर सर्व्हरसाठी डेटा सेंटर स्पेस आणि इंटरनेटशी कनेक्टिव्हिटी देखील प्रदान करू शकतात.

## CMS आणि वेब ऑथरिंग टूल्सचा परिचय (Introduction to CMS and web authoring tools)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कंटेंट मॅनेजमेन्ट सिस्टिम चा अर्थ स्पष्ट करा
- काही लोकप्रिय CMS ची यादी करा.
- कंपोजर वेब ऑथरिंग टूलचा वापर आणि मुख्य वैशिष्ट्ये स्पष्ट करा.

### CMS ची व्याख्या

CMS चा अर्थ " कंटेंट मॅनेजमेन्ट सिस्टिम " आहे. CMS हे एक सॉफ्टवेअर साधन आहे जे तुम्हाला कंटेंट तयार करण्यास, एडिट करण्यास आणि प्रकाशित करण्यास अनुमती देते.

### CMS चे डिस्क्रिप्शन

वेबपेज कंटेंट तयार करण्यासाठी आणि सुधारण्यासाठी अंतर्ज्ञानी युजर इंटरफेस प्रदान करणे हे CMS चे ध्येय आहे. प्रत्येक CMS एक वेब प्रकाशन साधन देखील प्रदान करते जे एक किंवा अधिक वापरकर्त्यांना वेबवर थेट अपडेट प्रकाशित करण्यास अनुमती देते. एडिटिंग एलिमेंटला कंटेंट मॅनेजमेंट ॲप्लिकेशन (CMA) म्हणतात, तर प्रकाशन साधनाला कंटेंट डिलिव्हरी ॲप्लिकेशन (CDA) म्हणतात. वेब डेव्हलपमेंट प्रक्रिया सुव्यवस्थित करण्यासाठी हे दोन एलिमेंट CMS मध्ये एकत्रित केले आहेत.

वेब प्रकाशनाच्या संदर्भात, कंटेंट साधा टेक्स्ट, फोटो, संगीत, व्हिडिओ, डॉक्युमेंट्स किंवा आपण विचार करू शकता अशा कोणत्याही गोष्टी असू शकतात. डॉक्युमेंट्स आणि लोकल संगणक फाइल्स मॅनेज करण्यासाठी सुरुवातीच्या CMS सॉफ्टवेअरचा वापर केला जात असताना, बहुतेक CMS सिस्टिम आता केवळ वेबवरील कंटेंट मॅनेज करण्यासाठी डिझाइन केल्या आहेत. CMS वापरण्याचा एक मोठा फायदा म्हणजे त्याला मॅनेज करण्यासाठी जवळजवळ कोणतेही तांत्रिक कौशल्य किंवा ज्ञान आवश्यक नसते.

कंटेंट मॅनेजमेन्ट सिस्टिम इंस्टाल करण्यायोग्य एप्लिकेशन आणि वेब आधारित युजर इंटरफेस म्हणून सबलब्ध आहेत. वेब इंटरफेसचा वापर वेबसाइट अपडेट करण्याची प्रक्रिया सुलभ करतो. याव्यतिरिक्त, सर्व वापरकर्त्यांकडे त्यांची कंटेंट मॅनेज करण्यासाठी नवीनतम साधने आहेत याची खात्री करून, बहुतेक वेब-आधारित CMSes ऑटोमॅटिक पणे अपडेट केले जातात.

CMS वापरण्याचा एक मोठा फायदा म्हणजे त्याला मॅनेज करण्यासाठी जवळजवळ कोणतेही तांत्रिक कौशल्य किंवा ज्ञान आवश्यक नसते. CMS तुमची सर्व कंटेंट मॅनेज करत असल्याने, तुम्हाला याची गरज नाही.

आज अनेक वेब-आधारित CMS साधने सबलब्ध आहेत. खालील काही सर्वात लोकप्रिय आहेत:

- वर्डप्रेस - टेम्पलेट-आधारित वेबसाइट किंवा ब्लॉग तयार करण्यासाठी डिझाइन केलेले विनाव्हॅल्यु वेब सॉफ्टवेअर

- ब्लॉगर - Google चे ब्लॉगिंग साधन स्पेशल तः ब्लॉग राखण्यासाठी डिझाइन केलेले आहे
- जूमला - एक लवचिक वेब प्रकाशन साधन जे कस्टम डेटाबेस आणि विस्तारांना सपोर्ट देते
- Drupal - एक ओपन सोर्स प्लॅटफॉर्म ज्याचा वापर समुदाय-आधारित साइट्स विकसित करण्यासाठी केला जातो
- Weebly - साध्या वैयक्तिक आणि व्यावसायिक वेबसाइट्स तयार करण्यासाठी एक वेब-आधारित प्लॅटफॉर्म • Wix - अत्यंत कस्टम करण्यायोग्य वेबसाइट तयार करण्यासाठी वेब प्रकाशन साधनांचा संग्रह.

### WYSIWYG वेब ऑथरिंग टूल्सचा परिचय

WYSIWYG चा अर्थ आहे "तुम्ही जे पाहता ते तुम्हाला मिळते". अशा एडिटर मध्ये तुम्ही तुमच्या डॉक्युमेंट्स चा सोर्स कोड थेट एडिट करत नाही, तर त्याचे प्रेसेंटेशन अंतिम डॉक्युमेंट्स त दिसेल. त्यामुळे कोडचे ब्लॉक्स मॅन्युअली लिहिण्याऐवजी (जसे तुम्ही ते वर्ड किंवा नोटपॅडमध्ये कराल), तुम्ही एडिटर विंडो वापरून डिझाईन एलिमेंटमध्ये फेरफार करता. याचा अर्थ असा आहे की डॉक्युमेंट्स किंवा प्रतिमा तयार होत असताना तुम्ही अंतिम परिणामासारखे काहीतरी पाहता.

अनेक वापरण्यास-सोपे WYSIWYG प्रोग्राम्स आहेत ज्यामध्ये एक जटिल आणि पूर्णपणे कार्यक्षम वेबसाइट तयार करण्यासाठी आवश्यक असलेली सर्व साधने आहेत, अगदी नवशिक्यांसाठीही. या साइट्सना HTML कोडसह काम करण्याचे पर्याय आहेत मग ते डिझाइन किंवा एडिटिंग असो. HTML संपादकासह वेबसाइट तयार करणे सोपे आहे, कारण सॉफ्टवेअर डेव्हलपर अशी साधने जोडणे सुरू ठेवतात जे तुम्हाला स्टाइल सह एडवॉन्स वैशिष्ट्ये विकसित करू देतात. अनेक WYSIWYG वेब ऑथरिंग टूल्स डायनॅमिक एचटीएमएल किंवा इतर अनेक वैशिष्ट्ये एका मोहक आणि सुसंगत डिझाइनसह साइटमध्ये एकत्रित करण्यासाठी एडवॉन्स वैशिष्ट्ये ऑफर करा.

### कंपोजरचा(Kompozer) परिचय

KompoZer ही एक संपूर्ण वेब ऑथरिंग सिस्टिम आहे जी Microsoft FrontPage, Adobe DreamWeaver आणि इतर हाय एंड प्रोग्राम्समध्ये आढळणारी वेब फाइल मॅनेजमेन्ट आणि वापरण्यास-सुलभ WYSIWYG वेब पृष्ठ एडिटिंग क्षमता एकत्र करते. kompozer सह तुम्ही वेब पेजेस

तयार करू शकता आणि HTML चे कोणतेही तांत्रिक कौशल्य किंवा ज्ञान नसलेली वेबसाइट मॅनेज करू शकता.

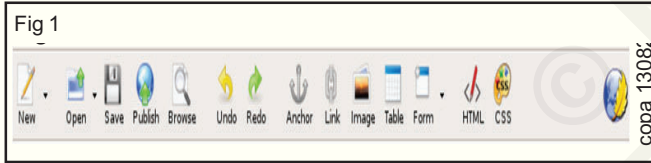
Kompozer चे एक सबयुक्त वैशिष्ट्य म्हणजे तुम्ही तुमची वेब पेज वेब होस्टिंग सर्व्हरवर प्रकाशित करण्यासाठी वापरू शकता. तुम्हाला फक्त तुमचे FTP खाते युजर नाव आणि पासवर्ड तसेच साइट URL माहित असणे आवश्यक आहे आणि Kompozer लॉग इन करेल आणि सर्व्हरवर तुमची वेब पेजेज ठेवेल.

KompoZer च्या फाइल मेनूमध्ये ते वापरताना शक्य असलेल्या सर्व प्रमुख ऑपरेशन्ससाठी मेनू आयटम असतात. KompoZer च्या टूलबार युजर इंटरफेसमध्ये हे समाविष्ट आहे:

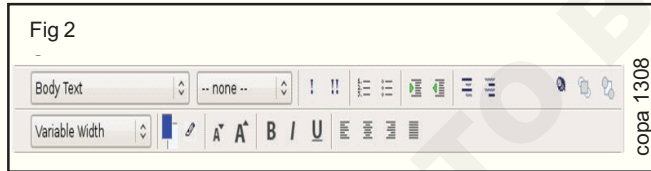
- रचना टूलबार
- फॉर्मॅट टूलबार
- टॅब ब्राउझर टूलबार
- एडिटिंग मोड टूलबार
- स्थिती टूलबार

### कंपोजिशन टूलबार

कंपोजिशन टूलबारमध्ये सर्वाधिक वापरल्या जाणाऱ्या प्रमुख ऑपरेशन्ससाठी बटणे असतात. खाली डीफॉल्ट कंपोजिशन टूलबारचा स्नॅपशॉट प्रत्येक बटणाच्या खाली संबंधित बटणाचे कार्य दर्शविणारा टेक्स्ट आहे. (आकृती क्रं 1)



### फॉर्मॅट टूलबार (पिक्चर 2)



KompoZer सह वेब पेजेज एडिट करताना फॉर्मॅट टूलबार हे अतिशय सबयुक्त साधन आहे. फॉर्मॅट टूलबारसह तुम्ही प्याराग्राफ फॉर्मॅट लागू करू शकता, फॉन्ट निवडू शकता, अग्रपार्ट किंवा बॅकग्राउंड रंग बदलू शकता, आकार वाढवू किंवा कमी करू शकता आणि टेक्स्ट च्या ब्लॉकवर जोर देऊ शकता. तुम्ही क्रमबद्ध किंवा अक्रमित याद्या देखील तयार करू

शकता आणि टेक्स्ट चा ब्लॉक डावीकडे, उजवीकडे किंवा मध्यभागी ठेवू शकता. कोणते बटण काय करते हे जाणून घेण्यासाठी फक्त बटणावर माउस फिरवा आणि बटणाचे कार्य दर्शविणारी एक टूल टीप दिसेल.

### टॅब ब्राउझर टूलबार (पिक्चर 3)



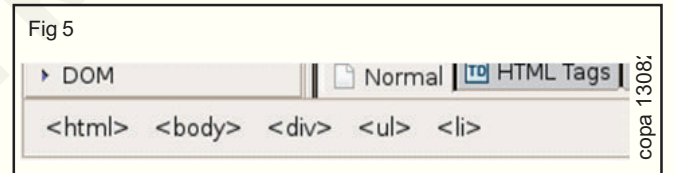
KompoZer तुम्हाला प्रत्येक उघडलेल्या डॉक्युमेंट्स साठी वेगवेगळे टॅब वापरून मल्टिपल वेब डॉक्युमेंट्स एकाच वेळी एडिट करण्याची परवानगी देते. मल्टिपल टॅब असल्याने तुमच्या डेस्कटॉपला स्वच्छ लूक मिळतो कारण प्रत्येक डॉक्युमेंट्स साठी मल्टिपल विंडो वापरल्यावर ते गोंधळलेले नसते. व्हिज्युअल इंडिकेटर म्हणून एडिट केलेल्या परंतु अद्याप सेव्ह न केलेल्या पेजसाठी "लाल फ्लॉपी चिन्ह" चिन्ह दिसते.

### एडिटिंग मोड टूलबार (पिक्चर 4)



एडिटिंग मोड टूलबार करंट डॉक्युमेंट्स साठी सध्या ऍक्टिव्ह असलेला व्ह मोड दर्शवितो. सामान्य व्ह, HTML टॅग व्ह, HTML सोर्स व्ह आणि प्रीव्यू मोड हे सबलब्ध पाहण्याचे मोड आहेत. तुम्ही माऊसच्या सहाय्याने इतर तीनपैकी कोणत्याही एका क्लिक करून तुमचा व्हिंग मोड सहजपणे बदलू शकता.

### स्थिती टूलबार (पिक्चर 5)



KompoZer चा स्टेटस बार HTML टॅग पदानुक्रमाच्या संदर्भात कर्सरची स्थिती दर्शवितो. तुम्ही फक्त राइट क्लिक करून आणि इच्छित पर्याय निवडून स्टेटस बारमधील कोणत्याही स्पेसिफिक टॅगचे प्रॉपर्टीज सहजपणे एडिट /नियुक्त करू शकता. स्टेटस बारमधील टॅगवर माऊसचे साधे लेफ्ट-क्लिक त्या टॅगने वेढलेला टेक्स्ट निवडतो.

## JavaScript समजून घेणे (Understanding JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- प्रोग्रामिंग आणि स्क्रिप्टिंग भाषा परिभाषित करा
- JavaScript काय आहे आणि Java Script चा इतिहास जाणून घ्या
- JavaScript कसे चालवायचे ते स्पष्ट करा
- तुम्हाला JavaScript चालवण्यासाठी आवश्यक असलेल्या साधनांची यादी करा
- नमुना JavaScript प्रोग्राम पहा
- JavaScript ची वैशिष्ट्ये जाणून घ्या
- JavaScript चे फायदे आणि तोटे यांचे डिस्क्रिप्शन करा
- JavaScript आवृत्त्या स्पष्ट करा

### प्रोग्रामिंग आणि स्क्रिप्टिंग भाषांचा परिचय

संगणक प्रोग्रामिंग संगणकाद्वारे कार्यान्वित होणाऱ्या सूचना लिहिण्याची प्रक्रिया आहे. कोड म्हणून ओळखल्या जाणाऱ्या सूचना, प्रोग्रामिंग भाषेत लिहिल्या जातात ज्या संगणक समजू शकतो आणि कार्य करण्यासाठी किंवा समस्या सोडवण्यासाठी वापरू शकतो.

एस्क्रिप्ट किंवा स्क्रिप्टिंग भाषा संकलित न करता कार्यान्वित करण्यास सक्षम असलेल्या फाईलमधील कमांड ची सीरीज असलेली संगणक भाषा आहे. सर्व्हर-साइड स्क्रिप्टिंग भाषांच्या चांगल्या उदाहरणांमध्ये पर्ल, पीएचपी आणि पायथन यांचा समावेश आहे. क्लायंट साइड स्क्रिप्टिंग भाषेचे सर्वोत्तम उदाहरण म्हणजे JavaScript.

### स्क्रिप्टचे फायदे

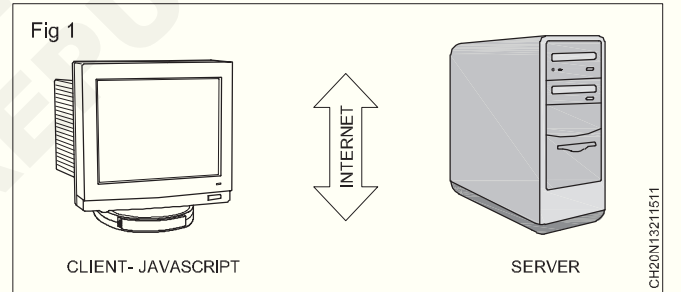
- ओपन सोर्स, वापरकर्त्यांना आवश्यक असल्यास स्क्रिप्ट पाहण्याची आणि एडिट करण्याची अनुमती देते.
- फाइल संकलित करणे आवश्यक नाही, परंतु आवश्यक असेल तेव्हा असू शकते.
- शिकायला आणि लिहायला सोपे.
- वेगवेगळ्या ऑपरेटिंग सिस्टीममध्ये पोर्ट करणे सोपे.
- वास्तविक प्रोग्रामपेक्षा विकसित होण्यासाठी खूप जलद – काही व्यक्ती आणि कंपनी वास्तविक प्रोग्रामसाठी प्रोटोटाइप म्हणून स्क्रिप्ट लिहितात.

### स्क्रिप्टचे तोटे

- ओपन सोर्स, इतरांना सोर्स कोड पाहण्याची अनुमती देते, जो काही कंपन्यांद्वारे प्रतिबंधित असू शकतो.
- स्क्रिप्ट चालवण्याआधी यूजर ने दुभाषी किंवा स्वतंत्र प्रोग्राम इंस्टाल करणे आवश्यक आहे.
- काही परिस्थितींमध्ये, ते संकलित प्रोग्रामपेक्षा हळू असू शकतात.जावा

### स्क्रिप्ट म्हणजे काय?

JavaScript ही अतिशय शक्तिशाली क्लायंट-साइड स्क्रिप्टिंग भाषा आहे. JavaScript चा वापर मुख्यत्वे वेबपेजसह यूजर चा संवाद वाढवण्यासाठी केला जातो (पिक्चर 1). दुसऱ्या शब्दांत, JavaScript च्या मदतीने तुम्ही तुमचे वेबपेज अधिक जीवंत आणि परस्परसंवादी बनवू शकता. गेम डेव्हलपमेंट आणि मोबाईल ॲप्लिकेशन डेव्हलपमेंटमध्ये जावास्क्रिप्टचा वापर मोठ्या प्रमाणावर केला जात आहे.



### JavaScript इतिहास

जावास्क्रिप्ट ब्रॅंडन इच यांनी 1995 मध्ये विकसित केली होती, जी त्या काळातील लोकप्रिय ब्राउझर नेटस्केपमध्ये दिसून आली. या भाषेला सुरुवातीला लाईव्ह स्क्रिप्ट असे म्हटले गेले आणि नंतर त्याचे नाव जावास्क्रिप्ट ठेवण्यात आले. असे बरेच प्रोग्रामर आहेत ज्यांना असे वाटते की JavaScript आणि Java समान आहेत. खरं तर, JavaScript आणि Java यांचा फारसा संबंध नाही. Java ही एक अतिशय क्लिष्ट प्रोग्रामिंग भाषा आहे तर JavaScript ही फक्त एक स्क्रिप्टिंग भाषा आहे. जावा स्क्रिप्टचे वाक्यरचना मुख्यतः प्रोग्रामिंग भाषा सी द्वारे प्रभावित आहे.

### JavaScript कसे चालवायचे?

स्क्रिप्टिंग भाषा असल्याने, JavaScript स्वतः चालवू शकत नाही. खरं तर, JavaScript कोड चालवण्यासाठी ब्राउझर जबाबदार आहे. जेव्हा युजर JavaScript सह HTML पेजची विनंती करतो तेव्हा स्क्रिप्ट ब्राउझरला पाठविली जाते आणि ते कार्यान्वित करणे ब्राउझरवर अवलंबून असते. JavaScript चा मुख्य फायदा म्हणजे सर्व आधुनिक वेब ब्राउझर JavaScript ला सपोर्ट करतात. त्यामुळे, तुमचा साइट अभ्यागत इंटरनेट एक्सप्लोरर,

गूगल क्रोम, फायरफॉक्स किंवा इतर कोणताही ब्राउझर वापरतो की नाही याबद्दल तुम्हाला काळजी करण्याची गरज नाही. JavaScript समर्थित असेल. तसेच, JavaScript विंडोज, लिनक्स किंवा मॅकसह कोणत्याही ऑपरेटिंग सिस्टमवर चालते.

### तुम्हाला JavaScript चालवण्यासाठी आवश्यक असलेली साधने

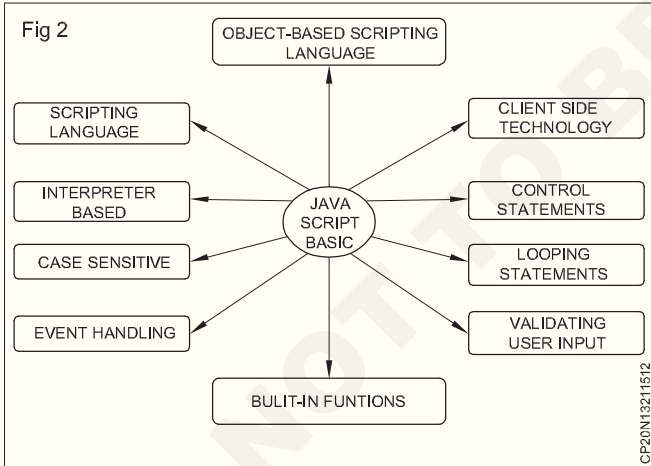
सुरुवातीला, कोड लिहिण्यासाठी टेक्स्ट एडिटर आणि वेब पेजेज प्रदर्शित करण्यासाठी ब्राउझर. नोटपॅड++, व्हिज्युअल स्टुडिओ कोड, सबलाइम टेक्स्ट, अॅटम किंवा इतर कोणत्याही टेक्स्ट एडिटरसह पसंतीचा टेक्स्ट एडिटर वापरतो. आणि गुगल क्रोम, फायरफॉक्स, मायक्रोसॉफ्ट एज, इंटरनेट एक्सप्लोरर इत्यादींसह कोणताही वेब ब्राउझर वापरू शकतो.

### नमुना JavaScript प्रोग्राम

```
<html>
<head>
<title>My First JavaScript code!!!</title>
<script type="text/javascript">
alert("Welcome to JavaScript Program!");
</script>
</head>
<body>
</body>
</html>
```

### JavaScript ची वैशिष्ट्ये

JavaScript हे क्लायंट साइड टेक्नॉलॉजी आहे, ते मुख्यतः क्लायंट साइड व्हॅलिडेशनसाठी वापरले जाते, परंतु त्यात बरीच वैशिष्ट्ये आहेत जी पिकचर 2 मध्ये दर्शविली आहेत.



- JavaScript ही ऑब्जेक्ट-आधारित स्क्रिप्टिंग भाषा आहे.
- हे यूजर ला ब्राउझरवर अधिक कंट्रोल देते.
- ते तारखा आणि वेळ हाताळते.
- ते यूजर चा ब्राउझर आणि OS सर्च ते,
- हे हलके वजनाचे आहे.
- ही एक स्क्रिप्टिंग भाषा आहे आणि ती जावा नाही.

- ही दुभाषी आधारित स्क्रिप्टिंग भाषा आहे.
- हे केस सेन्सिटिव्ह आहे.
- ही ऑब्जेक्ट आधारित भाषा आहे कारण ती पूर्वनिर्धारित वस्तू प्रदान करते.
- JavaScript मधील प्रत्येक स्टेटमेंट अर्धविराम (;) सह समाप्त करणे आवश्यक आहे.
- बहुतेक JavaScript कंट्रोल स्टेटमेंट्स सिंटॅक्स सी भाषेतील कंट्रोल स्टेटमेंट्सच्या सिंटॅक्स प्रमाणेच आहे.
- JavaScript चा महत्त्वाचा पार्ट म्हणजे स्क्रिप्टमध्ये नवीन फंक्शन्स तयार करण्याची क्षमता.

### JavaScript चे फायदे

- क्लायंटच्या बाजूने कार्यान्वित: उदाहरणार्थ, सर्व्हरला विनंती पाठवण्यापूर्वी युजर कोणतेही युजर इनपुट सत्यापित करू शकतो. त्यामुळे सर्व्हरवर कमी लोड होतो.
- तुलनेने एक सोपी भाषा: ही शिकण्यास अगदी सोपी आहे आणि वाक्यरचना इंग्रजीच्या जवळ आहे.
- अभ्यागतांना उदाहरण प्रतिसाद: कोणत्याही सर्व्हर परस्परसंवादाशिवाय, इच्छित रिझल्ट मिळविण्यासाठी पृष्ठ रीलोड होण्याची प्रतीक्षा करण्याची गरज नाही.
- अंतिम यूजर साठी जलद: यूजर च्या संगणकावर स्क्रिप्ट कार्यान्वित केल्यामुळे, कार्यावर अवलंबून, रिझल्ट जवळजवळ त्वरित पूर्ण होतात.
- परस्पर क्रियाशीलता वाढली: युजर जेव्हा कीबोर्ड वापरून त्यावर फिरतो किंवा ऍक्टिव्ह करतो तेव्हा प्रतिक्रिया देऊ शकतील असे इंटरफेस तयार करणे.
- रिच इंटरफेस: ड्रॅग आणि ड्रॉप एलिमेंट किंवा स्लाइडर साइट अभ्यागतांना एक समृद्ध इंटरफेस देऊ शकतात.

### JavaScript चे तोटे

- सुरक्षा समस्या: कोणतीही JavaScript स्निपेट्स, क्लायंटच्या बाजूने वेब पेजवर ताबडतोब जोडलेली असताना यूजर च्या सिस्टमचे शोषण करण्यासाठी देखील वापरली जाऊ शकते.
- कोणतेही मल्टीप्रोसेसर किंवा मल्टी थ्रेडिंग क्षमता नाही.
- कोणतेही सपोर्ट सबलब्ध नसल्यामुळे, कोणत्याही नेटवर्किंग एप्लिकेशन साठी JavaScript वापरले जाऊ शकत नाही.
- JavaScript आम्हाला फाइल्स वाचू किंवा लिहू देत नाही.
- JavaScript रेंडर बदलते: JavaScript वेगवेगळ्या लेआउट इंजिनद्वारे वेगळ्या मेथड्स ने रेंडर केले जाऊ शकते. परिणामी, यामुळे इंटरफेस आणि कार्यक्षमतेच्या बाबतीत विसंगती निर्माण होते.



जावास्क्रिप्टचा सर्च ब्रॅंडन इच यांनी 1995 मध्ये लावला आणि 1997 मध्ये ECMA स्टॅण्डर्ड बनले. ECMA स्क्रिप्ट हे भाषेचे अधिकृत नाव आहे.

## ECMA स्क्रिप्ट आवृत्त्या

पाहण्यासाठी	अधिकृत नाव	डिस्क्रिप्शन
1	ECMA स्क्रिप्ट 1 (1997)	पहिली आवृत्ती.
2	ECMA स्क्रिप्ट 2 (1998)	फक्त संपादकीय बदल.
3	ECMA स्क्रिप्ट 3 (1999)	नियमिती जोडले अभिव्यक्ती. जोडले प्रयत्न / पकडणे.
4	ECMA स्क्रिप्ट 4	कधीही सोडले नाही.
5	ECMA स्क्रिप्ट 5 (2009)	जोडले "कडक JSON सपोर्ट . जोडले Added String.trim(). Added Array.isArray(). अरे पुनरावृत्ती मेथड्स जोडल्या.
5.1	ECMAScript 5.1 (2011)	संपादकीय बदल.
6	ECMA स्क्रिप्ट 2015	let आणि const जोडले. डीफॉल्ट पॅरामीटर व्हॅल्यू जोडली. Array.find(). जोडले Array.findIndex().
7	ECMA स्क्रिप्ट 2016	घातांक जोडले ऑपरेटर (**). जोडले Added Array.prototype.includes.
8	ECMA स्क्रिप्ट 2017	सूट्रिंग पॅडिंग जोडले. नवीन ऑब्जेक्ट प्रॉपर्टीज जोडले. Async फंक्शन जोडली. सामायिक मेमरी जोडली
9	ECMA स्क्रिप्ट 2018	वशि्रांती/सुप्रेड जोडले प्रॉपर्टीज जोडले असकिरोनस पुनरावृत्ती. असकिरोनस पुनरावृत्ती. Promise.finally(). Reg Exp ला जोडले.

## वेब सर्व्हर आणि एक्सटर्नल JavaScript फाइल्सचा परिचय (Introduction to Web servers and External JavaScript files)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- वेब सर्व्हर आणि त्यांची वैशिष्ट्ये स्पष्ट करा
- एक्सटर्नल JavaScript फाइल्स स्पष्ट करा.

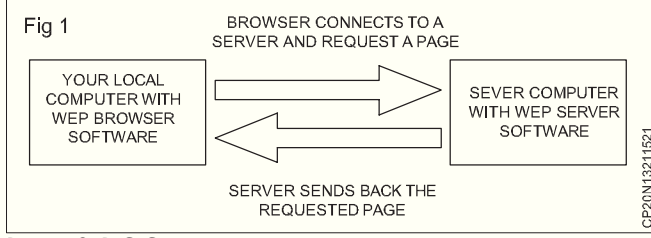
### वेब सर्व्हर म्हणजे काय आणि ते कसे कार्य करते?

वेब सर्व्हर कोणत्याही वेब होस्टिंगसाठी कोर आहेत. (आकृती क्रं 1)

वेब सर्व्हर एक प्रोग्राम आहे जो वापरतो HTTP वापरकर्त्यांना त्यांच्या विनंत्यांना प्रतिसाद म्हणून वेब पेजेज तयार करणाऱ्या फाइल्स सर्व्हर करण्यासाठी, जे त्यांच्या संगणक HTTP द्वारे पाठवले जाते

कनेक्शन. कोणताही सर्व्हर जो वितरित करतो XML डॉक्युमेंट्स दुसऱ्या डिव्हाइसवर वेब सर्व्हर असू शकते. वेब सर्व्हर हा इंटरनेट सर्व्हर आहे जो कंटेंट आणि सर्विस वितरित करण्यासाठी HTTP विनंत्यांना प्रतिसाद देतो अशी चांगली व्याख्या असू शकते. वेब सर्व्हर नेहमी इंटरनेटशी कनेक्ट केलेला असतो. इंटरनेटशी कनेक्ट होणाऱ्या प्रत्येक वेब सर्व्हरमध्ये एक असेल अद्वितीय एड्रेस ज्यामध्ये 0 आणि 255 मधील चार नंबर ची सीरीज आहे. एक कालावधी (.) या नंबर ना वेगळे करतो. तसेच, वेब सर्व्हर सक्षम

करतो.होस्टिंग प्रदातेएकाच सर्व्हरवर मल्टिपल डोमेन(वापरकर्ते) मॅनेज करण्यासाठी. वेब होस्ट ही एक कंपनी आहे जी सर्व्हरच्या क्लस्टरवर लोकांना त्यांची स्वतःची कंटेंट आणि वेबपेजेज सर्विस देण्यासाठी सक्षम करण्यासाठी जागा भाड्याने देते.



### वेब सर्व्हरचे विविध टाइप

खुल्या बाजारात विविध टाइप चे वेब सर्व्हर सबलब्ध आहेत. चला सर्वात लोकप्रिय वेब सर्व्हरबद्दल चर्चा करूया. Apache, IIS, Nginx आणि Lite Speed हे त्यापैकी काही आहेत.

### अपाचे वेब सर्व्हर

Apache Software Foundation ने विकसित केलेला जगातील सर्वात लोकप्रिय वेब सर्व्हरपैकी एक. अपाचे हे ओपन सोर्स सॉफ्टवेअर आहे जे लिनक्स, युनिक्स, विंडोज, फ्रीबीएसडी, मॅक ओएस एक्स आणि बरेच काही सह जवळजवळ सर्व ऑपरेटिंग सिस्टमला सपोर्ट देते. सुमारे ६०% मशीन्स Apache Web Server वर चालतात. (पिक्चर 2)



अपाचे वेब सर्व्हरचे कस्टमाइज करणे सोपे आहे कारण त्यात मॉड्यूलर रचना आहे. हे एक ओपन सोर्स देखील आहे ज्याचा अर्थ असा आहे की सर्व्हरवर स्वतःचे मॉड्यूल जोडू शकतात आणि आवश्यकतेनुसार बदल करू शकतात. हे इतर कोणत्याही वेब सर्व्हरपेक्षा अधिक स्थिर आहे आणि अडमिनिस्ट्रेटिव्ह समस्या सोडवणे सोपे आहे. हे मल्टिपल प्लॅटफॉर्मवर यशस्वीरित्या इंस्टाल केले जाऊ शकते. अलीकडील अपाचे प्रकाशन त्याच्या पूर्वीच्या आवृत्त्यांशी कम्यारीजन करताना अधिक विनंत्या हाताळण्याची व्यवहार्यता प्रदान करतात.

### IIS वेब सर्व्हर

आयआयएस हे मायक्रोसॉफ्टचे उत्पादन आहे. या सर्व्हरमध्ये अपाचेप्रमाणेच सर्व वैशिष्ट्ये आहेत. परंतु हे ओपन सोर्स नाही आणि वैयक्तिक मॉड्यूलस जोडणे सोपे नाही आणि बदल करणे थोडे कठीण काम होते. (पिक्चर 3)

मायक्रोसॉफ्टने हे उत्पादन विकसित केले आहे आणि ते राखून ठेवते, अशा प्रकारे ते सर्व विंडोज ऑपरेटिंग सिस्टम प्लॅटफॉर्मसह कार्य करते. तसेच, काही समस्या असल्यास ते चांगले ग्राहक सपोर्ट प्रदान करतात.



### Nginx वेब सर्व्हर

आणखी एक ओपन ओपन सोर्स वेब सर्व्हर Nginx आहे, त्यात IMAP/POP3 प्रॉक्सी सर्व्हर समाविष्ट आहे. Nginx त्याच्या उच्च कार्यक्षमता, स्थिरता, साधे कॉन्फिगरेशन आणि कमी संसाधन वापरासाठी ओळखले जाते. (पिक्चर 4)



हा वेब सर्व्हर विनंत्या हाताळण्यासाठी थ्रेड्स वापरत नाही त्याऐवजी अधिक स्केलेबल इव्हेंट-चालित आर्किटेक्चर जे लोड इंटरनल मेमरी लहान आणि अंदाजे प्रमाणात वापरते. हे अलीकडच्या काळात लोकप्रिय होत आहे आणि ते जगभरातील सर्व डोमेनपैकी सुमारे 7.5% होस्ट करत आहे. अलीकडच्या काळात बहुतेक वेब होस्टिंग कंपन्या याचा वापर करत आहेत.

**एक्सटर्नल JavaScript फाइल्स:** HTML कोडमध्ये JavaScript लिहिल्याने काहीवेळा गोंधळ निर्माण होतो आणि HTML फायली बदलल्याने JavaScript फाइल्सवरही रिझल्ट होऊ शकतो. HTML आणि JavaScript फायली वेगळ्या करणे चांगले आहे जेणेकरून, एका फाइलमधील बदलांचा इतर फायलींवर रिझल्ट होणार नाही.

एक्सटर्नल JavaScript फायली स्वतंत्रपणे लिहिल्या पाहिजेत

पुढीलप्रमाणे:-

फाइल myjs.js कंटेंट :

```
function popup() {
  alert("Hello World");
}
```

आता आपण HTML फाइलमध्ये खालीलप्रमाणे फाइल आयात करू शकतो:-

एक्सटर्नल फाइल आयात करणे तुलनेने वेदनारहित आहे. प्रथम, तुम्ही आयात करत असलेली फाइल वैध JavaScript आणि फक्त JavaScript असणे आवश्यक आहे. दुसरे, फाइलमध्ये फाइल एक्सपान्शन ".js" असणे आवश्यक आहे. शेवटी, तुम्हाला फाइलचे पोजिशन माहित असणे आवश्यक आहे.

आपण असे गृहीत धरू की आपल्याकडे "myjs.js" फाइल आहे ज्यामध्ये हॅलो वर्ल्ड अलर्ट फंक्शन एक ओळ आहे. तसेच, आपण कोड अप करणार आहोत ती HTML फाइल सारखीच फाइल आहे असे मानू या. फाइल आयात करण्यासाठी तुम्ही तुमच्या HTML डॉक्युमेंट्स त पुढील गोष्टी कराल.

JavaScript कोड:

```
<html>
<head>
<script src="myjs.js">
</script>
```

```
</head>
```

```
<body>
```

```
<input type="button" onclick="popup()" value="Click Me!">
```

```
</body>
```

```
</html>
```

आता ही HTML फाइल myjs.js फाइल इंपोर्ट करते आणि परिणामी ती HTML बटण एलिमेंटवरून पॉपअप() फंक्शन ऍक्सेस करू शकते.

© NIMI  
NOT TO BE REPUBLISHED

## JavaScript व्हेरिएबल आणि डेटा टाइप वापरणे (Using JavaScript Variable and data types)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- JavaScript मध्ये व्हेरिएबल्स स्पष्ट करा
- JavaScript मध्ये विविध डेटा टाइप स्पष्ट करा

### व्हेरिएबल

JavaScript व्हेरिएबल्स डेटा व्हॅल्यू संचयित करण्यासाठी कंटेनर आहेत.

उदाहरणार्थ 1, a, b, आणि c, हे चल आहेत:

### उदाहरण १

```
var a = 12;
```

```
var b = 10;
```

```
var c = a + b
```

उदाहरण १ वरून आपण ते समजू शकतो

- a व्हॅल्यू 12 साठवते
- b हे व्हॅल्यू 10 साठवते
- c हे व्हॅल्यू 22 साठवते

उदाहरणार्थ 2, mark1, mark2 आणि एकूण, हे चल आहेत:

### उदाहरण २

```
var mark1 = 85;
```

```
var mark2 = 66;
```

```
var total = marks1 + mark2
```

प्रोग्रॅमिंगमध्ये, बीजगणिताप्रमाणेच, आम्ही व्हॅल्यू धारण करण्यासाठी mark1 आणि mark2 व्हेरिएबल्स वापरतो आणि एकूण = mark1 + mark2 सारख्या अभिव्यक्तींमध्ये व्हेरिएबल्स वापरतो. वरील उदाहरणावरून, आणि एकूण 151 ची गणना करा.

### JavaScript आयडेंटिफायर

सर्व JavaScript व्हेरिएबल्स अनन्य नावांनी ओळखले जाणे आवश्यक आहे. या द्वितीय नावांना अभिज्ञापक म्हणतात. आयडेंटिफायर ही a आणि b सारखी छोटी नावे असू शकतात किंवा मार्क1, मार्क2, एकूण, वय, बेरीज, एकूण व्हॅल्यूम यासारखी अधिक कॅरक्टर नात्मक नावे असू शकतात.

व्हेरिएबल्ससाठी नावे तयार करण्याचे सामान्य नियम आहेत:

- नावांमध्ये अक्षरे, अंक, अंडरस्कोअर आणि डॉलर सिम्बॉल्स असू शकतात.
- नावे अक्षराने सुरू होणे आवश्यक आहे

- नावे \$ आणि \_ ने देखील सुरू होऊ शकतात
- नावे केस सेन्सिटिव्ह आहेत (a आणि A भिन्न चल आहेत)
- JavaScript कीवर्ड सारखे राखीव शब्द नाव म्हणून वापरले जाऊ शकत नाहीत

**टीप: JavaScript अभिज्ञापक केस-संवेदी आहेत.**

### असाइनमेंट ऑपरेटर

JavaScript मध्ये, समान चिन्ह (=) एक "असाइनमेंट" ऑपरेटर आहे, "इक्वल टू" ऑपरेटर नाही.

```
x = x + 10;
```

हे x + 10 चे व्हॅल्यू x ला नियुक्त करते. हे x + 10 चे व्हॅल्यू मोजते आणि रिझल्ट x मध्ये ठेवते. x चे व्हॅल्यू 10 ने वाढले आहे.

### JavaScript डेटा टाइप

JavaScript व्हेरिएबल्समध्ये 100 सारखी नंबर आणि "संतोष कुमार" सारखी टेक्स्ट व्हॅल्यू असू शकतात.

प्रोग्रॅमिंगमध्ये, टेक्स्ट व्हॅल्यू ना टेक्स्ट स्ट्रिंग म्हणतात. जावा स्क्रिप्ट अनेक टाइप चे डेटा हाताळू शकते, परंतु सध्या, फक्त नंबर आणि स्ट्रिंग्सचा विचार करा. स्ट्रिंग्स दुहेरी किंवा सिंगल कोट्समध्ये लिहिलेल्या आहेत. अंक कोट न करता लिहिलेले आहेत. जर तुम्ही कोट्समध्ये नंबर टाकली तर ती टेक्स्ट स्ट्रिंग म्हणून मानली जाईल.

### उदाहरण ३

```
var pi = 3.14;
```

```
var person = "santhoshkumar";
```

```
var city = "coimbatore";
```

### JavaScript व्हेरिएबल्स घोषित करणे

JavaScript मध्ये व्हेरिएबल तयार करणे याला व्हेरिएबल घोषित करणे म्हणतात. JavaScript व्हेरिएबल var कीवर्डसह घोषित केले आहे.

```
var प्रशिक्षणार्थी नाव;
```

घोषणेनंतर, व्हेरिएबलचे कोणतेही व्हॅल्यू नसते. तांत्रिकदृष्ट्या त्याचे व्हॅल्यू अनडिफाइन आहे. व्हेरिएबलला व्हॅल्यू नियुक्त करण्यासाठी, समान सिम्बॉल्स वापरा.

```
traineeName = "Santhosh Kumar"
```

जेव्हा तुम्ही व्हेरिबल घोषित करता तेव्हा तुम्ही त्याला व्हॅल्यू नियुक्त करू शकता.

```
var traineeName = "Santhosh Kumar";
```

खालील उदाहरणात, आम्ही ट्रेनीनेम नावाचे एक व्हेरिबल तयार करतो आणि त्याला "संतोष कुमार" हे व्हॅल्यू नियुक्त करतो.

मग आम्ही id="demo" सह HTML परिच्छेदातील व्हॅल्यू "आउटपुट" करतो:

```
<p id="demo"></p>
<script>
var traineeName = "santhoshkumar";
document.getElementById("demo").innerHTML = traineeName;
</script>
```

टीप: स्क्रिप्टच्या सुरुवातीला सर्व व्हेरिबल्स घोषित करणे हा एक चांगला प्रोग्रामिंग सराव आहे.

तुम्ही एका विधानात अनेक व्हेरिबल्स घोषित करू शकता. यासह स्टेटमेंट सुरू करा होते आणि द्वारे व्हेरिबल्स स्वल्पविराम वेगळे करा

#### उदाहरण ४

```
var traineeName = "santhoshkumar",city =
"coimbatore", total="151"
```

#### अनडिफाइन व्हॅल्यू

कॉम्प्युटर प्रोग्राममध्ये, व्हेरिबल्स अनेकदा मूल्याशिवाय घोषित केले जातात. व्हॅल्यू असे काहीतरी असू शकते ज्याची गणना करणे आवश्यक आहे किंवा काहीतरी जे नंतर प्रदान केले जाईल, जसे

युजर इनपुट.

व्हॅल्यू शिवाय घोषित व्हेरिबलचे व्हॅल्यू असेल अनडिफाइन

चल प्रशिक्षणार्थीचे नाव या विधानाच्या अंमलबजावणीनंतर व्हॅल्यू अनडिफाइन असेल.

#### उदाहरण 5

```
var traineeName;
```

#### JavaScript व्हेरिबल्स पुन्हा घोषित करणे

तुम्ही JavaScript व्हेरिबल पुन्हा घोषित केल्यास, ते त्याचे व्हॅल्यू गमावणार नाही. या विधानांच्या अंमलबजावणीनंतरही व्हेरिबल ट्रेनीनेममध्ये "संतोषकुमार" हे व्हॅल्यू असेल.

#### उदाहरण 6

```
var traineeName = "santhoshkumar";
var traineeName
```

## JavaScript अर्थमेटिक

ऑपरेटर वापरून JavaScript व्हेरिबल्ससह अर्थमेटिक करा

like = and +

#### उदाहरण 7

```
var x = 8 + 2 + 5
```

आता x चे व्हॅल्यू आहे १५.

तुम्ही स्ट्रिंग देखील जोडू शकता, परंतु स्ट्रिंग एकत्रित केल्या जातील:

#### उदाहरण 8

```
var x = "Dharani" + " " + "Shree"
```

आता x चे व्हॅल्यू आहे धरणी श्री

पुढील उदाहरणाचा रिझल्ट देतो ७२५.

#### उदाहरण ९

```
var x = "7" + 2 + 5;
```

टीप: जर तुम्ही कोट्समध्ये एक नंबर ठेवली, तर उर्वरित नंबर स्ट्रिंग म्हणून मानल्या जातील आणि जोडल्या जातील.

पुढील उदाहरणाचा रिझल्ट देतो 75.

#### उदाहरण 10

```
var x = 3 + 4 + "5";
```

#### डेटा टाइप

प्रोग्रामिंगमध्ये, डेटा टाइप ही एक महत्वाची संकल्पना आहे. व्हेरिबल्सवर ऑपरेट करण्यास सक्षम होण्यासाठी, डेटा टाइप बदल जाणून घेणे महत्वाचे आहे.

JavaScript व्हेरिबल्समध्ये अनेक डेटा टाइप जसे की नंबर, स्ट्रिंग, ऑब्जेक्ट्स आणि बरेच काही असू शकते.

#### उदाहरण 11

```
var side = 10; // Number
```

```
var firstName = "Rithika"; // String
```

```
var x = {firstName:"Harini", lastName:"Kumar"}; // Object
```

डेटा प्रकारांशिवाय, संगणक सुरक्षितपणे याचे निराकरण करू शकत नाही.

#### उदाहरण 12

```
var a = 10 + "Apple";
```

JavaScript will treat the example above as,

```
var a = "10" + "Apple"
```

#### आउटपुट आहे 10 Apple

टीप: नंबर आणि स्ट्रिंग जोडताना, JavaScript नंबरला स्ट्रिंग मानेल.

JavaScript डावीकडून उजवीकडे अभिव्यक्तींचे असेसमेंट करते. वेगळे सिकेंस भिन्न रिझल्ट देऊ शकतात.

### उदाहरण 13

```
var y = 20 + 5 + "Apple"
```

रिझल्ट आहे 25 सफरचंद

### उदाहरण 14

```
var y = "Apple"+20 + 5
```

रिझल्ट आहे सफरचंद 205.

टीप: पहिल्या उदाहरणात, JavaScript 20 आणि 5 ला नंबर मानते, जोपर्यंत ते "Apple" पर्यंत पोहोचत नाही. दुसऱ्या उदाहरणात, पहिले ऑपरेंड ही स्ट्रिंग असल्याने, सर्व ऑपरेंड्स स्ट्रिंग म्हणून मानली जातात.

### डायनॅमिक डेटा टाइप

JavaScript मध्ये डायनॅमिक टाइप आहेत. याचा अर्थ भिन्न डेटा टाइप ठेवण्यासाठी समान व्हेरिएबल वापरले जाऊ शकते:

### उदाहरण 15

```
var z; // Now z is undefined
z = 10; // Now z is a Number
z = "Sakthi"; // Now z is a String
```

### JavaScript स्ट्रिंग्स

स्ट्रिंग किंवा टेक्स्ट स्ट्रिंग म्हणजे "हरिणी कुमार" सारख्या पात्रांची सीरीज. स्ट्रिंग्स अवतरणांसह लिहिलेल्या आहेत. आपण सिंगल किंवा दुहेरी अवतरण वापरू शकता.

### उदाहरण 16

```
var bikeName = 'Yamaha R15'; // Using single quotes
```

जोपर्यंत ते स्ट्रिंगच्या सभोवतालच्या कोट्सशी जुळत नाहीत तोपर्यंत तुम्ही स्ट्रिंगमध्ये कोट्स वापरू शकता:

### उदाहरण 17

```
var answer = "It's OK"; // Single quote inside double quotes
var answer = 'Patel is called "Iron Man"'; // Double quotes inside single quotes
```

### JavaScript क्रमांक

JavaScript मध्ये फक्त एकाच टाइप ची नंबर असते. नंबर दशांशासह किंवा त्याशिवाय लिहिल्या जाऊ शकतात.

### उदाहरण 18

```
var num1 = 87.0; // Written with decimals
var num2 = 87; // Written without decimals
```

अतिरिक्त मोठ्या किंवा अतिरिक्त लहान नंबर वैज्ञानिक (घातांक) संकेताने लिहिल्या जाऊ शकतात:

### उदाहरण 19

```
var exp1 = 232e5; // result is 23200000
var z = 123e-5; // result is 0.00232
```

### उदाहरण 20

```
var p = 3;
var q = 3;
var r = 5;
(p == q) // Returns true
(p == r) // Returns false
```

टीप: बूलियन बहुतेकदा कंडीशनल टेस्टिंग मध्ये वापरले जातात.

### JavaScript अॅरे

JavaScript अॅरे चौरस कंसात लिहिलेले असतात. अॅरे आयटम स्वल्पविरामाने विभक्त केले जातात. खालील कोड बाइक्स नावाचा अॅरे घोषित करतो (तयार करतो), ज्यामध्ये तीन आयटम (बाईकची नावे) असतात:

### उदाहरण 21

```
var bikes = ["Yamaha", "TVS", "Royal Enfield"];
```

टीप: अॅरे निर्देशांक शून्य-आधारित आहेत, याचा अर्थ पहिला आयटम [0] आहे, दुसरा आहे [1], आणि असेच.

### JavaScript ऑब्जेक्ट्स

JavaScript ऑब्जेक्ट्स कुरळे ब्रेसेसने लिहिलेल्या असतात. ऑब्जेक्ट प्रॉपर्टीज नाव:व्हॅल्यू जोड्या, स्वल्पविरामाने विभक्त केले जातात.

### उदाहरण 22

```
var personName = {firstName:"Harini",lastName:"Kumar",age:13,height:"155 cms"};
```

वरील 22 च्या उदाहरणातील ऑब्जेक्ट (personname) मध्ये 4 प्रॉपर्टीज आहेत: firstName, lastName, वय आणि उंची.

### ऑपरेटरचा टाइप

JavaScriptच्या टाइप चेऑपरेटरचा वापर JavaScript व्हेरिएबलचा टाइप सर्वण्यासाठी केला जातो. ऑपरेटरचा टाइप व्हेरिएबल किंवा एक्सप्रेसनचा टाइप परत करतो. उदाहरण 23

### उदाहरण 23

```
typeof "" // Returns "string"
typeof "Rithika" // Returns "string"
typeof "Harini Kumar" // Returns "string"
typeof 0 // Returns "number"
typeof 81 // Returns "number"
typeof 8.14 // Returns "number"
typeof (3+2) // Returns "number"
```

## अनडिफाइन

JavaScript मध्ये, व्हॅल्यू नसलेल्या व्हेरिएबलचे व्हॅल्यू अनडिफाइन असते. टाइप देखील अनडिफाइन आहे.

### उदाहरण 24

```
var bike; // Value is undefined, type is
          undefined
```

**टीप :** व्हॅल्यू अनडिफाइन वर सेट करून कोणतेही व्हेरिएबल रिकामे केले जाऊ शकते. टाइप देखील अनडिफाइन असेल.

### रिक्त व्हॅल्यू

रिक्त मूल्याचा अपरिभाषिताशी काहीही संबंध नाही. रिकाम्या स्ट्रिंगमध्ये कायदेशीर व्हॅल्यू आणि टाइप दोन्ही असतात.

### उदाहरण 25

```
var bike = ""; // The value is "", the typeof
               is "string"
```

### null

JavaScript मध्ये null म्हणजे "काहीही नाही". हे असे काहीतरी असावे जे अस्तित्वात नाही. JavaScript मध्ये, null चा डेटा टाइप एक ऑब्जेक्ट आहे. तुम्ही एखादी वस्तू शून्य वर सेट करून रिकामी करू शकता.

### उदाहरण 26

```
var personName = {firstName:"Harini", last Name;
                  "Kumar", age:13, height:"155 cms"};
personName      = null; //Now value in null, but
                    type is still an object
```

तुम्ही एखादी वस्तू अनडिफाइन वर सेट करून रिकामी देखील करू शकता:

### उदाहरण 27

```
var personName = {firstName:"Harini", lastName:
                  "Kumar", age:13, height:"155 cms"};
personName      = undefined; // Now both value
                             and type is undefined.
```

### अनडिफाइन आणि शून्य मधील फरक

अनडिफाइन आणि शून्य मूल्यात समान आहेत परंतु प्रकारात भिन्न आहेत.

## JavaScript कॉन्स्टंट्स आणि ऑपरेटर वापरणे (Using JavaScript Constants and Operators)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- JavaScript मध्ये कॉन्स्टंट्स स्पष्ट करा
- JavaScript मध्ये ऑपरेटर समजावून सांगा

## उदाहरण 28

```
typeof undefined // undefined
typeof null      // object
null === undefined // false
null == undefined // true
```

### Primitive Data

Primitive डेटा व्हॅल्यू हे कोणतेही अतिरिक्त प्रॉपर्टीज आणि मेथड्स नसलेले सिंगल साधे डेटा व्हॅल्यू आहे. ऑपरेटरचा टाइप या Primitive प्रकारांपैकी एक परत करू शकतो.

- स्ट्रिंग
- नंबर
- बुलियन
- अनडिफाइन

### उदाहरण 29

```
typeof "Rajesh" // Returns "string"
typeof 1.44     // Returns "number"
typeof true    // Returns "boolean"
typeof false   // Returns "boolean"
typeof a       // if a has no value, it
returns       // "undefined"
```

### कॉम्प्लेक्स डेटा

ऑपरेटरचा टाइप दोन कॉम्प्लेक्स प्रकारांपैकी एक परत करू शकतो:

- कार्य
- वस्तू

ऑपरेटरचा टाइप ऑब्जेक्ट्स, अरे आणि नल दोन्हीसाठी ऑब्जेक्ट रिटर्न करतो. हे फंक्शन्ससाठी ऑब्जेक्ट परत करत नाही.

### उदाहरण 30

```
typeof {name, 'Karthik', age 27} // Returns "object"
typeof [10, 20, 30, 40, 50]     // Returns "object"
                                (not "array", see
                                note below)
typeof null                     // Returns "object"
typeof function sampleFunc() { } // Returns
"function"
```

**टीप:** typeof ऑपरेटर अरेसाठी "ऑब्जेक्ट" परत करतो कारण JavaScript मध्ये अरे हे ऑब्जेक्ट असतात.

## कॉन्स्टंट्स

कॉन्स्टंट्स हा एक स्पेशल टाइप चा चल असतो, एक व्हॅल्यू संचयित करतो जे प्रोग्राम दरम्यान कधीही बदलत नाही.

Constant तयार करण्यासाठी वाक्यरचना आहे.

```
const CONSTANT_NAME:DataType = value
```

वरील वाक्यरचनामध्ये, "const" हा स्पेशल कीवर्ड आहे, जो कॉन्स्टंट्स परिभाषित करण्यासाठी राखीव आहे. तुम्ही बघू शकता, हा सिंटॅक्स बऱ्याच व्हेरिएबल डिक्लेरेशनसारखा दिसतो परंतु var कीवर्ड "const" ने बदलला आहे. बहुतेक प्रोग्रामर स्थिरांकांच्या नावासाठी सर्व कॅप्स वापरतात आणि त्यांना व्हेरिएबल्सपासून वेगळे करतात.

### उदाहरण १

```
const FRIEND = 'Shanthi';  
const BROTHER_AGE = 46;
```

टीप: कीवर्ड **const** हा थोडासा दिशाभूल करणारा आहे. हे स्थिर व्हॅल्यू परिभाषित करत नाही. हे मूल्यासाठी स्थिर रेफरन्स परिभाषित करते. यामुळे, आपण स्थिर Primitive व्हॅल्यू बदलू शकत नाही, परंतु आपण स्थिर वस्तूंचे प्रॉपर्टीज बदलू शकतो.

### Primitive व्हॅल्यू

जर आपण स्थिरांकाला Primitive व्हॅल्यू नियुक्त केले तर आपण Primitive व्हॅल्यू बदलू शकत नाही.

### उदाहरण २

```
const PI = 3.141592653589793;  
PI = 3.14; // This will give  
an error  
PI = PI + 10; // This will also give  
an error
```

### कॉन्स्टंट्स ऑब्जेक्ट बदलू शकतात

कॉन्स्टंट्स ऑब्जेक्ट चे प्रॉपर्टीज बदला.

### उदाहरण ३

```
// You can create a const object:  
const bike = {type: "Yamaha", model: "R15",  
color: "blue"};  
  
// You can change a property:  
bike.color = "grey";  
  
// You can add a property:  
bike.owner = "Sree";
```

परंतु तुम्ही स्थिर वस्तू पुन्हा नियुक्त करू शकत नाही

### उदाहरण ४

```
const bike = {type:"Yamaha", model:"R15",  
color:"blue"};  
bike = {type:"Tvs", model:"Star city", color:"black"};  
  
// ERROR
```

### सतत अॅरे बदलू शकतात

तुम्ही स्थिर अॅरेचे एलिमेंट बदलू शकता.

### उदाहरण 5

```
// You can create a constant array:  
constant bikes = ["TVS", "Yamaha", "Royal  
Enfield"];  
  
// You can change an element;  
bikes[0] = "suzuki";  
  
bikes.push("Bajaj"); //you can add an element
```

परंतु तुम्ही स्थिर अॅरे पुन्हा नियुक्त करू शकत नाही.

### उदाहरण 6

```
const bikes = {"TVS", "Yamaha", "Royal Enfield"};  
bikes = ["TVS", "Yamaha", Bajaj"]; //ERROR
```

### ऑपरेटर्स

JavaScript मध्ये आठ टाइप चे ऑपरेटर आहेत. हे आहेत

- अॅडिटीव्ह ऑपरेटर
- गुणाकार ऑपरेटर
- बिटवाइज ऑपरेटर
- समानता ऑपरेटर
- रिलेशनल ऑपरेटर
- युनरी ऑपरेटर
- टर्नरी ऑपरेटर
- असाइनमेंट ऑपरेटर

**अॅडिटीव्ह ऑपरेटर:** अॅडिटीव्ह ऑपरेटर्समध्ये बेरीज (+) आणि वजाबाकी(-) दोन्ही समाविष्ट आहेत कारण वजाबाकी ही ऋण संख्येसह बेरीज आहे.

### उदाहरण 7

```
32+67; // this is 99  
d+e; // Adds d with e  
3-7; // return -4
```

कधीकधी JavaScript जोडल्याने अनपेक्षित रिझल्ट मिळू शकतात.





ऑपरेटरपेक्षा कमी हे पहिले व्हॅल्यू दुसऱ्या मूल्यापेक्षा कमी आहे की नाही हे तपासते आणि वैध असल्यास, ते खोटे दाखवते. वरील उदाहरणात, 5 हे 2 पेक्षा कमी नाही, त्यामुळे ते सत्य नाही आणि if ब्लॉकमध्ये कोड कार्यान्वित होणार नाही.

इतर तीन ऑपरेटर त्याच प्रकारे जास्त (>), त्यापेक्षा मोठे किंवा समान (>=), यापेक्षा कमी किंवा समान (<=) तपासत आहेत.

मध्ये ऑपरेटर ऑब्जेक्टमध्ये दिलेला निर्देशांक आहे की नाही हे तपासतो.

```
var MyObj= {star:"Algol", constellation: "Perseus"};
if("star" in MyObj) {
    // do something
}
```

म्हणून स्टार एक निर्देशांक आहे जो कोड कार्य करेल. परंतु ऑपरेटरमध्ये नंबर त्मक प्रकारांवर कार्य करू नका कारण ते फक्त नंबर साठी कार्य करते.

उदाहरण 11

```
var mydate=new Date();
if(mydate instanceof Date) {
    //do something
}
```

उदाहरण ऑपरेटर ऑब्जेक्टचे उदाहरण किंवा ऑब्जेक्ट व्हेरिबल हे एखाद्या स्पेसिफिक ऑब्जेक्टचे उदाहरण आहे की नाही हे तपासतो.

येथे mydate हे अंगभूत दिनांक ऑब्जेक्टचे उदाहरण आहे. त्यामुळे कोड if ब्लॉकमध्ये कार्यान्वित केला जाईल.

### युनरी ऑपरेटर

डिलेट करा, शून्य, टाइप, ++, --, +, -, ~, ! Javascript मध्ये unary ऑपरेटर आहेत.

उदाहरण 12

```
a = -10;
p=++a;
q=a++;
s+=p;
प्री आणि पोस्ट इन्क्रिमेंट आणि डिक्रिमेंट ऑपरेटर आहेत.
```

p=++a; च्या समतुल्य आहे

```
a=a+1;
p=a;
आणि q=a++; च्या समतुल्य आहे
q=a;
```

a=a+1;

### डिलेट ऑपरेटर

डिलीट ऑपरेटरचा वापर ऑब्जेक्ट्समधील प्रॉपर्टीज हटवण्यासाठी केला जाऊ शकतो.

### उदाहरण 13

```
var person = {firstName:"John", lastName:"Doe", age:50,
eyeColor:"blue"};
```

डिलीट ऑपरेटर ऑब्जेक्ट गुणधर्मांवर वापरण्यासाठी डिझाइन केले आहे. व्हेरिबल किंवा फंक्शन्सवर त्याचा कोणताही रिझल्ट होत नाही.

डिलीट ऑपरेटर पूर्वनिर्धारित Java Script ऑब्जेक्ट गुणधर्मांवर वापरला जाऊ नये. तो तुमचा अर्ज कॅश करू शकतो.

### युनरी + ऑपरेटर

युनरी + ऑपरेटरचा वापर व्हेरिबलला संख्येमध्ये रूपांतरित करण्यासाठी केला जाऊ शकतो.

### उदाहरण 14

```
var y = "5"; // y is a string
var x = + y; // x is a number
```

जर व्हेरिबल रूपांतरित केले जाऊ शकत नाही, तरीही ती नंबर होईल, परंतु NaN (नंबर नाही) या मूल्यासह:

### उदाहरण 15

```
var y = "John"; // y is a string
var x = + y; // x is a number (NaN)
```

त्याच प्रकारे Unary - ऑपरेटर देखील चालते.

**टर्नरी किंवा कंडिशनल ऑपरेटर** : इतर असल्यास ते कॉम्पॅक्ट म्हणून वापरले जाऊ शकते.

### उदाहरण 16

```
a = (b>5 ? 4:7); means
if(b>5)
a=4;
```

### असाइनमेंट ऑपरेटर:

असाइनमेंट ऑपरेटर व्हेरिबलमध्ये व्हॅल्यू नियुक्त करण्यासाठी वापरला जातो. याशिवाय = खालीलप्रमाणे कंपाउंड असाइनमेंट ऑपरेटर आहेत-

*=	/=	%=
+=	-=	<<=
>>=	>>>=	&=
^=	=	

a = q; म्हणजे a चे पूर्वीचे व्हॅल्यू हटवून व्हेरिबलमध्ये q चे व्हॅल्यू नियुक्त केले आहे.

आता a\* = 3; a = a\*3 च्या समतुल्य आहे; आणि त्याप्रमाणे इतर सर्व कंपाउंड असाइनमेंट ऑपरेटर वागतात.

## जावा स्क्रिप्ट मधील स्टेटमेंट्स, लूप आणि पॉपअप बॉक्स कंट्रोल करा (Control statements, Loops and Popup boxes in JavaScript)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कंट्रोल स्टेटमेंट स्पष्ट करा
- विविध लूपबद्दल चर्चा करा
- पॉपअप बॉक्सचे यूजेस स्पष्ट करा.

**कंट्रोल स्टेटमेंट:** जेव्हा आपण एखाद्या स्पेसिफिक प्रोग्रामसाठी कोड लिहितो, तेव्हा आपण काहीवेळा भिन्न कृती करण्यासाठी विविध निर्णय घेतो. हे कंडीशनल /कंट्रोल विधानांद्वारे केले जाऊ शकते.

JavaScript मध्ये आमच्याकडे खालील कंडीशनल स्टेटमेंट आहेत:

वापरातरकार्यान्वित करण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी, जर स्पेसिफाइड कंडीशन सत्य असेल तर वापराइतर अंमलात आणण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी, जर तीच कंडीशन चुकीची असेल तर वापराअन्यथा जरटेस्टिंग साठी नवीन कंडीशन स्पेसिफाइड करण्यासाठी, जर पहिली कंडीशन खोटी असेल तर वापरास्विचअंमलात आणण्यासाठी कोडचे अनेक पर्यायी ब्लॉक्स स्पेसिफाइड करण्यासाठी. जर स्टेटमेंट

कंडीशन सत्य असल्यास अंमलात आणण्यासाठी JavaScript कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी if स्टेटमेंट वापरा.

सिन्टेक्स

```
If (condition) {
    block of code to be executed if the condition is true
}
```

### उदाहरण १

Make a "Good day" greeting if the time is less than 18:00:

```
if (time < 18) {
    greeting = "Good day";
}
```

रिझल्ट अभिवादन असेल:

शुभ दिवस

### दुसरे स्टेटमेंट

कंडिशन असत्य असल्यास अंमलात आणण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी else स्टेटमेंट वापरा.

```
if (condition) {
    block of code to be executed if the condition is true
```

```
} else {
    block of code to be executed if the condition is false
}
```

### उदाहरण २

जर वेळ 18:00 पेक्षा कमी असेल, तर "शुभ दिवस" ग्रीटिंग तयार करा, अन्यथा "शुभ संध्याकाळ":

```
if (time < 18) {
    greeting = "Good day";
} else {
    greeting = "Good evening";
```

```
}
```

दरिझल्ट अभिवादन असेल:

शुभ दिवस

### बाकी जर स्टेटमेंट

पहिली कंडीशन असत्य असल्यास नवीन कंडीशन स्पेसिफाइड करण्यासाठी else if स्टेटमेंट वापरा.सिन्टेक्स

सिन्टेक्स

### उदाहरण ३

10:00 पेक्षा कमी वेळ असल्यास, "गुड मॉर्निंग" ग्रीटिंग तयार करा, नसल्यास, परंतु वेळ 18:00 पेक्षा कमी असल्यास, "शुभ दिवस" ग्रीटिंग तयार करा, अन्यथा "शुभ संध्याकाळ":

```
if (time < 10) {
    greeting = "Good morning";
} else if (time < 18) {
    greeting = "Good day";
} else {
    greeting = "Good evening";
}
```

x चा रिझल्ट असेल:

शुभ दिवस

## JavaScript स्विच स्टेटमेंट

अंमलात आणल्या जाणाऱ्या कोडच्या अनेक ब्लॉकपैकी एक निवडण्यासाठी स्विच स्टेटमेंट वापरा. सिन्टेक्स

### Syntax

```
switch(expression) {  
  case n1:  
    code block  
    break;  
  case n2:  
    code block  
    break;  
  default:  
    default code block  
}
```

### हे कसे कार्य करते:

- स्विच अभिव्यक्तीचे एकदा व्हॅल्यु मापन केले जाते.
- अभिव्यक्तीच्या मूल्याची कम्प्यारीजन प्रत्येक केसच्या व्हॅल्यु शी केली जाते.
- जुळत असल्यास, कोडचा संबंधित ब्लॉक कार्यान्वित केला जातो.

### उदाहरण ४

Use today's weekday number to calculate weekday name: (Sunday=0, Monday=1, Tuesday=2, ...)

```
switch (new Date().getDay()) {  
  case 0:  
    day = "Sunday";  
    break;  
  case 1:  
    day = "Monday";  
    break;  
  case 2:  
    day = "Tuesday";  
    break;  
  case 3:  
    day = "Wednesday";  
    break;  
  case 4:  
    day = "Thursday";  
    break;  
  case 5:  
    day = "Friday";
```

```
break;
```

```
case 6:
```

```
day = "Saturday";
```

रिझल्ट दिवस असेल:

मंगळवार

### ब्रेक कीवर्ड

जेव्हा JavaScript कोड इंटरप्रिटर ब्रेक कीवर्डपर्यंत पोहोचतो, तेव्हा तो स्विच ब्लॉकमधून बाहेर पडतो.

हे ब्लॉकच्या आत कोड आणि/किंवा केस टेस्टिंग ची अधिक अंमलबजावणी करणे थांबवेल.

### डीफॉल्ट कीवर्ड

केस जुळत नसल्यास डीफॉल्ट कीवर्ड रन करण्यासाठी कोड स्पेसिफाइड करतो:

### उदाहरण 5

आज शनिवार किंवा Sun नसल्यास, एक डीफॉल्ट संदेश लिहा:

```
switch (new Date().getDay()) {  
  case 6:  
    text = "Today is Saturday";  
    break;  
  case 0:  
    text = "Today is Sunday";  
    break;  
  default:  
    text = "Looking forward to the Weekend";  
}
```

रिझल्ट टेक्स्ट असेल:

वीकेंडची वाट पाहत आहे

### कॉमन कोड आणि फॉल-थ्रू

काहीवेळा, स्विच ब्लॉकमध्ये, तुम्हाला भिन्न केसेस समान कोड वापरण्याची किंवा सामान्य डीफॉल्टमध्ये फॉल-थ्रू करण्याची इच्छा असेल.

पुढील उदाहरणावरून लक्षात घ्या की केसेस समान कोड ब्लॉक शेअर करू शकतात आणि डीफॉल्ट केस स्विच ब्लॉकमध्ये शेवटची केस असणे आवश्यक नाही:

### उदाहरण 6

```
switch (new Date().getDay()) {  
  case 1:  
  case 2:  
  case 3:
```

```

default:
  text = "Weekend is coming";
  break;
case 4:
case 5:
  text = "Weekend is soon";
break;
case 0:
case 6:
  text = "Now in Weekend";
}

```

### JavaScript लूप

लूप सुलभ आहेत, जर तुम्हाला तोच कोड पुन्हा पुन्हा चालवायचा असेल, प्रत्येक वेळी वेगळ्या मूल्यासह.

असह काम करताना बहुतेकदा असे होते:

लिहिण्याऐवजी:

```

text += train[0] + "<br>";
text += train [1] + "<br>";
text += train [2] + "<br>";
text += train [3] + "<br>";
text += train [4] + "<br>";
text += train [5] + "<br>";

```

You can write:

```

for (i = 0; i < train.length; i++) {
  text += train [i] + "<br>";
}

```

### वेगवेगळ्या टाइप चे लूप

JavaScript विविध टाइप च्या लूपचे सपोर्ट करते:

- साठी - कोडच्या ब्लॉकमधून अनेक वेळा लूप होतो
- फॉर/इन - ऑब्जेक्टच्या गुणधर्मांमधून लूप
- करताना - स्पेसिफाइड कंडीशन सत्य असताना कोडच्या ब्लॉकमधून लूप करते
- करा/करताना - स्पेसिफाइड कंडीशन सत्य असताना कोडच्या ब्लॉकमधून लूप करतेफॉर लूप

जेव्हा तुम्ही लूप तयार करू इच्छित असाल तेव्हा फॉर लूप हे बरेचदा साधन असते. फॉर लूपमध्ये खालील वाक्यरचना आहे:

सिन्टेक्स लूप (कोड ब्लॉक) सुरू होण्यापूर्वी स्टेटमेंट 1 कार्यान्वित केले जाते. त्याला आरंभिक पार्ट म्हणतात

```

for (statement 1; statement 2; statement 3) {
  code block to be executed
}

```

स्टेटमेंट २ लूप (कोड ब्लॉक) चालवण्याची स्थिती परिभाषित करते. त्याला कंडिशन पार्ट म्हणतात.

लूप (कोड ब्लॉक) कार्यान्वित झाल्यानंतर स्टेटमेंट 3 प्रत्येक वेळी कार्यान्वित केले जाते. त्याला इन्क्रिमेंट/डिक्रिमेंट पार्ट म्हणतात.

### उदाहरण 7

```

for (i = 0; i < 5; i++) {
  text += "The number is " + i + "<br>";
}

```

वरील उदाहरणावरून, तुम्ही वाचू शकता:

स्टेटमेंट 1 लूप सुरू होण्यापूर्वी व्हेरिएबल सेट करते (var i = 0).

स्टेटमेंट 2 लूप चालवण्याची स्थिती परिभाषित करते (मी 5 पेक्षा कमी असणे आवश्यक आहे).

प्रत्येक वेळी लूपमधील कोड ब्लॉक कार्यान्वित केल्यावर स्टेटमेंट 3 व्हॅल्यू (i++) वाढवते.

### पार्ट आरंभ (Initialisation Part)

साधारणपणे तुम्ही लूपमध्ये वापरलेले व्हेरिएबल (var i = 0) सुरू करण्यासाठी स्टेटमेंट 1 वापराल. हे नेहमीच होत नाही, JavaScript काळजी करत नाही. स्टेटमेंट 1 ऐच्छिक आहे. तुम्ही स्टेटमेंट 1 मध्ये अनेक व्हॅल्यू सुरू करू शकता (स्वल्पविरामाने विभक्त):

### उदाहरण 8

```

for (i = 0, len = train.length, text = ""; i < len; i++) {
  text += train [i] + "<br>";
}

```

आणि तुम्ही स्टेटमेंट 1 वगळू शकता (जसे की जेव्हा तुमची व्हॅल्यू लूप सुरू होण्यापूर्वी सेट केली जातात):

### उदाहरण ९

```

var i = 2;
var len = train.length;
var text = "";
for (; i < len; i++) {
  text += train [i] + "<br>";
}

```

### कंडीशन पार्ट

स्टार्टिंग व्हेरिएबलच्या स्थितीचे असेसमेंट करण्यासाठी अनेकदा स्टेटमेंट 2 वापरले जाते. हे नेहमीच होत नाही, JavaScript काळजी करत नाही. स्टेटमेंट 2 देखील ऐच्छिक आहे.

स्टेटमेंट 2 सत्य परत आल्यास, लूप पुन्हा सुरू होईल, जर ते खोटे परतले तर, लूप संपेल.

तुम्ही स्टेटमेंट 2 वगळल्यास, तुम्हाला लूपमध्ये ब्रेक देणे आवश्यक आहे. अन्यथा पळवाट कधीच संपणार नाही. यामुळे तुमचा ब्राउझर क्रॅश होईल. या ट्यूटोरियलच्या नंतरच्या अध्यायात ब्रेकबद्दल वाचा.

## इन्क्रीमेंट / डिक््रीमेंट पार्ट

अनेकदा स्टेटमेंट 3 स्टार्टिंग व्हेरिएबल वाढवते.

हे नेहमीच नसते, JavaScript ला काळजी नसते आणि स्टेटमेंट 3 पर्यायी आहे.

स्टेटमेंट 3 नकारात्मक वाढ (i--), किंवा मोठी वाढ (i = i + 15), किंवा इतर काहीही करू शकते.

स्टेटमेंट 3 देखील वगळले जाऊ शकते (जसे की जेव्हा तुम्ही लूपमध्ये तुमची व्हॅल्यू वाढवता):

उदाहरण 10

```
var i = 0;
len = train.length;
for (; i < len; ) {
    text += train [i] + "<br>";
    i++;
}
```

## साठी/इन लूप (For/In Loop)

जावास्क्रिप्ट फॉर/इन स्टेटमेंट ऑब्जेक्टच्या गुणधर्मांमधून लूप करते:

```
var person = {fname:"Raja", lname:"Sen", age:35};
var text = "";
var x;
for (x in person) {
    text += person[x];
}
```

## While Loop

स्पेसिफाइड कंडिशन सत्य आहे तोपर्यंत व्हेल लूप कोडच्या ब्लॉकमधून लूप होतो.

## सिन्टेक्स

```
while (condition) {
    code block to be executed
}
```

खालील उदाहरणामध्ये, जोपर्यंत व्हेरिएबल (i) 10 पेक्षा कमी असेल तोपर्यंत लूपमधील कोड पुन्हा पुन्हा चालू होईल:

## उदाहरण 11

```
while (i < 10) {
    text += "The number is " + i;
    i++;
}
```

जर तुम्ही कंडिशनमध्ये वापरलेले व्हेरिएबल वाढवायला विसरलात, तर लूप कधीही संपणार नाही. यामुळे तुमचा ब्राउझर क्रॅश होईल.

## डू/व्हाइल लूप

do/while loop हा while loop चा एक टाइप आहे. हा लूप कोड ब्लॉक एकदा कार्यान्वित करेल, कंडिशन सत्य आहे की नाही हे तपासण्यापूर्वी, कंडिशन सत्य आहे तोपर्यंत तो लूपची पुनरावृत्ती करेल.

## सिन्टेक्स

```
do {
    code block to be executed
}
while (condition);
```

## सिन्टेक्स

खालील उदाहरण do/while लूप वापरते. लूप नेहमी किमान एकदा अंमलात आणला जाईल, जरी कंडिशन चुकीची असली तरीही, कारण कंडिशन तपासण्यापूर्वी कोड ब्लॉक कार्यान्वित केला जातो:

## उदाहरण 12

```
do {
    text += "The number is " + i;
    i++;
}
while (i < 10);
```

कंडिशनमध्ये वापरलेले व्हेरिएबल वाढविण्यास विसरू नका, अन्यथा लूप कधीही संपणार नाही!

## साठी आणि असताना कम्प्यारीजन करणे (Comparing For and While)

जर तुम्ही मागील प्रकरण, फॉर लूपबद्दल वाचले असेल, तर तुम्हाला आढळेल की एक while लूप फॉर लूप प्रमाणेच आहे, स्टेटमेंट 1 आणि स्टेटमेंट 3 वगळले आहे.

या उदाहरणातील लूप ट्रेन अॅरेमधून कारची नावे गोळा करण्यासाठी फॉर लूप वापरते:

## Example 13

```
train = ["Duronto", "Satabdi", "Garib Rath", "Rajdhani"];
var i = 0;
var text = "";
```

```
for (;train[i];) {
    text += train[i] + "<br>";
    i++;
}
```

या उदाहरणातील लूप ट्रेन अरेमधून कारची नावे गोळा करण्यासाठी थोडा वेळ लूप वापरतो:

```
train = ["Duronto", "Satabdi", "Garib Rath", "Rajdhani"];
var i = 0;
var text = "";
while (train[i]) {
    text += train[i] + "<br>";
    i++;
}
```

### लूपमधील ब्रेक स्टेटमेंट

ब्रेक स्टेटमेंटचा वापर लूप पूर्ण होण्यापूर्वी समाप्त करण्यासाठी केला जातो. हे लूप निरुपयोगीपणे पुनरावृत्ती न करण्यासाठी मशीनचा वेळ वाचवते.

उदाहरणार्थ: रेखीय शोधात, जर आपल्याला आयटम सापडला तर आपण लूप खंडित करू शकतो कारण ती अनावश्यकपणे चालवण्याची गरज नाही.

### उदाहरण 14

```
for(i=0;i<l;i++) {
    if(arr[i]==item) {
        alert("Found at :"+i);
        fl=1;
        break;
    }
}
if(fl==0) alert("Not Found");
```

येथे, आयटम आढळल्यास, लूप ब्रेक होतो आणि CPU वेळ पिकचर 2 सेकंद केला जातो.

### पॉपअप बॉक्स

JavaScript मध्ये तीन टाइप चे पॉपअप बॉक्स असतात. ते आहेत

- 1 अलर्ट बॉक्स
- 2 कन्फर्म बॉक्स आणि
- 3 प्रॉम्प्ट बॉक्स.

### अलर्ट बॉक्स

जर तुम्हाला इनफार्मेशन यूजर पर्यंत पोहोचेल याची खात्री करायची असेल तर अनेकदा अलर्ट बॉक्स वापरला जातो. जेव्हा अलर्ट बॉक्स पॉप अप होतो, तेव्हा यूजर ला पुढे जाण्यासाठी "ओके" क्लिक करावे लागेल.

### सिन्टेक्स

```
window.alert("sometext");
```

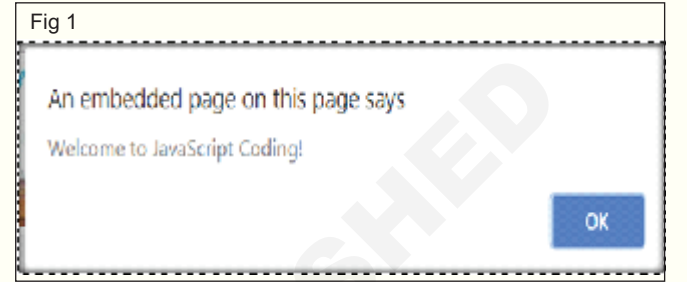
टीप: window.alert() मेथड्स विंडो prefix शिवाय लिहिली जाऊ शकते.

### उदाहरण 15

अलर्ट ("Welcome to Java Script Coding!");

रिझल्ट आकृती 1 मध्ये दर्शविला आहे.

आकृती १



### कन्फर्म बॉक्स

कन्फर्म करण बॉक्स बहुतेक वेळा काहीतरी सत्यापित करण्यासाठी किंवा स्वीकारण्यासाठी वापरला जातो. जेव्हा कन्फर्म करण बॉक्स पॉप अप होतो, तेव्हा यूजर ला पुढे जाण्यासाठी "ओके" किंवा "रद्द करा" क्लिक करावे लागेल. यूजर ने "ओके" वर क्लिक केल्यास, बॉक्स सत्य परत येतो. यूजर ने "रद्द करा" वर क्लिक केल्यास, बॉक्स खोटा परत येईल.

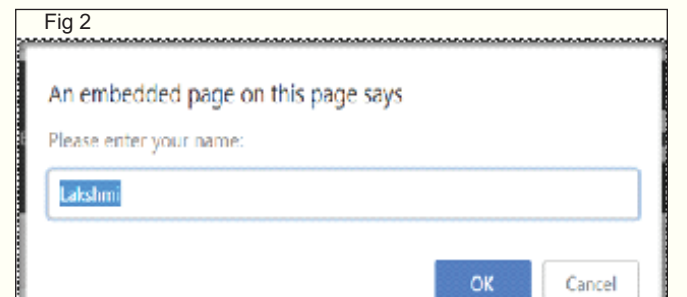
सिन्टेक्स

```
window.confirm("sometext");
```

टीप: window.confirm() मेथड्स विंडो सबसर्गाशिवाय लिहिली जाऊ शकते.

### उदाहरण 16

```
if (confirm("Click a button!"))
{
    txt = " You clicked OK!";
}
else
```



रिझल्ट आकृती 2 मध्ये दर्शविला आहे.

टीप: ओके बटणावर क्लिक केल्यावर “तुम्ही ओके क्लिक केले!” असा संदेश प्रदर्शित होतो. आणि रद्द करा बटणावर क्लिक केल्यावर ते “तुम्ही रद्द करा क्लिक केले!” असा संदेश प्रदर्शित होतो.

#### प्रॉम्प्ट बॉक्स

यूजर ने पृष्ठ प्रविष्ट करण्यापूर्वी व्हॅल्यु इनपुट केल्यास प्रॉम्प्ट बॉक्सचा वापर केला जातो. जेव्हा प्रॉम्प्ट बॉक्स पॉप अप होतो, तेव्हा यूजर स इनपुट व्हॅल्यु प्रविष्ट केल्यानंतर पुढे जाण्यासाठी “ओके” किंवा “रद्द करा” क्लिक करावे लागेल. जर यूजर ने “ओके” क्लिक केल्यावर बॉक्स इनपुट व्हॅल्यु परत करतो. यूजर ने “रद्द करा” वर क्लिक केल्यास बॉक्स शून्य होईल.

#### सिन्टेक्स

```
window.prompt("sometext", "default text");
```

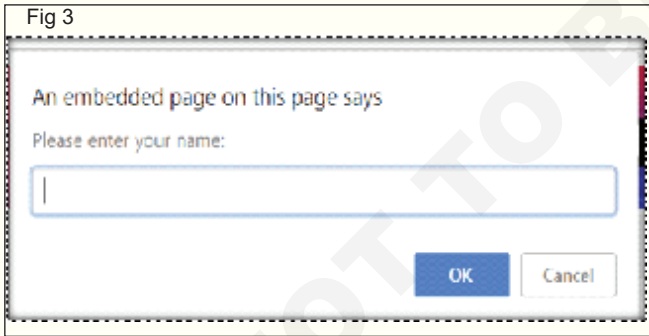
टीप: `window.prompt()` मेथड्स विंडो सबसर्गाशिवाय लिहिली जाऊ शकते.

#### उदाहरण 17

```
var tname = prompt("Please Enter your Name", "Lakshmi");
```

```
if (tname == null || tname == "")  
{txt = "User cancelled the prompt.";}  
else  
{txt = "Hello " + tname + "! Congratulations!!!!!!";}  
}
```

रिझल्ट आकृती 3 मध्ये दर्शविला आहे.



टीप: ओके बटणावर क्लिक केल्यास “नमस्कार लक्ष्मी! अभिनंदन!!!!” आकृती 4 मध्ये दर्शविल्याप्रमाणे ‘लक्ष्मी’ हे नाव रद्द केल्यास ते “यूजर ने प्रॉम्प्ट रद्द केले” असा संदेश देते. तसेच रद्द करा बटणावर क्लिक केल्यावर बॉक्समध्ये ‘लक्ष्मी’ असा टेक्स्ट असला तरीही तो “यूजर ने प्रॉम्प्ट रद्द केला” असा संदेश देतो.

Fig 4



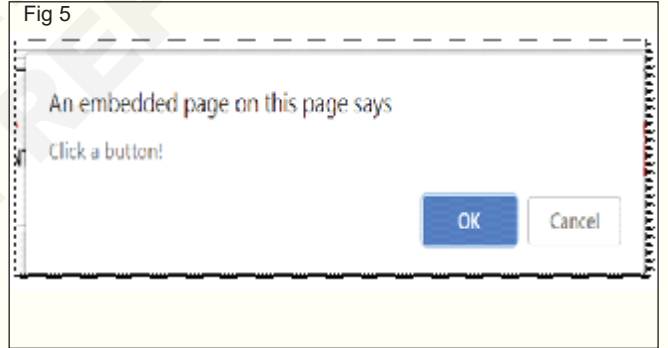
#### लाईन ब्रेक्स

पॉपअप बॉक्समध्ये लाइन ब्रेक्स प्रदर्शित करण्यासाठी,  $n$  वर्णानंतर बॅकस्लॅश वापरा.

#### उदाहरण 18

```
alert("Hello\nWelcome!");
```

रिझल्ट आकृती 5 मध्ये दर्शविला आहे.





## JavaScript मध्ये अरे (Arrays in JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- अरे परिभाषित करा
- अरेच्या संकल्पना स्पष्ट करा
- अरे पद्धतीचे डिस्क्रिप्शन करा
- अरेची क्रमवारी (सॉर्टिंग) जाणून घ्या

### अरे म्हणजे काय?

अरे हे एक स्पेशल व्हेरिअबल आहे, जे एका वेळी एकापेक्षा जास्त व्हॅल्यू धारण करू शकते.

तुमच्याकडे आयटमची लिस्ट असल्यास (उदाहरणार्थ, ट्रेनच्या नावांची लिस्ट), ट्रेन्स सिंगल व्हेरिअबलमध्ये स्टोर करणे यासारखे दिसू शकते.

```
var train1 = "Jan Satabdi";
```

```
var train1 = "Garib Rath";
```

```
var train1 = "Duronto";
```

तथापि, जर तुम्हाला ट्रेनमधून पळवाट काढायची असेल आणि एखादी स्पेसिफिक शोधायची असेल तर? आणि जर तुमच्याकडे 3 गाड्या नसतील तर 300 असतील तर?

उपाय एक अरे आहे!

### JavaScript अरे

JavaScript अरे एकाच व्हेरिअबलमध्ये मल्टिपल व्हॅल्यू संचयित करण्यासाठी वापरली जातात.

### अरे तयार करणे

अरे लिटरल वापरणे हा JavaScript अरे तयार करण्याचा सर्वात सोपा पाथ आहे. सिन्टेक्स :

सिन्टेक्स

```
var array-name = [item1, item2, ...];
```

### उदाहरण १

```
var trains = ["Duronto", "Jan Satabdi", "RAJDHANI"];
```

### JavaScript कीवर्ड नवीन वापरणे

खालील उदाहरण देखील अरे तयार करते आणि त्यास व्हॅल्यू नियुक्त करते: उदाहरण २

```
var trains = new Array("Duronto", "Jan Satabdi", "RAJDHANI");
```

वरील दोन उदाहरणे अगदी सारखीच आहेत. नवीन Array() वापरण्याची गरज नाही.

साधेपणा, वाचनीय (रिडेबल)ता आणि अंमलबजावणी गतीसाठी, प्रथम (अरे शाब्दिक मेथड्स) वापरा.

### अरेच्या एलिमेंटमध्ये एक्सेस करा

तुम्ही इंडेक्स क्रमांकाचा रेफरेन्स देऊन अरे एलिमेंटचा रेफरेन्स घेता.

हे स्टेटमेंट myTrains मधील पहिल्या एलिमेंटच्या मूल्यात एक्सेस करते:

```
var नाव = ट्रेन्स[0];
```

हे स्टेटमेंट ट्रेनमधील प्रथम एलिमेंट सुधारित करते:

```
trains[0] = "जन शताब्दी";
```

[0] हा अरेमधील पहिला एलिमेंट आहे. [१] दुसरा आहे. अरे इंडेक्स 0 ने सुरू होतात.

### अरे प्रदर्शित करत आहे:

id="demo" सह <p> एलिमेंटमध्ये अरे प्रदर्शित करण्यासाठी आम्ही स्क्रिप्ट वापरू:

### उदाहरण ३

```
<p id="demo"></p>
```

```
<script>
```

```
var trains = ["Duronto", "Jan Satabdi", "RAJDHANI"];
```

```
document.getElementById("demo").innerHTML = trains;
```

```
</script>
```

पहिली ओळ (लिपीत) ट्रेन्स नावाची अरे तयार करते.

दुसरी ओळ id="demo" सह एलिमेंट "सर्च ते" आणि त्यातील "innerHTML" मध्ये अरे "डिस्प्ले" करते.

मोकळी जागा आणि लाइन ब्रेक्स महत्त्वाचे नाहीत. घोषणा अनेक ओळींचा असू शकते.

### उदाहरण ४

```
var trains = [
```

```
"Duronto",
```

```
"Jan Satabdi",
```

```
"RAJDHANI"
```

```
];
```

शेवटच्या एलिमेंट नंतर स्वल्पविराम लावू नका (जसे "राजधानी"). हे सर्व ब्राउझरमध्ये विसंगत आहे.

अरे एकाच नावाखाली अनेक व्हॅल्यू धारण करू शकते आणि तुम्ही इंडेक्स नंबरचा रेफरन्स देऊन व्हॅल्यू मध्ये एक्सेस करू शकता.

### तुमच्याकडे एका अरेमध्ये वेगवेगळ्या ऑब्जेक्ट असू शकतात

JavaScript व्हेरिएबल्स ऑब्जेक्ट्स असू शकतात. अरे हे स्पेशल टाईप चे ऑब्जेक्ट्स आहेत. यामुळे, तुमच्याकडे एकाच अरेमध्ये विविध टाईप चे व्हेरिएबल्स असू शकतात.

आपल्याकडे अरेमध्ये ऑब्जेक्ट असू शकतात. तुमच्याकडे अरेमध्ये फंक्शन्स असू शकतात. तुमच्याकडे अरेमध्ये अरे असू शकतात:

```
my Array[0] = Date.now;
```

```
my Array [1] = माझे कार्य;
```

```
my Array [2] = myTrains;
```

### अरे हे ऑब्जेक्ट्स आहेत

अरे हे एक स्पेशल टाईप चे ऑब्जेक्ट्स आहेत. JavaScript मधील ऑपरेटर टाईप अरेसाठी "ऑब्जेक्ट" परत करतो.

परंतु, JavaScript अरेचे डिस्क्रिप्शन अरे म्हणून केले जाते.

अरे त्याच्या "एलिमेंट्स" मध्ये एक्सेस करण्यासाठी नंबर वापरतात. या उदाहरणात, व्यक्ती[0] राजाला परत करते:

### ऑब्जेक्ट्स

```
var person = ["Raja", "Sen", 46];
```

ऑब्जेक्ट त्याच्या "सदस्य" मध्ये एक्सेस करण्यासाठी नावे वापरतात. या उदाहरणात, person.firstName राजा परत करतो:

### ऑब्जेक्ट

```
var व्यक्ती = {प्रथम नाव:"राजा", शेवटचे नाव:"सेन", वय:४६};
```

### अरेची लांबी प्रॉपर्टीज an ची लांबी मिळवते

अरे (अरे एलिमेंटची नंबर).

### उदाहरण 6

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];
```

```
fruits.length; // the length of fruits is 4
```

लेंथ प्रॉपर्टी सर्वोच्च अरे निर्देशांकापेक्षा नेहमीच एक असते.

### अरे एलिमेंट जोडत आहे

अरेमध्ये नवीन एलिमेंट जोडण्याचा सर्वात सोपा पाथ म्हणजे लेंथ प्रॉपर्टीज वापरणे:

### उदाहरण 7

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];
```

उच्च निर्देशांकांसह एलिमेंट जोडल्याने अनडिफाइन तयार होऊ शकते अरेमध्ये "छिद्र":

### उदाहरण 8

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];
```

```
fruits[10] = "Lemon"; // adds a new element (Lemon)
```

```
to fruits
```

### लूपिंग अरे एलिमेंट

अरेद्वारे लूप करण्याचा सर्वोत्तम पाथ म्हणजे लूपसाठी स्टॅण्डर्ड वापरणे:

### उदाहरण ९

```
var index;
```

```
var fruits = ["Banana", "Orange", "Apple", "Mango"];
```

```
for (index = 0; index < fruits.length; index++) {
```

```
text += fruits[index];
```

```
}
```

### असोसिएटिव्ह अरे? पाथ नाही! (Associative Arrays? No Way!)

बऱ्याच प्रोग्रामिंग भाषा नामांकित इंडेक्ससह अरेला सपोर्ट देतात. नामांकित इंडेक्ससह असलेल्या अरेना असोसिएटिव्ह अरे (किंवा हॅश) म्हणतात.

JavaScript नामांकित इंडेक्ससह अरेला सपोर्ट करत नाही.

चुकीचे:

```
var person = new Array()
```

```
person ["firstName"] = "Raja";
```

```
person ["lastName"] = "Sen";
```

```
person ["age"] = 46;
```

वरील उदाहरण ते कार्य करते असे दिसते. पण तसे होत नाही.

तुम्ही प्रयत्न केल्यास, व्यक्ती ["firstName"] राजा परत करेल, परंतु व्यक्ती [0] अनडिफाइन परत येईल, आणि person.length 0 देईल.

तुम्हाला सहयोगी अरे तयार करायचे असल्यास, त्याऐवजी एक ऑब्जेक्ट तयार करा.

### अरे कधी वापरायचे? ऑब्जेक्ट्स कधी वापरायचे?

JavaScript असोसिएटिव्ह अरेला सपोर्ट करत नाही.

जेव्हा तुम्हाला एलिमेंटची नावे स्ट्रिंग्स हवी असतील तेव्हा तुम्ही ऑब्जेक्ट वापरावे. जेव्हा तुम्हाला एलिमेंटची नावे अनुक्रमिक नंबर हवी असतील तेव्हा तुम्ही अरे वापरावे.

### नवीन अरे टाळा()

JavaScript चा बिल्ट-इन अरे कन्स्ट्रक्टर नवीन Array() वापरण्याची गरज नाही.

## त्याऐवजी [] वापरा.

ही दोन भिन्न स्टेटमेंट दोन्ही पॉइंट नावाचा नवीन रिक्त अर्रे तयार करतात.

```
var पॉइंट्स = नवीन अर्रे(); // वाईट
```

```
var गुण = []; // चांगले
```

ही दोन भिन्न स्टेटमेंट दोन्ही 6 नंबर असलेली नवीन अर्रे तयार करतात. var पॉइंट = नवीन अर्रे(40, 100, 1, 5, 25, 10) // खराब

```
var गुण = [४०, १००, १, ५, २५, १०]; // चांगले
```

नवीन कीवर्ड तुमचा कोड क्लिष्ट करतो आणि वाईट दुष्परिणाम निर्माण करतो. var पॉइंट्स = नवीन अर्रे(40, 100); // सह अर्रे तयार करते

दोन एलिमेंट (40 आणि 100)

## मी एलिमेंटपैकी एक काढून टाकल्यास काय होईल?

```
var पॉइंट्स = नवीन अर्रे(40); // 40 अनडिफाइन एलिमेंटसह अर्रे तयार करते !!!!!
```

अर्रे कसे ओळखायचे?

एक सामान्य प्रश्न आहे: व्हेरिएबल अर्रे आहे हे मला कसे कळेल? समस्या अशी आहे की JavaScript ऑपरेटर टाइप परत करतो

“वस्तु”:

```
var फळ = ["केळी", "संत्रा", "सफरचंद", "आंबा"];
```

```
फळांचे टाइप ; // टाइप चा रिटर्न ऑब्जेक्ट
```

ऑपरेटरचा टाइप ऑब्जेक्ट परत करतो कारण JavaScript अर्रे एक ऑब्जेक्ट आहे. या समस्येचे निराकरण करण्यासाठी तुम्ही तुमचे स्वतःचे isArray() फंक्शन तयार करू शकता: जर आर्ग्यूमेंट्स अर्रे असेल तर वरील फंक्शन नेहमी खरे ठरते.

```
{  
return myArray.constructor.toString().indexOf("Array")  
> 1;  
}
```

किंवा अधिक तंतोतंत: जर वितर्काचा ऑब्जेक्ट प्रोटो टाइप “[ऑब्जेक्ट अर्रे]” असेल तर ते खरे होते.

## JavaScript अर्रे मेथड्स

अर्रेचे स्ट्रिंगमध्ये रूपांतर करत आहे

### toString() मेथड्स

JavaScript मेथड्स toString() अर्रेला (स्वल्पविरामाने विभक्त) अर्रे व्हॅल्यू च्या स्ट्रिंगमध्ये रूपांतरित करते.

### उदाहरण 10

```
var trade = ["COPA", "IT", "ICTSM", "CHNM", "Fitter"];  
document.getElementById("demo").innerHTML = trade.  
toString();
```

## रिझल्ट

COPA,IT,ICTSM,CHNM,Fitter

### join() मेथड्स

join() मेथड्स सर्व अर्रे एलिमेंटना स्ट्रिंगमध्ये जोडते. ते toString() प्रमाणेच वागते, परंतु त्याव्यतिरिक्त तुम्ही विभाजक स्पेसिफाइड करू शकता.

### उदाहरण 11

```
var trade = ["COPA", "IT", "ICTSM", "CHNM", "Fitter"];  
document.getElementById("demo").innerHTML =  
trade.join("-");
```

## रिझल्ट

COPA - IT - ICTSM - CHNM - Fitter

### पॉपिंग आणि पुशिंग

जेव्हा तुम्ही अर्रेसह कार्य करता, तेव्हा एलिमेंट काढून टाकणे आणि नवीन एलिमेंट जोडणे सोपे असते.

### पॉपिंग

pop()मेथड्स अर्रेमधून शेवटचा एलिमेंट काढून टाकते.

### उदाहरण 12

```
var trade = ["COPA", "IT", "ICTSM", "CHNM", "Fitter"];  
trade.pop(); // Removes the last element ("Fitter")  
from trade.
```

## रिझल्ट

COPA,IT,ICTSM,CHNM

pop() मेथड्स “पॉप आउट” केलेली व्हॅल्यू परत करते.

### उदाहरण 13

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
var x = trade.pop(); // the value of x is  
"CHNM".
```

### पुशिंग

push() मेथड्स अर्रेमध्ये नवीन एलिमेंट जोडते (शेवटी).

### उदाहरण 14

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
trade.push("DTPO"); // Adds a new element  
("DTPO") to trade.
```

## रिझल्ट

COPA,IT,ICTSM,CHNM,DTPO

पुश() मेथड्स नवीन अँरे लांबी परत करते.

### उदाहरण 15

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
var x = trade.push("DTPO"); // the value of x is 5
```

### शिफ्टिंग एलिमेंट

शिफ्टिंग हे पॉपिंगच्या बरोबरीचे आहे, शेवटच्या ऐवजी पहिल्या एलिमेंटवर कार्य करणे. शिफ्ट() मेथड्स प्रथम अँरे एलिमेंट काढून टाकते आणि इतर सर्व एलिमेंटना कमी निर्देशांकात "शिफ्ट" करते.

### उदाहरण 16

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
trade.shift(); // Removes the first element  
"COPA" from trade.
```

### रिझल्ट

IT, ICTSM, CHNM

शिफ्ट() मेथड्स "शिफ्ट आउट" केलेली स्ट्रिंग परत करते:

### उदाहरण 17

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];
```

### उदाहरण 18

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
trade.unshift("ElecMech"); // Adds a new element  
"ElecMech" to trade
```

### रिझल्ट

ElecMech,COPA,IT,ICTSM,CHNM

unshift() मेथड्स नवीन अँरे लांबी परत करते.

### उदाहरण 19

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
trade.unshift("ElecMech"); // Returns 5
```

### एलिमेंट बदलणे

अँरे एलिमेंटचा इंडेक्स क्रमांक वापरून एक्सेस केला जातो: अँरे इंडेक्स ० ने सुरू होते. [०] हा पहिला अँरे एलिमेंट आहे, [१] दुसरा आहे, [२] तिसरा आहे ...

### उदाहरण 20

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];  
trade[2] = "DTPO"; // Changes the third element  
of trade to "DTPO"
```

### रिझल्ट

COPA, IT, DTPO, CHNM

लांबी प्रॉपर्टीज अँरेमध्ये नवीन एलिमेंट जोडण्याचा एक सोपा पाथ प्रदान करते:उदाहरण 21

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];  
trade[trade.length] = "ICTSM"; // Appends  
"ICTSM" to fruits
```

### रिझल्ट

COPA,IT,DTPO,CHNM,ICTSM

### एलिमेंट हटवत आहे

JavaScript अँरे हे ऑब्जेक्ट असल्याने, JavaScript ऑपरेटर डिलीट वापरून एलिमेंट हटवता येतात.

### उदाहरण 22

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];  
delete trade[0]; // Changes the first element in  
trade to undefined
```

टीप: डिलीट वापरल्याने अँरेमध्ये अनडिफाइन्ड छिद्रे राहू शकतात. त्याऐवजी pop() किंवा shift() वापरा

### अँरेचे पार्टिशन करणे

अँरेमध्ये नवीन आयटम जोडण्यासाठी splice() मेथड्स वापरली जाऊ शकते:

### उदाहरण 23

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];  
trade.splice(2, 0, "Turner", "Machinist");
```

टीप: पहिले पॅरामीटर (२) नवीन एलिमेंट कुठे जोडले जावेत ते पोजिशन परिभाषित करते (स्प्लाय केलेले). दुसरे पॅरामीटर (०) किती एलिमेंट काढले जावे हे परिभाषित करते. उर्वरित पॅरामीटर्स ("टर्नर", "मशिनिस्ट") जोडण्यासाठी नवीन एलिमेंट परिभाषित करा.

### रिझल्ट

COPA, IT, टर्नर, मशीनिस्ट, DTPO, CHNM

### एलिमेंट काढून टाकण्यासाठी splice() वापरणे

चतुर पॅरामीटर सेटिंगसह, तुम्ही अँरेमध्ये "छिद्र" न ठेवता एलिमेंट काढून टाकण्यासाठी splice() वापरू शकता.

### उदाहरण 24

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];  
trade.splice(0, 1); // Removes the first element of  
trade
```

## रिझल्ट

IT, DTPO, CHNM

टीप : पहिले पॅरामीटर (0) नवीन एलिमेंट जोडले जावेत (स्लाय केलेले) पोजिशन परिभाषित करते. दुसरे पॅरामीटर (1) किती एलिमेंट काढून टाकायचे ते परिभाषित करते. बाकीचे पॅरामीटर्स वगळले आहेत. कोणतेही नवीन एलिमेंट जोडले जाणार नाहीत.

### अॅरे विलीन करणे किंवा एकत्र करणे

concat() मेथड्स विद्यमान अॅरे विलीन करून नवीन अॅरे तयार करते.

#### उदाहरण 25 (दोन अॅरे विलीन करणे)

```
var names1 = ["Devi", "Deepa"];
var names2 = ["Poorna", "Saranya", "Shalini"];
var myTrainee = names1.concat(names2);
//Concatenates (joins) names1 and names2.
```

## रिझल्ट

टीप: concat() मेथड्स विद्यमान अॅरे बदलत नाही. ते नेहमी नवीन अॅरे देते.

concat() मेथड्स कितीही अॅरे आर्ग्युमेंट्स घेऊ शकते.

#### उदाहरण 26 (तीन अॅरे विलीन करणे)

```
var arr1 = ["Priya", "Mythili"];
var arr2 = ["Sangeetha", "Nancy", "Sahana"];
var arr3 = ["Ramya", "Kavi"];
var myTrainee = arr1.concat(arr2, arr3);
// Concatenates arr1 with arr2 and arr3
```

concat() मेथड्स आर्ग्युमेंट्स म्हणून व्हॅल्यू देखील घेऊ शकते.

#### उदाहरण 27 (व्हॅल्यू सह अॅरे विलीन करणे)

```
var arr1 = ["Priya", "Mythili"];
var myTrainee = arr1.concat(["Ramya", "Kavi"]);
```

### अॅरेचे तुकडे करणे

स्लाइस() मेथड्स अॅरेच्या तुकड्याला नवीन अॅरेमध्ये कापते. हे उदाहरण अॅरे एलिमेंट 2 ("DTPO") पासून सुरू होणाऱ्या अॅरेचा काही पार्ट कापते. स्लाइस() मेथड्स

नवीन अॅरे तयार करते. हे सोर्स अॅरेमधून कोणतेही एलिमेंट काढून टाकत नाही.

#### उदाहरण 28

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];
var trade1 = trade.slice(2);
```

स्लाइस() मेथड्स स्लाइस (1, 3) सारखी दोन आर्ग्युमेंट घेऊ शकते. मेथड्स नंतर स्टार्ट आर्ग्युमेंट्स आणि शेवटच्या युक्तिवादापर्यंत (परंतु

समाविष्ट नाही) एलिमेंट निवडते.

#### उदाहरण 29

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];
var trade1 = trade.slice(1,3);
```

शेवटचा आर्ग्युमेंट्स वगळल्यास, पहिल्या उदाहरणांप्रमाणे, स्लाइस() मेथड्स उर्वरित अॅरेचे तुकडे करते.

#### उदाहरण 30

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];
var trade1 = trade.slice(2);
```

### ऑटोमॅटिक toString()

जेव्हा एखादे Primitive व्हॅल्यू अपेक्षित असते तेव्हा JavaScript ऑटोमॅटिक अॅरेला स्वल्पविरामाने विभक्त केलेल्या स्ट्रिंगमध्ये रूपांतरित करते. जेव्हा तुम्ही अॅरे आउटपुट करण्याचा प्रयत्न करता तेव्हा हे नेहमीच होते.

ही दोन उदाहरणे समान रिझल्ट देईल:

#### उदाहरण 31

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];
document.getElementById("demo").innerHTML =
trade.toString();
```

#### उदाहरण 32

```
var trade = ["COPA", "IT", "DTPO", "CHNM"];
document.getElementById("demo").innerHTML =
trade;
```

टीप: सर्व JavaScript ऑब्जेक्ट्समध्ये toString() मेथड्स असते.

### अॅरे क्रमवारी (सॉर्टिंग) लावत आहे

sort() मेथड्स अॅरेला अक्षरानुसार क्रमवारी (सॉर्टिंग) लावते.

#### उदाहरण 33

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];
trade.sort(); // Sorts the elements of trade
```

## रिझल्ट

CHNM, COPA, ICTSM, IT

### अॅरे उलट करत आहे

रिव्हर्स() मेथड्स अॅरेमधील एलिमेंटना उलट करते. तुम्ही अॅरेला उतरत्या क्रमाने क्रमवारी (सॉर्टिंग) लावण्यासाठी वापरू शकता.

#### उदाहरण 34

```
var trade = ["COPA", "IT", "ICTSM", "CHNM"];
trade.sort(); // Sorts the elements of trade
trade.reverse(); // Reverse the order of the elements
```

# JavaScript मध्ये ॲनिमेशन आणि मल्टीमीडिया फाइल्सच्या संकल्पना (Concepts of Animation and Multimedia files in JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- JavaScript मधील ॲनिमेशन सेटिंग्ज जाणून घ्या
- JavaScript मध्ये मल्टीमीडिया स्पष्ट करा.

## ॲनिमेशन

### एलिमेंटची स्टाइल करणे

ॲनिमेशन शक्य करण्यासाठी, ॲनिमेटेड एलिमेंट "पैरेंट कंटेनर" च्या रिलेटिव्ह ॲनिमेटेड असणे आवश्यक आहे.

कंटेनर एलिमेंट स्टाइल = "स्थिती: रिलेटिव्ह " सह तयार केला पाहिजे.

ॲनिमेशन एलिमेंट स्टाइल = "स्थिती: परिपूर्ण" सह तयार केला पाहिजे.

### उदाहरण

```
<!doctype html>
<html>
<style>
#myContainer {
width: 400px;
height: 400px;
position: relative;
background: pink;
}
#myAnimation {
width: 50px;
height: 50px;
position: absolute;
background: green;
}
</style>
<body>
<h1>My First JavaScript Animation</h1>
<div id="myContainer">
<div id="myAnimation"></div>
</div>
</body>
</html>
?
```

## ॲनिमेशन कोड

JavaScript ॲनिमेशन एलिमेंटच्या स्टाइल मध्ये क्रमिक बदल प्रोग्रामिंगद्वारे केले जातात. बदल टाइमरद्वारे कॉल केले जातात. जेव्हा टाइमर इंटरवल कमी असते, तेव्हा ॲनिमेशन सतत दिसते. बेसिक कोड आहे:

### उदाहरण

```
var id = setInterval(frame, 5);
function frame() {
if (/* test for finished */) {
clearInterval(id);
} else {
/* code to change the element style */
}
}
```

## JavaScript वापरून ॲनिमेशन तयार करा

### उदाहरण

```
<style>
#myContainer {
width: 400px;
height: 400px;
position: relative;
background: pink;
}
#myAnimation {
width: 50px;
height: 50px;
position: absolute;
background-color: green;
}
</style>
<body>
<p>
<button onclick="myMove()">Click Me</button>
```

```

</p>
<div id="myContainer">
<div id="myAnimation"></div>
</div>
<script>
function myMove() {
var elem = document.getElementById("myAnimation");
var pos = 0;
var id = setInterval(frame, 10);
function frame() {
if (pos == 350) {
clearInterval(id);
} else {
pos++;
elem.style.top = pos + 'px';
elem.style.left = pos + 'px';
}
}
}
</script>
</body>
</html>

```

## मल्टीमीडिया फाइल्स

### मल्टीमीडिया म्हणजे काय?

मल्टीमीडिया अनेक वेगवेगळ्या फॉर्मॅटमध्ये येतो. आपण ऐकू किंवा पाहू शकता हे जवळजवळ काहीही असू शकते. वेब पेजमध्ये अनेकदा विविध टाइप चे आणि स्वरूपांचे मल्टीमीडिया एलिमेंट असतात.

उदाहरणे: प्रतिमा, संगीत, ध्वनी, व्हिडिओ, रेकॉर्ड, पिक्चर पट, ॲनिमेशन आणि बरेच काही.

### मल्टीमीडिया फॉर्मॅट

मल्टीमीडिया एलिमेंट (ऑडिओ किंवा व्हिडिओसारखे) मीडिया फाइल्समध्ये स्टोर केले जातात. फाईलचा टाइप सर्चण्याचा सर्वात सामान्य पाथ म्हणजे फाइल एक्सपान्शन पाहणे. मल्टीमीडिया फाइल्समध्ये फॉर्मॅट आणि विविध एक्सपान्शन असतात जसे: .swf, .wav, .mp3, .mp4, .mpg, .wmv आणि .avi.

### Playing Videos in HTML

To show a video in HTML, use the <video> element:

### Example

```

<video width="320" height="240" controls>
<source src="movie.mp4" type="video/mp4">

```

```

<source src="movie.ogg" type="video/ogg">
Your browser does not support the video tag.
</video>

```

### हे कसे कार्य करते

कंट्रोल स्पेशल व्हिडिओ कंट्रोल जोडते, जसे की प्ले, पॉज आणि व्हॉल्यूम. नेहमी रुंदी आणि उंचीचे प्रॉपर्टीज समाविष्ट करणे चांगली कल्पना आहे. उंची आणि रुंदी सेट न केल्यास, व्हिडिओ लोड होत असताना पृष्ठ चमकू शकते. <सोर्स > एलिमेंट तुम्हाला पर्यायी व्हिडिओ फाइल्स स्पेसिफाइड करण्याची परवानगी देतो ज्यामधून ब्राउझर निवडू शकतो. ब्राउझर प्रथम ओळखले जाणारे फॉर्मॅट वापरेल. <video> आणि </video> टॅगमधील टेक्स्ट केवळ <video> एलिमेंटला सपोर्ट न करणाऱ्या ब्राउझरमध्ये प्रदर्शित केला जाईल.

HTML <video> Autoplay

व्हिडिओ सुरू करण्यासाठी ऑटोप्ले स्पेशल ता वापरा:

### उदाहरण

```

<video width="320" height="240" autoplay>
<source src="movie.mp4" type="video/mp4">
<source src="movie.ogg" type="video/ogg">

```

Your browser does not support the video tag.

टीप: ऑटोप्ले स्पेशल ता iPad आणि iPhone सारख्या मोबाइल डिव्हाइसमध्ये काम करत नाही

### HTML व्हिडिओ - मीडिया टाइप

फाइल फॉर्मॅट	माध्यम टाइप
फाइल फॉर्मॅट	माध्यम टाइप
MP4	व्हिडिओ/mp4
WebM	व्हिडिओ/वेबएम
Ogg	व्हिडिओ/ogg

### HTML व्हिडिओ - मेथड्स , प्रॉपर्टीज आणि इव्हेंट्स

HTML5 <video> एलिमेंटसाठी DOM मेथड्स , प्रॉपर्टीज आणि इव्हेंट परिभाषित करते. हे तुम्हाला व्हिडिओ लोड करण्यास, प्ले करण्यास आणि विराम देण्याची तसेच कालावधी आणि व्हॉल्यूम सेट करण्यास अनुमती देते. काही DOM इव्हेंट देखील आहेत जे तुम्हाला सूचित करू शकतात जेव्हा एखादा व्हिडिओ प्ले करणे सुरू होते, विराम दिला जातो, इ.

## HTML5 व्हिडिओ टॅग

टॅग करा	डिस्क्रिप्शन
<video>	व्हिडिओ क्वि पक्किचर पट परभाषति करते
<source>	मीडिया एलिमेंटसाठी मल्टपिल मीडिया रसिोर्सेस परभाषति करते, जसे की <व्हिडिओ> आण्ि <ऑडिओ>
<track>	मीडियाप्लेअरमध्ये टेक्स्ट ट्रॅक परभाषति करते

## वेबवरील ऑडिओ

HTML5 <audio> एलिमेंट वेब पृष्ठामध्ये ऑडिओ एम्बेड करण्याचा स्टॅण्डर्ड पाथ स्पेसिफाइड करतो.

The HTML <audio> Element

HTML मध्ये ऑडिओ फाइल प्ले करण्यासाठी, <audio> एलिमेंट वापरा:

## उदाहरण

<audio controls>

<source src="horse.ogg" type="audio/ogg">

<source src="horse.mp3" type="audio/mpeg">

Your browser does not support the audio element.

</audio>

## HTML ऑडिओ - हे कसे कार्य करते

कंट्रोल स्पेशल ता ऑडिओ कंट्रोल जोडते, जसे की प्ले, पॉज आणि व्हॉल्यूम. <source> एलिमेंट तुम्हाला पर्यायी ऑडिओ फाइल्स स्पेसिफाइड करण्याची परवानगी देतो ज्यामधून ब्राउझर निवडू शकतो. ब्राउझर प्रथम ओळखले जाणारे फॉर्मेट वापरेल. <audio> आणि </audio> टॅगमधील टेक्स्ट केवळ <audio> एलिमेंटला सपोर्ट न करणाऱ्या ब्राउझरमध्ये प्रदर्शित केला जाईल.

## HTML ऑडिओ - मीडिया टाइप

फाइल फॉर्मेट	माध्यम टाइप
MP3	ऑडिओ/एमपीईजी
OGG	ऑडिओ/ओजीजी
WAV	ऑडिओ/wav

## HTML ऑडिओ - मेथड्स , प्रॉपर्टीज आणि इव्हेंट्स

HTML5 <ऑडिओ> एलिमेंटसाठी DOM मेथड्स , प्रॉपर्टीज आणि इव्हेंट परिभाषित करते. हे तुम्हाला ऑडिओ लोड करण्यास, प्ले करण्यास आणि विराम देण्यास तसेच कालावधी आणि व्हॉल्यूम सेट करण्यास अनुमती देते. असे DOM इव्हेंट देखील आहेत जे तुम्हाला सूचित करू शकतात जेव्हा एखादा ऑडिओ प्ले करणे सुरू होते, विराम दिला जातो इ.

## HTML5 ऑडिओ टॅग

टॅग करा	डिस्क्रिप्शन
<ऑडिओ>	साउंड्स कंटेंट परभाषति करते
<सोर्स >	मीडिया एलिमेंटसाठी मल्टपिल मीडिया रसिोर्सेस परभाषति करते, जसे की <व्हिडिओ> आण्ि <ऑडिओ>



## JavaScript वापरून डायनॅमिक HTML पेजेज विकसित करा (Develop dynamic HTML pages using JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कार्य परिभाषित करा
- वर्किंग फंक्शन स्पष्ट करा
- फंक्शन वापरण्याचे फायदे स्पष्ट करा
- व्हेरिबल्सची व्याप्ती स्पष्ट करा.

**JavaScript फंक्शन्स:** JavaScript फंक्शन हे स्पेसिफिक कार्य करण्यासाठी डिझाइन केलेले कोडचे ब्लॉक आहे.

JavaScript फंक्शन कार्यान्वित केले जाते जेव्हा “काहीतरी” त्यास आवाहन करते (त्याला कॉल करते).

### उदाहरण 1

```
function myFunction(p1, p2) {
  return p1 * p2; // the function returns the product of p1
  and p2
}
```

### JavaScript फंक्शन सिंटॅक्स

JavaScript फंक्शन फंक्शन कीवर्डसह परिभाषित केले जाते, त्यानंतर नाव आणि त्यानंतर कंस () .

फंक्शनच्या नावांमध्ये अक्षरे, अंक, अंडरस्कोअर आणि डॉलर सिम्बॉल्स असू शकतात (व्हेरिबल्ससारखेच नियम).

कंसात स्वल्पविरामाने विभक्त केलेल्या पॅरामीटर नावांचा समावेश असू शकतो: (पॅरामीटर1, पॅरामीटर2, ...)

कार्यान्वित होणारा कोड, फंक्शनद्वारे, कुरळे कंसात ठेवला जातो: {}

```
functionName(parameter1, parameter2, parameter3) {
  code to be executed
}
```

कार्य **पॅरामीटर्स** फंक्शन व्याख्येमध्ये लिस्ट बद्ध केलेली नावे आहेत. कार्य **आर्ग्यूमेंट्स** फंक्शन द्वारे प्राप्त केलेली वास्तविक व्हॅल्यू आहेत जेव्हा ते मागवले जाते. फंक्शनच्या आत, आर्ग्यूमेंट्स लोकल व्हेरिबल्स म्हणून वापरले जातात. फंक्शन हे a सारखेच आहे **कार्यमेथड्स** किंवा **सबरूटीन**, इतर प्रोग्रामिंग भाषांमध्ये.

### फंक्शन इनव्होकेशन

जेव्हा “काहीतरी” फंक्शनला कॉल करते तेव्हा फंक्शनमधील कोड कार्यान्वित होईल.

- जेव्हा एखादी इवेंट्स घडते (जेव्हा युजर बटण क्लिक करतो)

- जेव्हा ते JavaScript कोडवरून मागवले जाते (कॉल केले जाते).
- ऑटोमेटिक (स्वतःला बोलावलेले)

### फंक्शन रिटर्न

जेव्हा JavaScript रिटर्न स्टेटमेंटवर पोहोचते, तेव्हा फंक्शन कार्यान्वित करणे थांबवेल. स्टेटमेंटमधून फंक्शन सुरू केले असल्यास, इनव्हॉकिंग स्टेटमेंटनंतर कोड कार्यान्वित करण्यासाठी JavaScript “परत” येईल. फंक्शन्स अनेकदा रिटर्न व्हॅल्यूची गणना करतात. returns व्हॅल्यू “कॉलर” ला परत “परत” केले जाते:

### उदाहरण 2

Calculate the product of two numbers, and return the result:

```
var x = myFunction(4, 3); // Function is called, return
value will end up in x
```

```
function myFunction(a, b) {
  return a * b; // Function returns the product of a and b }
```

x मध्ये रिझल्ट होईल: 12

### फंक्शन का?

तुम्ही कोड पुन्हा वापरू शकता: कोड एकदा परिभाषित करा आणि तो अनेक वेळा वापरा. भिन्न रिझल्ट देण्यासाठी, तुम्ही समान कोड वेगवेगळ्या युक्तिवादांसह अनेक वेळा वापरू शकता.

### उदाहरण 3

फॅरेनहाइटचे सेल्सिअसमध्ये रूपांतर करा:

```
function toCelsius(fahrenheit) {
  return (5/9) * (fahrenheit-32);
}
```

### JavaScript फंक्शन्स ऑब्जेक्ट्स आहेत

JavaScript मध्ये, फंक्शन्स ऑब्जेक्ट्स असतात. JavaScript फंक्शन्समध्ये प्रॉपर्टीज आणि मेथड्स असतात. फंक्शन्समध्ये तुम्ही तुमचे स्वतःचे प्रॉपर्टीज आणि मेथड्स जोडू शकता.

## JavaScript फंक्शन्स सुद्धा व्हेरिबल्स आहेत

JavaScript मध्ये, फंक्शन्स व्हेरिबल्स म्हणून वापरली जाऊ शकतात:

### उदाहरण 4

Instead of:

```
temp = toCelsius(32);
```

```
text = "The temperature is " + temp + " Centigrade";
```

You can use:

```
text = "The temperature is " + toCelsius(32) + " Centigrade";
```

JavaScript फंक्शन्स सामान्य व्हेरिबल्सप्रमाणे पुन्हा परिभाषित केले जाऊ शकतात. हे इतर फंक्शन्सना व्हॅल्यू म्हणून देखील पास केले जाऊ शकते.

### JavaScript स्कोप

स्कोप हा व्हेरिबल्सचा संच आहे ज्यामध्ये तुम्हाला एक्सेस आहे.

JavaScript मध्ये, ऑब्जेक्ट्स आणि फंक्शन्स देखील व्हेरिबल्स आहेत. JavaScript मध्ये, स्कोप हा व्हेरिबल्स, ऑब्जेक्ट्स आणि फंक्शन्सचा संच आहे ज्यामध्ये तुम्हाला एक्सेस आहे. JavaScript मध्ये फंक्शन स्कोप आहे: फंक्शन्समध्ये स्कोप बदलतो.

### लोकल JavaScript व्हेरिबल्स

JavaScript फंक्शनमध्ये घोषित व्हेरिबल्स, फंक्शनसाठी लोकल बनतात. लोकल व्हेरीबल ना लोकल व्याप्ती आहे. ते फंक्शनमध्येच एक्सेस करू शकतात.

### उदाहरण 5

```
// code here can not use train
```

```
function myFunction() {
```

```
var train = "Jan Satabdi";
```

```
// code here can use train
```

```
}
```

लोकल व्हेरिबल्स फक्त त्यांच्या फंक्शन्समध्येच ओळखले जातात, त्याच नावाचे व्हेरिबल्स वेगवेगळ्या फंक्शन्समध्ये वापरले जाऊ शकतात. फंक्शन सुरू झाल्यावर लोकल व्हेरिबल्स तयार होतात आणि फंक्शन पूर्ण झाल्यावर हटवले जातात.

### ग्लोबल JavaScript व्हेरिबल्स

फंक्शनच्या बाहेर घोषित व्हेरिबल्स, ग्लोबल बनते. ग्लोबल व्हेरिबलला ग्लोबल व्याप्ती आहे: वेब पृष्ठावरील सर्व स्क्रिप्ट आणि फंक्शन त्यात एक्सेस करू शकतात.

### उदाहरण 6

```
var train = "Jan Satabdi";
```

```
// code here can use train
```

```
function myFunction() {
```

```
// code here can use train
```

```
}
```

### ऑटोमैटिक ग्लोबल

घोषित न केलेल्या व्हेरिबलला तुम्ही व्हॅल्यू नियुक्त केल्यास, ते ऑटोमैटिक ग्लोबल व्हेरिबल होईल.

हे कोड उदाहरण ट्रेनला ग्लोबल व्हेरिबल म्हणून घोषित करेल, जरी ते फंक्शनमध्ये कार्यान्वित केले असले तरीही.

### उदाहरण 7

```
// code here can use train
```

```
function myFunction() {
```

```
train = "Jan Satabdi";
```

```
// code here can use train
```

```
}
```

### JavaScript व्हेरिबल्सचा लाइएफटाइम

JavaScript व्हेरिबल घोषित केल्यावर त्याचे लाइएफटाइम सुरू होते. फंक्शन पूर्ण झाल्यावर लोकल व्हेरिबल्स हटवले जातात. तुम्ही पेज बंद करता तेव्हा ग्लोबल व्हेरिबल्स हटवले जातात.

### फंक्शन आर्ग्युमेंट्स

फंक्शन आर्ग्युमेंट्स (पॅरामीटर्स) फंक्शन्समध्ये लोकल व्हेरिबल्स म्हणून काम करतात.

### HTML मध्ये ग्लोबल व्हेरिबल्स

JavaScript सह, ग्लोबल व्याप्ती हे संपूर्ण JavaScript वातावरण आहे.

HTML मध्ये, ग्लोबल व्याप्ती विंडो ऑब्जेक्ट आहे: सर्व ग्लोबल व्हेरिबल्स विंडो ऑब्जेक्टशी संबंधित आहेत.

### उदाहरण 8

```
// code here can use window.train
```

```
function myFunction() {
```

```
train = "Jan Satabdi";
```

```
}
```

# JavaScript मध्ये स्ट्रिंग आणि नंबर मेथड्स (String and number methods in JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- स्ट्रिंग स्पष्ट करा
- विविध स्ट्रिंग मेथड्स स्पष्ट करा.

## JavaScript स्ट्रिंग्स

JavaScript स्ट्रिंगचा वापर टेक्स्ट साठवण्यासाठी आणि हाताळण्यासाठी केला जातो. JavaScript स्ट्रिंग म्हणजे शून्य किंवा अधिक अक्षरे कोट्समध्ये लिहिलेली असतात. तुम्ही सिंगल किंवा डबल कोट्स वापरू शकता.

### उदाहरण 1

```
var cityname = "Chennai"; // Double quotes
var cityname = 'Chennai'; // Single quotes
```

जोपर्यंत ते स्ट्रिंगच्या आजूबाजूच्या कोट्सशी जुळत नाहीत तोपर्यंत तुम्ही स्ट्रिंगमध्ये कोट्स वापरू शकता.

### उदाहरण 2

```
var notes = "You're Welcome";
var ans = "Coimbatore is called 'Cotton City'";
var ans = 'Coimbatore is called "Cotton City"';
```

स्ट्रिंग लांबी प्रॉपर्टी

स्ट्रिंगची लांबी बिल्ट इन प्रॉपर्टी लांबीमध्ये आढळते.

### उदाहरण 3

```
var s = "Computer Operator and Programming Assistant";
var slen = s.length;
```

बॅकस्लॅश एस्केप कॅरक्टर .

बॅकस्लॅश (\) एस्केप कॅरक्टर स्पेशल कॅरक्टर ना स्ट्रिंग कॅरक्टर मध्ये बदलते.

कोड	रिझल्ट	डिस्क्रिप्शन
\	'	एकच कोट
"	"	दुहेरी कोट
\\	\	बॅकस्लॅश

सिक्वेंस \" स्ट्रिंगमध्ये दुहेरी अवतरण समाविष्ट करतो.

### उदाहरण 4

```
var x = "Bhubaneswar is known as the \"Temple City of India\"";
```

सिक्वेंस \" स्ट्रिंगमध्ये एकच कोट समाविष्ट करतो:

### उदाहरण 5

```
var x = 'you\'re Welcome.';
```

The sequence \\ inserts a backslash in a string:

## उदाहरण 6

```
var x = "The character \\ is called backslash.";
```

## स्ट्रिंग ऑब्जेक्ट्स असू शकतात

साधारणपणे, JavaScript स्ट्रिंग ही Primitive व्हॅल्यू असतात, जी अक्षरांपासून तयार केली जातात.

```
var tName = "Veni";
```

परंतु स्ट्रिंग्सना नवीन कीवर्ड्ससह ऑब्जेक्ट्स म्हणून देखील परिभाषित केले जाऊ शकते.

```
var tName = new String("Veni");
```

## JavaScript स्ट्रिंग मेथड्स

स्ट्रिंग मेथड्स आपल्याला स्ट्रिंग्ससह कार्य करण्यास मदत करतात.

## स्ट्रिंगमध्ये स्ट्रिंग सर्चणे

**indexOf()** method

**indexOf()** मेथड्स स्ट्रिंगमधील स्पेसिफाइड टेक्स्ट च्या पहिल्या घटनेची इंडेक्स (स्थिती) परत करते:

## उदाहरण 7

```
var str = "When I do good I feel good ";
var num = str.indexOf("good");
```

## रिझल्ट :

व्हेरिएबलमध्ये पोजिशन व्हॅल्यू 10 आहे.

टीप: JavaScript शून्य वरून पोजिशन मोजते. 0 हे स्ट्रिंगमधील पहिले पोजिशन आहे, 1 हे दुसरे आहे, 2 हे तिसरे आहे ...

**lastIndexOf()** method

**lastIndexOf()** मेथड्स स्ट्रिंगमधील स्पेसिफाइड टेक्स्ट च्या शेवटच्या घटनेची इंडेक्स (position) परत करते.

## उदाहरण 8

```
var str = "When I do good I feel good";
var num = str.lastIndexOf("good");
```

## रिझल्ट :

व्हेरिएबलमध्ये पोजिशन व्हॅल्यू 22 आहे.

दोन्ही **indexOf()**, आणि **lastIndexOf()** टेक्स्ट न मिळाल्यास -1 परत करतात.

## उदाहरण 9

```
var s1r = "When I do good I feel good";
```

```
var num = s1r.lastIndexOf("better");
```

### रिझल्ट :

व्हेरिएबलमध्ये पोजिशन व्हॅल्यू -1 आहे.

दोन्ही मेथड्स शोधासाठी स्टार्टिंग स्थिती म्हणून दुसरे पॅरामीटर स्वीकारतात.

## उदाहरण 10

```
var s1r = "When I do good I feel good";
```

```
var num = s1r.indexOf("good", 15);
```

### रिझल्ट :

व्हेरिएबलमध्ये पोजिशन व्हॅल्यू 22 आहे.

स्ट्रिंगमध्ये स्ट्रिंग सर्च त आहे

Search () मेथड्स स्पेसिफाइड मूल्यासाठी स्ट्रिंग सर्च ते आणि जुळणीची स्थिती परत करते.

## उदाहरण 11

```
var s1r = "When I do good I feel good";
```

```
var num = s1r.search("good");
```

### रिझल्ट :

व्हेरिएबलमध्ये पोजिशन व्हॅल्यू 10 आहे.

टीप: सर्च () मेथड्स दुसरी स्टार्ट पोजिशन आर्ग्युमेंट घेऊ शकत नाही.

## स्ट्रिंग पार्ट काढणे

### Slice() मेथड्स

**Slice()** स्ट्रिंगचा एक पार्ट काढतो आणि काढलेला पार्ट नवीन स्ट्रिंगमध्ये परत करतो. मेथड्स सुरुवातीची स्थिती आणि शेवटची स्थिती घेते.

हे उदाहरण 14 वरून 21 व्या स्थानापर्यंत स्ट्रिंगचा एक पार्ट कापतो.

## उदाहरण 12

```
var s1r = "Hockey,Kabadi,Cricket";
```

```
var res = s1r.slice(14, 21);
```

### रिझल्ट :

रिझल्टचा निकाल क्रिकेटवर लागेल

पॅरामीटर निगेटिव्ह असल्यास, पोजिशन स्ट्रिंगच्या शेवटी मोजले जाते. हे उदाहरण पोजिशन -14 ते पोजिशन -8 पर्यंत स्ट्रिंगच्या एका भागाचे तुकडे करते.

## उदाहरण 13

```
var s1r = "Hockey,Kabadi,Cricket";
```

```
var res = s1r.slice(-14, -8);
```

### रिझल्ट :

Res चा रिझल्ट होईल कबड्डी

तुम्ही दुसरे पॅरामीटर वगळल्यास, मेथड्स उर्वरित स्ट्रिंगचे तुकडे करेल

## उदाहरण 14

```
var res = s1r.slice(7);
```

### रिझल्ट :

रिझल्ट कबड्डी, क्रिकेट असेल

The substring() Method

substring() Slice() सारखेच आहे. फरक असा आहे की substring() नकारात्मक निर्देशांक स्वीकारू शकत नाही.

## उदाहरण 15

```
var s1r = "Hockey,Kabadi,Cricket";
```

```
var res = s1r.substring(7, 13);
```

### रिझल्ट :

Res चा निकाल कबड्डीत लागेल

### substr() मेथड्स

तुम्ही दुसरा पॅरामीटर वगळल्यास, substring() उर्वरित स्ट्रिंग कापेल. substr() मेथड्स **substr()** slice() सारखीच असते. फरक हा आहे की दुसरा पॅरामीटर काढलेल्या भागाची लांबी स्पेसिफाइड करते.

## उदाहरण 16

```
var s1r = "Hockey,Kabadi,Cricket";
```

```
var res = s1r.substr(7, 6);
```

### रिझल्ट :

Res चा रिझल्ट होईल कबड्डी

तुम्ही दुसरे पॅरामीटर वगळल्यास, substr() उर्वरित स्ट्रिंग कापून टाकेल.

## उदाहरण 17

```
var s1r = "Hockey,Kabadi,Cricket";
```

```
var res = s1r.substr(7);
```

### रिझल्ट :

Res चा रिझल्ट होईल कबड्डी, क्रिकेट

जर पहिला पॅरामीटर निगेटिव्ह असेल, तर पोजिशन स्ट्रिंगच्या शेवटी मोजले जाते.

## उदाहरण 18

```
var str = "Hockey,Kabadi,Cricket";
```

```
var res = str.substr(-7);
```

## रिझल्ट :

Res चा रिझल्ट होईल क्रिकेट

## स्ट्रिंग कंटेंट बदलत आहे

**replace()** मेथड स्ट्रिंगमधील एका विशेषिफाइड मूल्याला दुसऱ्या मूल्यासह बदलते.

### उदाहरण 19

```
str = "I like custard apple";
```

```
var newstr = str.replace("custard apple", "mango");
```

## रिझल्ट :

चा रिझल्ट newstr असेल मला आंबा आवडेल

**टीप:** **replace()** मेथड्स कॉल केलेली स्ट्रिंग बदलत नाही. ते एक नवीन स्ट्रिंग परत करते.

**डीफॉल्टनुसार, रिप्लेस() फंक्शन फक्त पहिल्या मॅचची जागा घेते.**

डीफॉल्टनुसार, रिप्लेस() फंक्शन केस सेन्सिटिव्ह असते. CUSTARD APPLE (अपर-केससह) लिहून चालणार नाही.

केस असेन्सिटिव्ह बदलण्यासाठी, /i ध्वज (असेन्सिटिव्ह ) सह नियमित अभिव्यक्ती वापरा.

### उदाहरण 20

```
str = "I like custard apple";
```

```
var newstr = str.replace(/CUSTARD APPLE/i, "Mango");
```

## रिझल्ट :

newstr चा रिझल्ट होईल मला आंबा आवडतो

लक्षात घ्या की रेग्युलर एक्सप्रेशनस कोट्सशिवाय लिहिलेले आहेत.

सर्व सामने बदलण्यासाठी, a / g ध्वज (ग्लोबल मॅच) सह नियमित अभिव्यक्ती वापरा.

### उदाहरण 21

```
str = " I like custard apple and apple";
```

```
var newstr = str.replace(/apple/g, "Mango");
```

## रिझल्ट :

newstr चा रिझल्ट मला आवडेल कस्टर्ड आंबा आणि आंबा

## अप्पर आणि लोअर केसमध्ये रूपांतरित करणे

**toUpperCase()** सह स्ट्रिंग अपरकेसमध्ये रूपांतरित केली जाते

### उदाहरण 22

```
var str1 = "Information Technology";
```

```
var str2 = str1.toUpperCase();
```

## रिझल्ट :

str2 चा रिझल्ट INFORMATION TECHNOLOGY असेल

toLowerCase() सह स्ट्रिंग लोअरकेसमध्ये रूपांतरित केली जाते.

### उदाहरण 23

```
var str1 = "INFORMATION TECHNOLOGY";
```

```
var str2 = str1.toLowerCase();
```

## रिझल्ट :

str2 चा रिझल्ट इनफार्मेशन तंत्रज्ञान असेल

**concat()** मेथड्स

**concat()** दोन किंवा अधिक स्ट्रिंग्स जोडते.

### उदाहरण 24

```
var txt1 = "Mr";
```

```
var txt2 = "Selvaraj";
```

```
var txt3 = txt1.concat(".", txt2);
```

## रिझल्ट :

txt3 चा रिझल्ट असेल सेल्वराज श्री

**concat()** प्लस ऑपरेटर ऐवजी मेथड्स वापरली जाऊ शकते. या दोन ओळी तेच करतात.

### उदाहरण 25

```
var txt = "Mr" + "." + "Selvaraj";
```

```
var txt = "Mr".concat(".", "Selvaraj");
```

String.trim()

String.trim() स्ट्रिंगच्या दोन्ही बाजूंनी व्हाइटस्पेस काढून टाकते.

### उदाहरण 26

```
var str = " India Gate ";
```

```
alert(str.trim());
```

## रिझल्ट :

इंडिया गेट अॅलर्ट बॉक्समध्ये अग्रभागी आणि मागे रिकाम्या जागेशिवाय प्रदर्शित केले जाते.

## स्ट्रिंग कॅरेक्टर्स काढत आहे

**charAt()** मेथड्स स्ट्रिंगमधील स्पेसिफाइड निर्देशांकावर कॅरेक्टर परत करते.

### उदाहरण 27

```
var str = "Thirumalai Nayakkar Mahal";
```

```
str.charAt(0); // returns T
```

**charCodeAt()** मेथड्स स्ट्रिंगमधील स्पेसिफाइड निर्देशांकावर वर्णाचा युनिकोड परत करते. मेथड्स 0 आणि 65535 मधील UTF-16 शंक्रू पूर्णांक मिळवते.

## उदाहरण 28

```
var str = "Hill Station";  
str.charCodeAt(0); // returns 72
```

## स्ट्रिंगला अरेंमध्ये रूपांतरित करणे

split() मेथडस सह स्ट्रिंग अरेंमध्ये रूपांतरित केली जाऊ शकते.

## उदाहरण 29

```
var txt1 = "Kovai,Nellai,Madurai"; // String  
var txt2 = txt1.split(","); // Split on commas
```

## रिझल्ट :

txt2 चा रिझल्ट **Kovai** असेल

विभाजक वगळल्यास, परत केलेल्या अरेंमध्ये इंडेक्स [0] मध्ये संपूर्ण स्ट्रिंग असेल.  
विभाजक "" असल्यास, परत केलेला अरें सिंगल कॅरक्टर चा अरें असेल.

## उदाहरण 30

```
var txt = "Anaimai"; // String  
txt.split(""); // Split in characters
```

## उदाहरण 31

```
<!DOCTYPE html>  
<html>  
<body>  
<p id="demo"></p>  
<script>  
var str = "Temple";  
var arr = str.split("");  
var text = "";  
var i;  
for (i = 0; i < arr.length; i++) {  
    text += arr[i] + "<br>"  
}  
document.getElementById("demo").innerHTML = text;  
</script>  
</body>  
</html>
```

## रिझल्ट :

T  
e  
m  
p  
l  
e  
?

## JavaScript क्रमांक मेथड्स

नंबर मेथड्स आपल्याला नंबर सह कार्य करण्यास मदत करतात.

## नंबर मेथड्स आणि प्रॉपर्टीज

2018 किंवा 1.44 सारख्या Primitive व्हॅल्यु मध्ये प्रॉपर्टीज आणि मेथड्स असू शकत नाहीत कारण ते ऑब्जेक्ट नाहीत.

परंतु JavaScript सह, मेथड्स आणि प्रॉपर्टीज Primitive व्हॅल्यु साठी देखील सबलब्ध आहेत, कारण मेथड्स आणि प्रॉपर्टीज कार्यान्वित करताना JavaScript Primitive व्हॅल्यु ना वस्तू मानते.

The **toString()** Method

**toString()** नंबर स्ट्रिंग म्हणून परत करते. सर्व नंबर मेथड्स कोणत्याही टाइप च्या नंबर वर (अक्षर, चल किंवा अभिव्यक्ती) वापरल्या जाऊ शकतात.

## उदाहरण 32

```
var n = 2018;  
n.toString(); // returns 2018 from variable x  
(2018).toString(); // returns 2018 from literal 2018  
(2000+18).toString(); // returns 2018 from expres-  
sion 2000 + 18
```

## toExponential() मेथड्स

**toExponential()** घातांकीय नोटेशन वापरून गोलाकार आणि लिखित संख्येसह स्ट्रिंग मिळवते. एक पॅरामीटर दशांश पॉइंट च्या मागे कॅरक्टर ची नंबर परिभाषित करते.

## उदाहरण 33

```
var x = 3. 869;  
x.toExponential(2); // returns 3.87e+0  
x.toExponential(4); // returns 3.8690e+0  
x.toExponential(6); // returns 3.869000e+0
```

पॅरामीटर पर्यायी आहे. तुम्ही ते स्पेसिफाइड न केल्यास, JavaScript नंबर पूर्ण करणार नाही.

The **toFixed()** Method

**toFixed()** स्पेसिफाइड दशांश संख्येसह लिहिलेल्या संख्येसह स्ट्रिंग मिळवते.

## उदाहरण 34

```
var x = 3. 869;  
x.toFixed(0); // returns 4  
x.toFixed(2); // returns 3.87  
x.toFixed(4); // returns 3.8690  
x.toFixed(6); // returns 3.869000
```

टीप: **toFixed(2)** पैशासह काम करण्यासाठी योग्य आहे.

The **toPrecision()** Method

**toPrecision()** स्पेसिफाइड लांबीसह लिहिलेल्या संख्येसह स्ट्रिंग मिळवते.

### उदाहरण 35

```
var x = 3. 869;
x.toPrecision(); // returns 3. 869
x.toPrecision(2); // returns 3.9
x.toPrecision(4); // returns 3. 869
x.toPrecision(6); // returns 3. 86900
```

The **valueOf()** Method

**valueOf()** नंबर म्हणून नंबर परत करते.

### उदाहरण 36

```
var x = 451;
x.valueOf(); // returns 451 from variable x
(451).valueOf(); // returns 451 from literal 451
(400 + 51).valueOf(); // returns 451 from expression 400 + 51
```

### व्हेरिएबल्सचे नंबर मध्ये रूपांतर करणे

3 JavaScript मेथड्स आहेत ज्याचा वापर व्हेरिएबल्सला नंबर मध्ये रूपांतरित करण्यासाठी केला जाऊ शकतो:

- 1 Number() मेथड्स
- 2 parseInt() मेथड्स
- 3 parseFloat() मेथड्स

या मेथड्स नंबर मेथड्स नसून ग्लोबल JavaScript मेथड्स आहेत.

### ग्लोबल JavaScript मेथड्स

JavaScript ग्लोबल मेथड्स सर्व JavaScript डेटा प्रकारांवर वापरल्या जाऊ शकतात. नंबर सह कार्य करताना या सर्वात संबंधित मेथड्स आहेत.

मेथड्स	डिस्क्रिप्शन
Number()	त्याच्या युक्तिवादातून रूपांतरित केलेली नंबर मिळवते
parseInt()	त्याच्या युक्तिवादाचे विश्लेषण करते आणि पूर्णांक मिळवते
parseFloat()	त्याचा आर्ग्यूमेंट्स पार्स करते आणि फ्लोटिंग पॉइंट नंबर मिळवते

The **Number()** Method

**Number()** JavaScript व्हेरिएबल्सला नंबर मध्ये रूपांतरित करण्यासाठी वापरले जाऊ शकते.

### उदाहरण 37

```
Number(true); // returns 1
Number(false); // returns 0
Number("25"); // returns 25
Number(" 25"); // returns 25
Number("25 "); // returns 25
Number("25.77"); // returns 25.77
Number("25,77"); // returns NaN
Number("2577"); // returns NaN
Number("IT"); // returns NaN
```

टीप : जर नंबर बदलता येत नसेल तर, NaN (नंबर नाही) परत केला जातो.

तारखांना वापरलेली Number() मेथड्स. Number() तारखेला क्रमांकामध्ये रूपांतरित देखील करू शकते.

### उदाहरण 38

```
Number(new Date("2018-09-15")); // returns 1536969600000
```

टीप: वरील क्रमांक() मेथड्स 1.1.1970 पासून मिलिसेकंदांची नंबर मिळवते.

The **parseInt()** Method

**parseInt()** स्ट्रिंग पार्स करते आणि पूर्ण नंबर मिळवते. मोकळ्या जागांना परवानगी आहे. फक्त पहिला क्रमांक परत केला जातो.

### उदाहरण 39

```
parseInt("25"); // returns 25
parseInt("25.33"); // returns 25
parseInt("25 20 30"); // returns 25
parseInt("25 years"); // returns 25
parseInt("years 25"); // returns NaN
```

### Try it yourself "

जर नंबर बदलता येत नसेल तर, NaN (नंबर नाही) परत केला जातो.

**parseFloat()** Method

**parseFloat()** स्ट्रिंग पार्स करते आणि नंबर मिळवते. मोकळ्या जागांना परवानगी आहे. फक्त पहिला क्रमांक परत केला जातो:

### उदाहरण 40

```
parseFloat("25"); // returns 25
parseFloat("25.77"); // returns 25.77
parseFloat("255075"); // returns 25
parseFloat("25 years"); // returns 25
parseFloat("years 25"); // returns NaN
```

टीप : जर नंबर बदलता येत नसेल तर, NaN (नंबर नाही) परत केला जातो.

?

नंबर प्रॉपर्टीज

प्रॉपर्टी	डिस्क्रिप्शन
MIN_VALUE	JavaScript मध्ये शक्य तितकी लहान नंबर मिळवते
MAX_VALUE	JavaScript मध्ये शक्य असलेली सर्वात मोठी नंबर मिळवते
POSITIVE_INFINITY	अनंताचे प्रतिनिधित्व करते (ऑनओव्हरफ्लो परत आले)
NEGATIVE_INFINITY	नकारात्मक अनंताचे प्रतिनिधित्व करते (ओव्हरफ्लोवर परत आले)
NaN	“नॉट-ए-नंबर” मूल्याचे प्रतिनिधित्व करते

JavaScript MIN\_VALUE आणि MAX\_VALUE

उदाहरण 41

```
var n = Number.MAX_VALUE;
```

रिझल्ट :

MAX\_VALUE JavaScript मधील सर्वात मोठी संभाव्य नंबर मिळवते.

1.7976931348623157e+308

उदाहरण 42

```
var n = Number.MIN_VALUE;
```

रिझल्ट :

MIN\_VALUE JavaScript मध्ये शक्य तितकी लहान नंबर मिळवते. 5e-324

JavaScript POSITIVE\_INFINITY

उदाहरण 43

```
var n = Number.POSITIVE_INFINITY;
```

## JavaScript मधील मैथ ऑब्जेक्ट (Math objects in JavaScript)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

• जावा स्क्रिप्टमध्ये मैथ ऑब्जेक्ट समजावून सांगा.

JavaScript मैथ ऑब्जेक्ट

JavaScript Math ऑब्जेक्ट तुम्हाला नंबर वर गणितीय फंक्शन करण्यास अनुमती देते.

```
Math.PI; // returns 3.141592653589793
```

Math.round(x) x चे व्हॅल्यू त्याच्या जवळच्या पूर्णांकावर मिळवते.

ओव्हरफ्लोवर POSITIVE\_INFINITY परत केले जाते.

उदाहरण 44

```
var n = 2 / 0;
```

JavaScript NEGATIVE\_INFINITY

उदाहरण 45

```
var n = Number.NEGATIVE_INFINITY;
```

ओव्हरफ्लोवर NEGATIVE\_INFINITY परत केले जाते:

उदाहरण 46

```
var x = -1 / 0;
```

JavaScript NaN - नंबर नाही

उदाहरण 47

```
var x = Number.NaN;
```

NaN हा JavaScript आरक्षित शब्द आहे जो दर्शवितो की नंबर ही कायदेशीर नंबर नाही. नंबर नसलेल्या स्ट्रिंगसह अर्थमेटिक करण्याचा प्रयत्न केल्यास NaN (नंबर नाही).

उदाहरण 48

```
var n = 500 / "Price"; // n will be NaN (Not a Number)
```

नंबर प्रॉपर्टीज व्हेरिबल्सवर वापरले जाऊ शकत नाहीत

नंबर प्रॉपर्टीज जावास्क्रिप्टच्या नंबर ऑब्जेक्ट रॅपरशी संबंधित आहेत ज्याला नंबर म्हणतात. या गुणधर्मांमध्ये फक्त Number.MAX\_VALUE म्हणून एक्सेस केला जाऊ शकतो.

newNumber.MAX\_VALUE वापरणे, जेथे 'newNumber' एक चल, अभिव्यक्ती किंवा व्हॅल्यू आहे, अनडिफाइन परत येईल.

उदाहरण 49

```
var a = 10;
```

```
var b = a.MAX_VALUE; // b becomes undefined
```



## उदाहरण 2

Math.pow(5, 2); // returns 25

Math.sqrt(x) x चे क्लास मूळ मिळवते:

## उदाहरण 3

Math.sqrt(25); // returns 5

Math.abs(x) x चे परिपूर्ण (सकारात्मक) व्हॅल्यू मिळवते:

## उदाहरण 4

Math.abs(-3.5); // returns 3.5

Math.ceil(x) x चे व्हॅल्यू त्याच्या जवळच्या पूर्णांकापर्यंत पूर्ण करते:

## उदाहरण 5

Math.ceil(5.4); // returns 6

Math.floor(x) x चे व्हॅल्यू त्याच्या जवळच्या पूर्णांकापर्यंत परत करते:

## उदाहरण 6

Math.floor(5.8); // returns 5

Math.sin(x) returns the sine (a value between -1 and 1) of the angle x (given in radians)..

तुम्हाला रेडियन ऐवजी अंश वापरायचे असल्यास, तुम्हाला अंशांचे रेडियनमध्ये रूपांतर करावे लागेल.

रेडियनमधील कोन = अंश x PI / 180 मध्ये कोन.

## उदाहरण 7

Math.sin(90 \* Math.PI / 180); // returns 1 (the sine of 90 degrees)

Math.sin(0 \* Math.PI / 180); // returns 0 (the sine of 0 degrees)

Math.cos(x) returns the cosine (a value between -1 and 1)

of the angle x (given in radians) तुम्हाला रेडियन ऐवजी अंश वापरायचे असल्यास, तुम्हाला अंश रेडियनमध्ये रूपांतरित करावे लागतील:

रेडियनमधील कोन = अंश x PI / 180 मध्ये कोन.

## गणित ऑब्जेक्ट मेथड्स

मेथड्स	डिस्क्रिप्शन
abs(x)	x चे परिपूर्ण व्हॅल्यू मिळवते
acos(x)	x चे अर्कोसाइन रेडियनमध्ये मिळवते
asin(x)	रेडियनमध्ये x चा आर्कसाइन मिळवते
atan(x)	-PI/2 आणि PI/2radians मधील अंकीय व्हॅल्यू म्हणून x चा आर्कटॅजंट मिळवते

## उदाहरण 8

Math.cos(0 \* Math.PI / 180); // returns 1 (0 अंशांचा cos)

वितर्काच्या लिस्ट मध्ये सर्वात कमी किंवा सर्वोच्च व्हॅल्यू सर्च प्यासाठी

Math.min() आणि Math.max() वापरले जाऊ शकतात:

## उदाहरण 9

math.min(20,40,6,0,-10); // returns -10

## उदाहरण 10

Math.max(20,40,6,0,-10); // returns 40

Math.random() returns a random number between 0 (inclusive), and 1 (exclusive):

## उदाहरण 11

Math.random(); // returns a random number

## मैथ प्रॉपर्टीज (Constants)

JavaScript 8 गणितीय कॉन्स्टंट्स प्रदान करते जे असू शकतात गणित ऑब्जेक्टसह एक्सेस केला:

## उदाहरण 12

Math.E // returns Euler's number

Math.PI // returns PI

Math.SQRT2 // returns the square root of 2

Math.SQRT1\_2 // returns the square root of 1/2

Math.LN2 // returns the natural logarithm of 2

Math.LN10 // returns the natural logarithm of 10

Math.LOG2E // returns base 2 logarithm of E

Math.LOG10E // returns base 10 logarithm of E

## मैथ कन्स्ट्रक्टर

इतर ग्लोबल वस्तूंप्रमाणे, गणित ऑब्जेक्टला कोणतेही कन्स्ट्रक्टर नाही. मेथड्स आणि प्रॉपर्टीज स्थिर आहेत. सर्व मेथड्स आणि प्रॉपर्टीज (स्थिर) प्रथम गणित ऑब्जेक्ट तयार न करता वापरता येतात.

मेथड्स	डिस्क्रिप्शन
atan2(y, x)	त्याच्या वितर्काच्या पार्ट फलाचा पृष्ठपॉइंट मिळवते
ceil(x)	x चे व्हॅल्यु त्याच्या जवळच्या पूर्णांकापर्यंत पूर्ण करते
cos(x)	x चे कोसाइन मिळवते (x रेडियनमध्ये आहे)
exp(x)	E <sup>x</sup> चे व्हॅल्यु मिळवते
floor(x)	x चे व्हॅल्यु त्याच्या जवळच्या पूर्णांकापर्यंत परत करते
log(x)	x चा नैसर्गिक लॉगरिदम (बेस E) मिळवते
max(x, y, z, ..., n)	सर्वोच्च मूल्यासह नंबर मिळवते
min(x, y, z, ..., n)	सर्वात कमी मूल्यासह नंबर मिळवते
pow(x, y)	x चे व्हॅल्यु y च्या घात मिळवते
random()	0 आणि 1 मधील यादृच्छिक नंबर मिळवते
round(x)	x चे व्हॅल्यु त्याच्या जवळच्या पूर्णांकावर परत करते
sin(x)	x ची साइन मिळवते (x रेडियनमध्ये आहे)
sqrt(x)	x चे क्लास मूळ मिळवते
tan(x)	कोनाची स्पर्शिका मिळवते

## JavaScript Dates

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- JavaScript date ऑब्जेक्ट्स स्पष्ट करा
- JavaScript date फॉर्मेट स्पष्ट करा
- JavaScript date मिळवण्याच्या मेथड्स स्पष्ट करा
- JavaScript date सेट मेथड्स स्पष्ट करा.

### JavaScript date ऑब्जेक्ट्स

डीफॉल्टनुसार, JavaScript ब्राउझरचा टाइम झोन वापरेल आणि पूर्ण टेक्स्ट स्ट्रिंग म्हणून date प्रदर्शित करेल.

Thu Sep 27 2018 09:09:39 GMT+0530 (India Standard Time)

### date ऑब्जेक्ट्स तयार करणे

नवीन Date() कन्स्ट्रक्टरसह date ऑब्जेक्ट्स तयार केले जातात.

नवीन date ऑब्जेक्ट तयार करण्याचे 4 पाथ आहेत. ते आहेत

- 1 नवीन date()
- 2 नवीन date (वर्ष, महिना, दिवस, तास, मिनिटे, सेकंद, मिलीसेकंद)
- 3 नवीन date (date स्ट्रिंग)
- 4 नवीन date (मिलिसेकंद)

### 1 नवीन date()

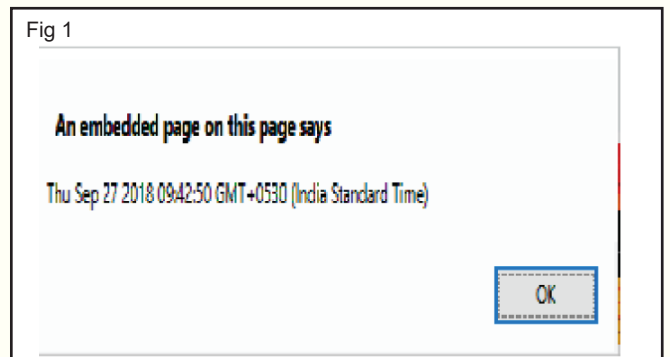
नवीन date () करंट date आणि वेळेसह नवीन date ऑब्जेक्ट तयार करते.

#### उदाहरण 1:

```
var d = new Date();
```

```
alert(d)
```

#### रिझल्ट :(आकृती क्रं 1)



टीप date ऑब्जेक्ट्स स्थिर आहेत. संगणक वेळ टिकिंग आहे, परंतु date ऑब्जेक्ट नाहीत.

## 2 नवीन date (वर्ष, महिना, ...)

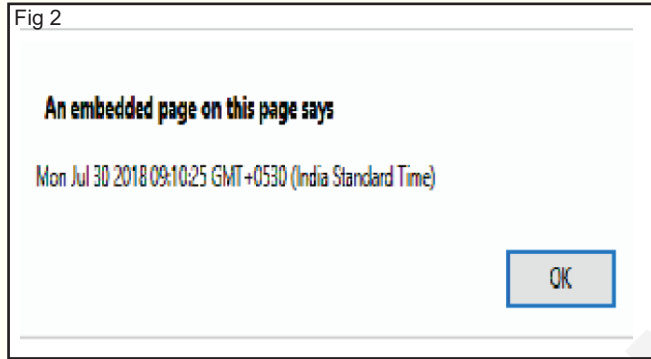
नवीन date (वर्ष, महिना, ...) स्पेसिफाइड date आणि वेळेसह नवीन date ऑब्जेक्ट तयार करते.

### उदाहरण 2

```
var d = new Date(2018, 06, 30, 09, 10, 25, 0);
```

7 नंबर वर्ष, महिना, दिवस, तास, मिनिट, सेकंद आणि मिलिसेकंद (त्या क्रमाने) स्पेसिफाइड करतात.

### रिझल्ट :(पिक्चर 2)



टीप: JavaScript 0 ते 11 महिने मोजते. जानेवारी 0 आहे. डिसेंबर 11 आहे.

6 नंबर वर्ष, महिना, दिवस, तास, मिनिट, सेकंद स्पेसिफाइड करतात.

### उदाहरण 3

```
var d = new date(2018, 06, 30, 09, 10, 25);
```

5 नंबर वर्ष, महिना, दिवस, तास आणि मिनिट स्पेसिफाइड करतात.

### उदाहरण 4

```
var d = new date(2018, 06, 30, 09, 10);
```

### उदाहरण 5

```
var d = new date(2018, 06, 30, 09);
```

3 नंबर वर्ष, महिना आणि दिवस स्पेसिफाइड करतात.

### उदाहरण 6

```
var d = new date(2018, 06, 30);
```

2 नंबर वर्ष आणि महिना स्पेसिफाइड करतात.

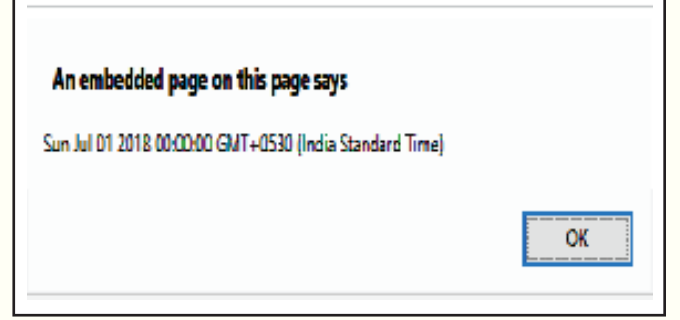
### उदाहरण 7

```
var d = new date(2018, 06);
```

### रिझल्ट : (पिक्चर 3)

आपण महिना वगळू शकत नाही. तुम्ही फक्त एक पॅरामीटर पुरवल्यास ते मिलीसेकंद मानले जाईल.

Fig 3



### उदाहरण 8

```
var d = new date(2018);
```

मागील शतक

एक आणि दोन अंकी वर्षाचा अर्थ 19xx असा केला जाईल.

### उदाहरण 9

```
var d = new date(96, 04, 12);
```

### 3 नवीन date (dateString)

नवीन date(dateString) तारखेच्या स्ट्रिंगमधून नवीन date ऑब्जेक्ट तयार करते.

### उदाहरण 10

```
var d = new Date("December 20, 2018 10:15:00");
```

### रिझल्ट :(पिक्चर 4)



टीप: JavaScript जानेवारी 01, 1970, 00:00:00 UTC (युनिव्हर्सल टाइम कोऑर्डिनेटेड) पासून तारखा मिलिसेकंदांच्या संख्येत स्टोर करते. शून्य वेळ जानेवारी 01, 1970 00:00:00 UTC आहे. आता वेळ आहे: 01 जानेवारी 1970 पूर्वी 1537962903199 मिलिसेकंद

### 4 new date (मिलिसेकंद)

new date (मिलिसेकंद) शून्य वेळ अधिक मिलीसेकंद म्हणून नवीन date ऑब्जेक्ट तयार करते.

### उदाहरण 11

```
var d = new date(0);
```

01 जानेवारी 1970 अधिक 100 000 000 मिलीसेकंद अंदाजे 02 जानेवारी 1970 आहे.

## उदाहरण 12

```
var d = new date(100000000);
```

रिझल्ट : (पिक्चर 5)

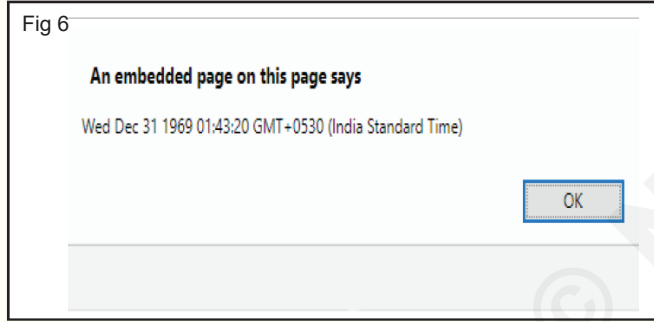


जानेवारी 01 1970 वजा 100 000 000 मिलिसेकंद म्हणजे अंदाजे 31 डिसेंबर 1969.

## उदाहरण 13

```
var d = नवीन date(-100000000);
```

रिझल्ट : (पिक्चर 6)



## उदाहरण 14

```
var d = नवीन date(86400000);
```

रिझल्ट :

Fri Jan 02 1970 05:30:00 GMT+0530 (India Standard Time)

टीप: नवीन date (मिलीसेकंद) वापरून, 1 जानेवारी, 1970, 00:00:00 युनिव्हर्सल टाइम (UTC) अधिक मिलिसेकंद म्हणून नवीन date ऑब्जेक्ट तयार करते. एक दिवस (24 तास) 86 400 000 मिलिसेकंद आहे.

## date मेथड्स

जेव्हा date ऑब्जेक्ट तयार केला जातो, तेव्हा अनेक मेथड्स आपल्याला त्यावर ऑपरेट करण्याची परवानगी देतात. date मेथड्स तुम्हाला लोकल वेळ किंवा UTC (युनिव्हर्सल, किंवा GMT) वेळ वापरून वर्ष, महिना, दिवस, तास, मिनिट, सेकंद आणि मिलिसेकंद date ऑब्जेक्ट मिळवू आणि सेट करू देतात.

## तारखा दाखवत आहे

डीफॉल्टनुसार JavaScript पूर्ण टेक्स्ट स्ट्रिंग स्वरूपात तारखा आउटपुट करेल. जेव्हा तुम्ही HTML मध्ये date ऑब्जेक्ट प्रदर्शित करता, तेव्हा ते

toString() मेथड्स सह ऑटोमैटिक स्ट्रिंगमध्ये रूपांतरित होते.

## उदाहरण 15

```
d = new Date();
```

```
alert(d);
```

Same as:

```
d = new Date();
```

```
alert(d.toString());
```

toUTCString() मेथड्स तारखेला UTC स्ट्रिंगमध्ये रूपांतरित करते.

## उदाहरण 16

```
var d = new date();
```

```
alert(d);
```

रिझल्ट : (पिक्चर 7)



toDate() मेथड्स तारखेला अधिक वाचनीय (रिडेबल) फॉर्मॅटमध्ये रूपांतरित करते.

## उदाहरण 17

```
var d = new date();
```

```
alert(d.toDateString());
```

रिझल्ट : (पिक्चर 8)



## JavaScript date फॉर्मॅट

JavaScript date इनपुट

JavaScript date इनपुट फॉर्मॅटचे साधारणपणे 3 टाइप असतात.

टाइप	उदाहरण
ISO date	"2002-06-30" (द इंटरनॅशनल स्टॅण्डर्ड)
short date	"06/30/2002"
long date	"जून 30 2002" किंवा "30 जून 2002"

## JavaScript date आउटपुट

इनपुट फॉर्मेटपासून स्वतंत्र, JavaScript पूर्ण टेक्स्ट स्ट्रिंग फॉर्मेटमध्ये तारखा आउटपुट करेल.

## JavaScript ISO date

ISO 8601 हे तारखा आणि वेळेचे प्रतिनिधित्व करण्यासाठी आंतरराष्ट्रीय स्टॅण्डर्ड आहे. ISO 8601 सिंटॅक्स (YYYY-MM-DD) हे देखील पसंतीचे JavaScript date फॉर्मेट आहे.

## उदाहरण 18 (complete date)

```
var d = new date("2002-06-30");
```

टीप: गणना केलेली date तुमच्या टाइम झोनशी संबंधित असेल. तुमच्या टाइम झोननुसार, वरील निकाल 29 जून ते 30 जून दरम्यान बदलतील.

## ISO तारखा (वर्ष आणि महिना)

दिवस (YYYYMM) स्पेसिफाइड केल्याशिवाय ISO तारखा लिहिल्या जाऊ शकतात.

## उदाहरण 19

```
var d = new date("2002-06");
```

## रिझल्ट :

शनि जून 01 2002 05:30:00 GMT+0530 (भारतीय प्रमाण वेळ)

## ISO तारखा (केवळ वर्ष)

ISO तारखा महिना आणि दिवसाशिवाय (YYYY) लिहिल्या जाऊ शकतात.

## उदाहरण 20

```
var d = new date("2011");
```

## रिझल्ट :

Sat Jan 01 2011 05:30:00 GMT+0530 (India Standard Time)

## ISO तारखा (date-वेळ)

ISO तारखा जोडलेले तास, मिनिटे आणि सेकंदांसह लिहिल्या जाऊ शकतात (YYYY-MM DDTHH:MM:SSZ)

## उदाहरण 21

```
var d = new date("2011-12-20T12:00:00Z");
```

## रिझल्ट :

मंगळ 20 डिसेंबर 2011 17:30:00 GMT+0530 (भारतीय प्रमाण वेळ)

टीप: date आणि वेळ कॅपिटल T सह विभक्त केली आहे. UTC वेळ कॅपिटल अक्षर Z सह परिभाषित केली आहे.

## JavaScript short date

लहान तारखा याप्रमाणे "MM/DD/YYYY" वाक्यरचनेने लिहिल्या जातात.

## उदाहरण 22

```
var d = new date("06/30/2002");
```

## रिझल्ट :

रवि जून 30 2002 00:00:00 GMT+0530 (भारतीय प्रमाण वेळ)

टीप: काही ब्राउझरमध्ये, अग्रगण्य शून्य नसलेले महिने किंवा दिवस त्रुटी निर्माण करू शकतात. "YYYY/MM/DD" चे वर्तन अनडिफाइन आहे. काही ब्राउझर फॉर्मेटचा अंदाज लावण्याचा प्रयत्न करतील. काही NaN परत करतील. "DD-MM-YYYY" चे वर्तन देखील अनडिफाइन आहे. काही ब्राउझर फॉर्मेटचा अंदाज लावण्याचा प्रयत्न करतील. काही NaN परत करतील.

## JavaScript long date

लांब तारखा बहुतेकदा अशा "MMM DD YYYY" वाक्यरचनासह लिहिल्या जातात.

## उदाहरण 23

```
var d = new date ("ऑगस्ट 31 2012");
```

महिना आणि दिवस कोणत्याही क्रमाने असू शकतात.

## उदाहरण 24

```
var d = new date("31 ऑगस्ट 2012");
```

आणि, महिना पूर्ण (जानेवारी) किंवा संक्षिप्त (जानेवारी) लिहिला जाऊ शकतो.

## उदाहरण 25

```
var d = new date ("ऑगस्ट 31 2012");
```

```
var d = new date ("ऑगस्ट 31 2012");
```

टीप: स्वल्पविराम दुर्लक्षित केले जातात. नावे केस असेन्सिटिव्ह आहेत.

## date इनपुट - date पार्सिंग

तुमच्याकडे वैध date स्ट्रिंग असल्यास, तुम्ही ती मिलिसेकंदमध्ये रूपांतरित करण्यासाठी Date.parse() मेथड्स वापरू शकता. Date.parse() date आणि जानेवारी 1, 1970 मधील मिलिसेकंदांची नंबर मिळवते.

## उदाहरण 26

```
var msec = Date.parse("sept 15, 1996");
```

```
document.getElementById("demo").innerHTML = msec;
```

## रिझल्ट :

```
842725800000
```

त्यानंतर तुम्ही मिलिसेकंदांची नंबर वापरून ती date ऑब्जेक्टमध्ये रूपांतरित करू शकता.

## उदाहरण 27

```
var msec = Date.parse("sept 15, 1996");
```

```
var d = new date(msec);
```

```
document.getElementById("demo").innerHTML = d;
```

**रिझल्ट :**

Sun Sep 15 1996 00:00:00 GMT+0530 (India Standard Time)

## JavaScript मिळवा date मेथड्स

या पद्धतीचा वापर date ऑब्जेक्टवरून इनफार्मेशन मिळविण्यासाठी केला जाऊ शकतो.

मेथड्स	डिस्क्रिप्शन
getFullYear()	चार अंकी नंबर म्हणून वर्ष मिळवा(yyyy)
getMonth()	नंबर म्हणून महिना मिळवा (0-11)
getDate()	दिवसाची नंबर म्हणून मिळवा (1-31)
getHours()	तास मिळवा (0-23)
getMinutes()	मिनिट मिळवा (0-59)
getSeconds()	दुसरा मिळवा (0-59)
getMilliseconds()	मिलिसेकंद मिळवा (0-999)
getTime()	वेळ मिळवा (1 जानेवारी 1970 पासून मिलिसेकंद)
getDay()	आठवड्याचा दिवस क्रमांक म्हणून मिळवा (0-6)
Date.now()	वेळ मिळेल. ECMAScript 5.

## getTime() मेथड्स

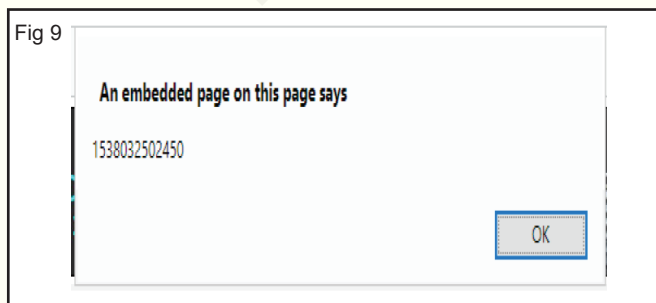
getTime() मेथड्स 1 जानेवारी 1970 पासून मिलिसेकंदांची नंबर मिळवते.

## उदाहरण 28

```
var d = new date();
```

```
alert(d.getTime());
```

**रिझल्ट :** (पिक्चर 9)



getFullYear() मेथड्स

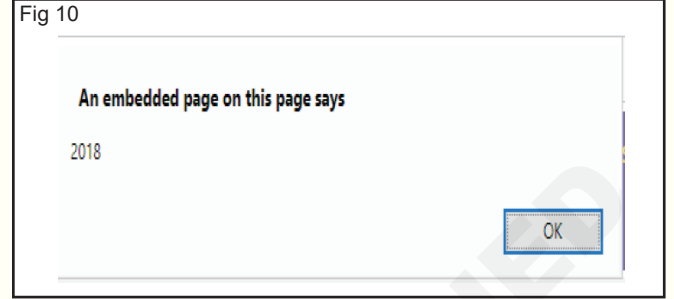
getFullYear() मेथड्स चार अंकी नंबर म्हणून तारखेचे वर्ष मिळवते.

## उदाहरण 29

```
var d = new date();
```

```
alert(d.getFullYear());
```

**रिझल्ट :** (पिक्चर 10)



getMonth() method

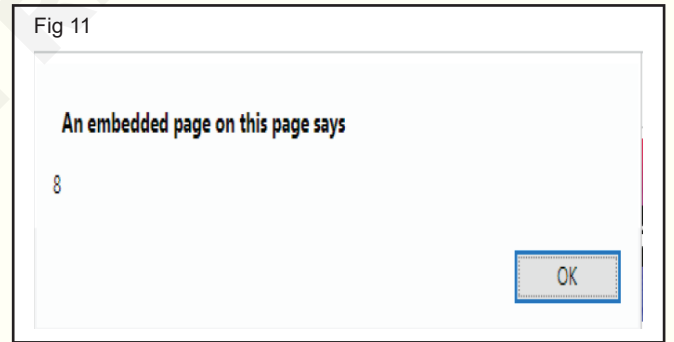
getMonth() मेथड्स एका तारखेचा महिना क्रमांक (0-11) म्हणून परत करते.

## उदाहरण 30

```
var d = new date();
```

```
document.getElementById("demo").innerHTML =  
d.getMonth();
```

**रिझल्ट :** (पिक्चर 11)



JavaScript मध्ये, पहिला महिना (जानेवारी) महिना क्रमांक 0 आहे, त्यामुळे डिसेंबर महिना क्रमांक 11 परत येतो.

तुम्ही नावांचा अॅरे वापरू शकता आणि महिना नाव म्हणून परत करण्यासाठी getMonth() वापरू शकता.

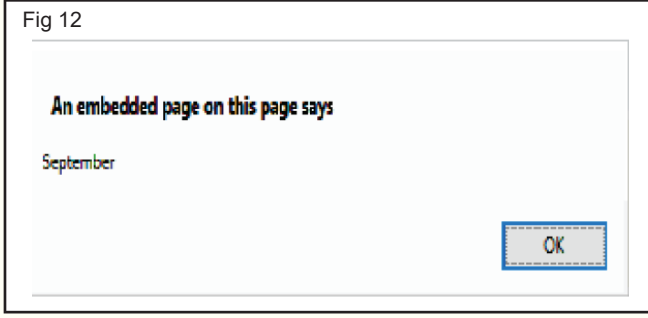
## उदाहरण 31

```
var d = new Date();
```

```
var months = ["January", "February", "March", "April",  
"May", "June", "July", "August", "September", "October",  
"November", "December"];
```

```
document.getElementById("demo").innerHTML =  
months[d.getMonth()];
```

रिझल्ट : (पिक्चर 12)



getDate() मेथड्स

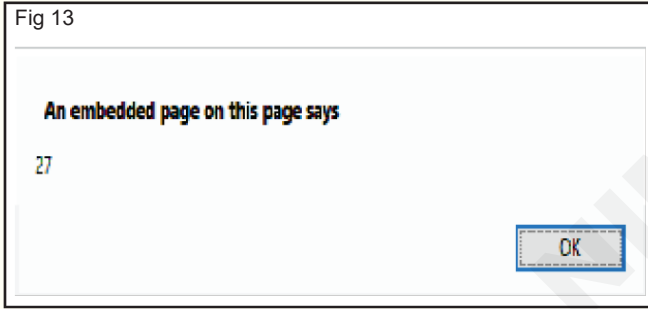
getDate() मेथड्स तारखेचा दिवस क्रमांक (1-31) म्हणून परत करते.

### उदाहरण 32

```
var d = new date();
```

```
alert(d.getDate());
```

रिझल्ट : (पिक्चर 13)



getHours() मेथड्स

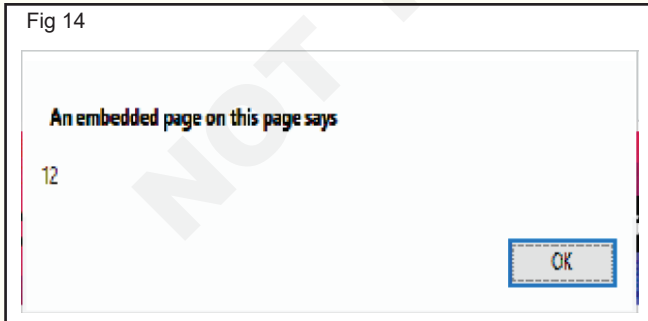
getHours() मेथड्स तारखेचे तास क्रमांक (0-23) म्हणून परत करते.

### उदाहरण 33

```
var d = new date();
```

```
alert(d.getHours());
```

रिझल्ट : (पिक्चर 14)



getMinutes() मेथड्स

getMinutes() मेथड्स तारखेची मिनिटे नंबर (0-59) म्हणून परत करते.

### उदाहरण 34

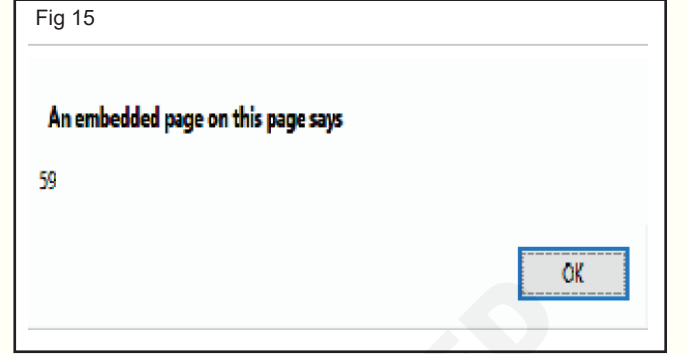
```
var d = new date();
```

```
alert(d.getMinutes());
```

रिझल्ट : (पिक्चर 15)

getSeconds() मेथड्स

getSeconds() मेथड्स क्रमांक (0-59) म्हणून तारखेचे सेकंद परत करते.



### उदाहरण 35

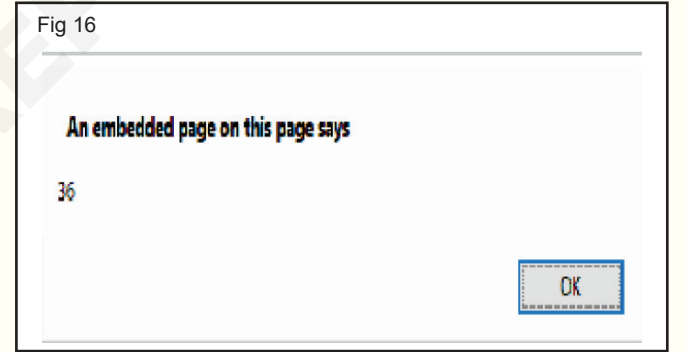
```
var d = new date();
```

```
alert(d.getSeconds());
```

रिझल्ट : (पिक्चर 16)

getMilliseconds() मेथड्स

getMilliseconds() मेथड्स तारखेचे मिलीसेकंद नंबर (0- 999) म्हणून परत करते.

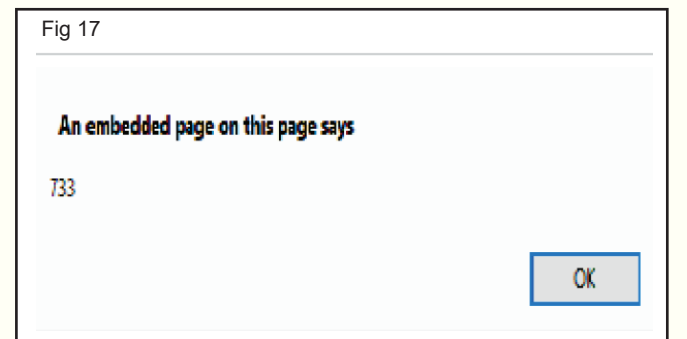


### उदाहरण 36

```
var d = new date();
```

```
alert(d.getMilliseconds());
```

रिझल्ट : (पिक्चर 17)



## getDay() मेथड्स

getDay() मेथड्स एका तारखेचा आठवड्याचा दिवस क्रमांक (0-6) म्हणून परत करते.

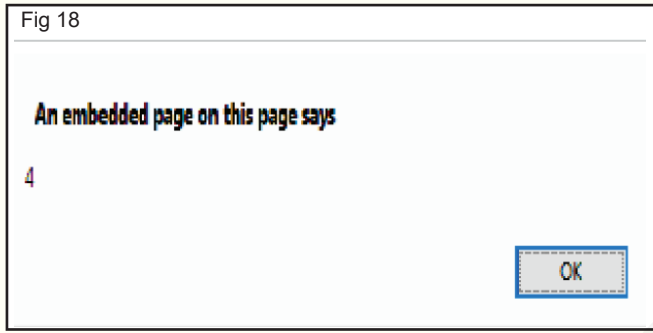
### उदाहरण 37

```
var d = new date();  
alert ( d . getDay ( ) );
```

रिझल्ट :(पिक्चर 18)

JavaScript मध्ये, आठवड्याचा पहिला दिवस (0) म्हणजे "Sun", जरी जगातील काही देश आठवड्याचा पहिला दिवस "सोमवार" मानत असले तरीही

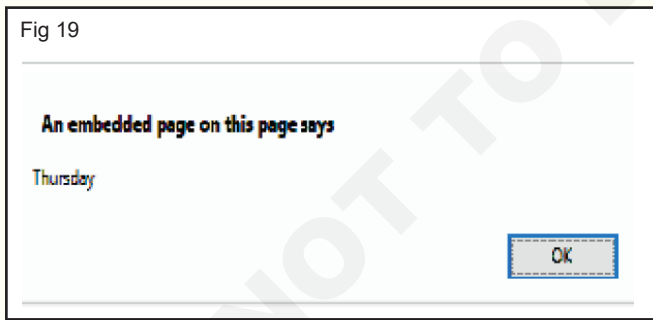
तुम्ही नावांचा अॅरे वापरू शकता आणि आठवड्याचा दिवस नाव म्हणून परत करण्यासाठी getDay() वापरू शकता.



### उदाहरण 38

```
var d = new Date();  
var days = ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"];  
alert(days[d.getDay()]);
```

रिझल्ट :(पिक्चर 19)



## UTC date मेथड्स

UTC तारखांसोबत काम करण्यासाठी UTC date मेथड्स वापरल्या जातात (युनिव्हर्सल टाइम झोन तारखा).

मेथड्स	डिस्क्रिप्शन
getUTCDate()	getDate() प्रमाणेच, परंतु UTC date परत करते
getUTCDday()	getDay() प्रमाणेच, परंतु UTC दिवस परत करतो
getUTCFullYear()	getFullYear() प्रमाणेच, परंतु UTC वर्ष परत करते
getUTCHours()	getHours() प्रमाणेच, परंतु UTC तास परत करतो
getUTCMilliseconds()	getMilliseconds() प्रमाणेच, परंतु UTC मिलिसेकंद परत करते
getUTCMminutes()	getMinutes() प्रमाणेच, परंतु UTC मिनिटे पुन्हा वळते
getUTCMmonth()	getMonth() प्रमाणेच, परंतु UTC महिना परत करतो
getUTCSeconds()	getSeconds() प्रमाणेच, परंतु UTC सेकंद पुन्हा वळते

## JavaScript सेट date मेथड्स

सेट डेट पद्धतीमुळे तुम्हाला डेट ऑब्जेक्टसाठी date व्हॅल्यू (वर्षे, महिने, दिवस, तास, मिनिटे, सेकंद, मिलिसेकंद) सेट करता येतात.

### date मेथड्स सेट करा

तारखेचा पार्ट सेट करण्यासाठी Set Date मेथड्स वापरल्या जातात.

मेथड्स	डिस्क्रिप्शन
setDate()	दिवसाची नंबर म्हणून सेट करा (1-31)
setFullYear()	वर्ष सेट करा (पर्यायी महिना आणि दिवस)
setHours()	तास सेट करा (0-23)
setMilliseconds()	मिलिसेकंद (0-999) सेट करा
setMinutes()	मिनिटे सेट करा (0-59)
setMonth()	महिना सेट करा (0-11)
setSeconds()	सेकंद सेट करा (0-59)
setTime()	वेळ सेट करा (1 जानेवारी 1970 पासून मिलिसेकंद)

### setFullYear() मेथड्स

setFullYear() मेथड्स date ऑब्जेक्टचे वर्ष सेट करते. या उदाहरणात 2020 पर्यंत.

### उदाहरण 39

```
<script>  
var d = new Date();  
d.setFullYear(2020);
```



```
alert(d);
</script>
```

रिझल्ट : (पिक्चर 20)



टीप: `setFullYear()` मेथड्स वैकल्पिकरित्या महिना आणि दिवस सेट करू शकते.

उदाहरण 40

```
<script>
var d = new Date();
d.setFullYear(2018, 10, 2);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 21)



टीप: 0 पासून महिन्याची नंबर . डिसेंबर महिना 11 आहे

**setMonth() मेथड्स**

`setMonth()` मेथड्स `date` ऑब्जेक्टचा महिना सेट करते (0-11).

उदाहरण 41

```
<script>
var d = new Date();
d.setMonth(2);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 22)

**setDate() मेथड्स**

`setDate()` मेथड्स `date` ऑब्जेक्टचा दिवस सेट करते (1-31).

Fig 22



उदाहरण 42

```
<script>
var d = new Date();
d.setDate(18);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 23)

`setDate()` मेथड्स तारखेला दिवस जोडण्यासाठी देखील वापरली जाऊ शकते.

Fig 23



उदाहरण 43

```
<script>
var d = new Date();
d.setDate(d.getDate() + 25);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 24)

दिवस जोडल्यास, महिना किंवा वर्ष बदलल्यास, बदल `date` ऑब्जेक्टद्वारे ऑटोमॅटिक पणे हाताळले जातात.

Fig 24



### setHours() मेथड्स

setHours() मेथड्स date ऑब्जेक्टचे तास सेट करते (0-23).

#### उदाहरण 44

```
<script>
var d = new Date();
d.setHours(20);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 25)

### setMinutes() मेथड्स

setMinutes() मेथड्स date ऑब्जेक्टची मिनिटे सेट करते (0-59)



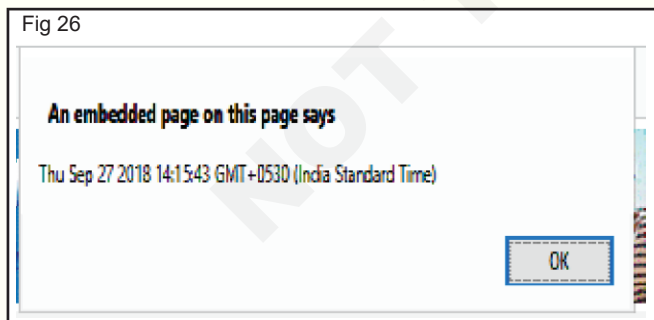
#### उदाहरण 45

```
<script>
var d = new Date();
d.setMinutes(15);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 26)

### setSeconds() मेथड्स

setSeconds() मेथड्स date ऑब्जेक्टचे सेकंद सेट करते (0-59).



### उदाहरण 46

```
<script>
var d = new Date();
d.setSeconds(20);
alert(d);
</script>
```

रिझल्ट : (पिक्चर 27)

### तारखांची कम्प्यारीजन करा

तारखांची सहज कम्प्यारीजन करता येते.

खालील उदाहरण आजच्या तारखेची 14 जानेवारी 2100 शी कम्प्यारीजन करते.



#### उदाहरण 47

```
var date1 = new Date(2010, 00, 15); //Year, Month, Date
var date2 = new Date(2011, 00, 15); //Year, Month, Date
if (date1 > date2)
{
    alert("Date One is greather then Date Two.");
}
else
{
    alert("Date Two is greather then Date One.");
}
```

## IIS वापरून वेब प्रकल्प डिप्लॉय करा (Deploy web project using IIS)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- DOM परिभाषित करा
- DOM मेथड्स स्पष्ट करा
- DOM डॉक्युमेंट्स स्पष्ट करा
- HTML DOM एलिमेंटचे डिस्क्रिप्शन करा
- HTML DOM इव्हेंट स्पष्ट करा
- ओपन सोर्स सॉफ्टवेअर जाणून घ्या

### JavaScript HTML DOM

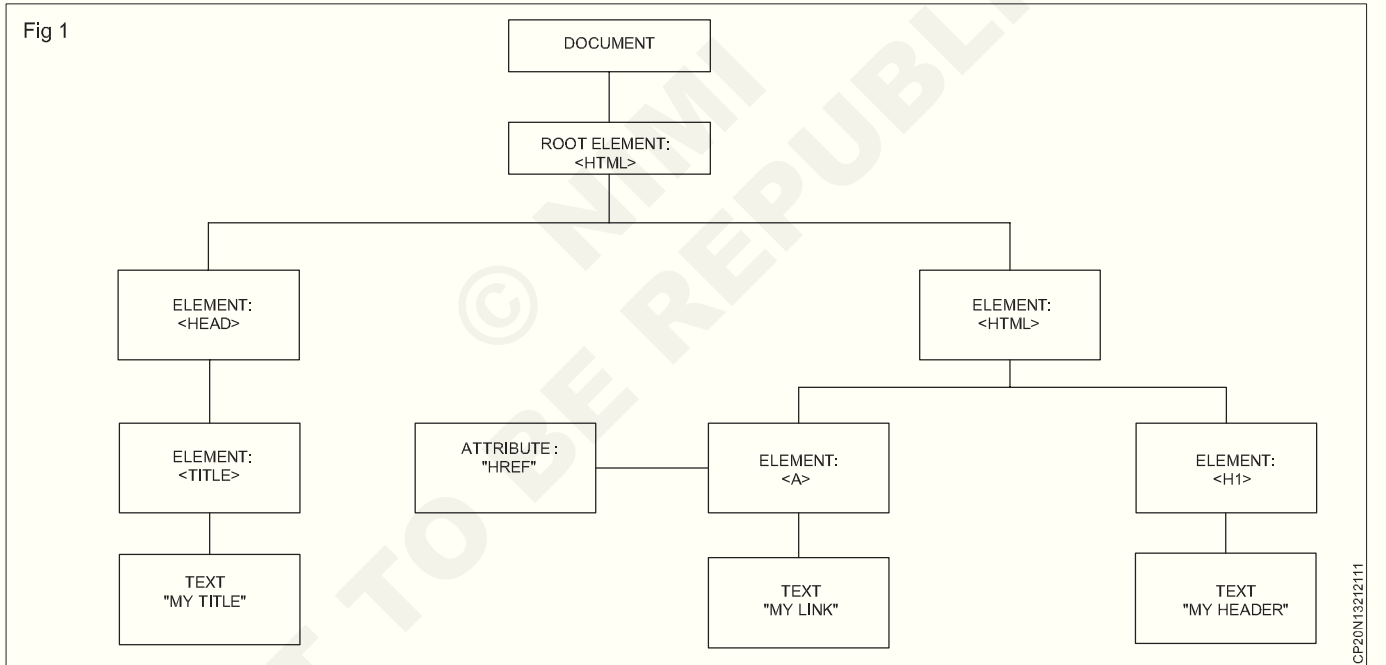
HTML DOM सह, JavaScript HTML डॉक्युमेंट्स च्या सर्व एलिमेंटमध्ये एक्सेस आणि सुधारणा करू शकते.

**HTML DOM (डॉक्युमेंट्स ऑब्जेक्ट मॉडेल):** जेव्हा वेब पृष्ठ लोड केले जाते, तेव्हा ब्राउझर पृष्ठाचे डॉक्युमेंट्स ऑब्जेक्ट मॉडेल तयार करतो.

HTML DOM मॉडेल ऑब्जेक्ट्सच्या झाडाप्रमाणे तयार केले आहे:

ऑब्जेक्ट्सचे HTML DOM ट्री

ऑब्जेक्ट मॉडेलसह, JavaScript ला डायनॅमिक HTML तयार करण्यासाठी आवश्यक असलेली सर्व पॉवर मिळते.



CP20N1321211

- JavaScript पेजमधील सर्व HTML एलिमेंट बदलू शकते.
- JavaScript पेजमधील सर्व HTML स्पेशल ता बदलू शकते
- JavaScript पेजमधील सर्व CSS स्टाइल बदलू शकते
- JavaScript विद्यमान HTML एलिमेंट आणि स्पेशल ता काढून टाकू शकते
- JavaScript नवीन HTML एलिमेंट आणि स्पेशल ता जोडू शकते
- JavaScript पृष्ठावरील सर्व विद्यमान HTML इव्हेंटवर प्रतिक्रिया देऊ शकते

### DOM(डोम)

DOM हे W3C (वर्ल्ड वाइड वेब कंसोर्टियम) स्टॅण्डर्ड आहे.

DOM डॉक्युमेंट्स मध्ये एक्सेस करण्यासाठी स्टॅण्डर्ड परिभाषित करते:

“W3C डॉक्युमेंट ऑब्जेक्ट मॉडेल (DOM) एक प्लॅटफॉर्म आणि भाषा-तटस्थ इंटरफेस आहे जो प्रोग्राम्स आणि स्क्रिप्ट्सना डॉक्युमेंटची कंटेंट, रचना आणि स्टाइल डायनॅमिकरित्या एक्सेस आणि अपडेट करण्यास अनुमती देतो.”

W3C DOM स्टॅण्डर्ड 3 वेगवेगळ्या भागांमध्ये विभक्त केले आहे:

- कोर DOM - सर्व डॉक्युमेंट्स प्रकारांसाठी स्टॅण्डर्ड मॉडेल

- XML DOM - XML डॉक्युमेंट्स साठी स्टॅण्डर्ड मॉडेल
- HTML DOM - HTML डॉक्युमेंट्स साठी स्टॅण्डर्ड मॉडेल

### HTML DOM म्हणजे काय?

HTML DOM हे HTML साठी एक स्टॅण्डर्ड ऑब्जेक्ट मॉडेल आणि प्रोग्रामिंग इंटरफेस आहे. ते परिभाषित करते:

- HTML एलिमेंट ऑब्जेक्ट्स म्हणून
- सर्व HTML एलिमेंटचे प्रॉपर्टीज
- सर्व HTML एलिमेंटमध्ये एक्सेस करण्याच्या मेथड्स
- सर्व HTML एलिमेंटसाठी इव्हेंट

दुसऱ्या शब्दांत: HTML एलिमेंट कसे मिळवायचे, बदलायचे, ऑड करायचे किंवा डिलेट करायचे यासाठी HTML DOM हे एक स्टॅण्डर्ड आहे.

### HTML DOM मेथड्स

HTML DOM मेथड्स म्हणजे तुम्ही करू शकता अशा क्रिया (HTML एलिमेंटवर)

HTML DOM प्रॉपर्टीज ही व्हॅल्यू आहेत (HTML एलिमेंटची) जी तुम्ही सेट करू शकता किंवा बदलू शकता.

### DOM प्रोग्रामिंग इंटरफेस

HTML DOM मध्ये JavaScript (आणि इतर प्रोग्रामिंग भाषांसह) एक्सेस केला जाऊ शकतो.

DOM मध्ये, सर्व HTML एलिमेंट ऑब्जेक्ट्स म्हणून परिभाषित केले जातात. प्रोग्रामिंग इंटरफेस प्रत्येक ऑब्जेक्टचे प्रॉपर्टीज आणि मेथड्स आहे. प्रॉपर्टी हे व्हॅल्यू आहे जे तुम्ही मिळवू शकता किंवा सेट करू शकता (जसे की HTML एलिमेंटची कंटेंट बदलणे). मेथड्स ही एक क्रिया आहे जी तुम्ही करू शकता (जसे की HTML एलिमेंट जोडणे किंवा हटवणे).

### उदाहरण १

खालील उदाहरण id="demo" सह <p> एलिमेंटची कंटेंट (आतील एचटीएमएल) बदलते:

```
<html>
<body>
<p id="demo"></p>
<script>
```

### HTML एलिमेंट सर्चत आहे

मेथड्स	डिस्क्रिप्शन
document.getElementById()	एलिमेंट आयडीनुसार एक एलिमेंट शोधा
document.getElementsByTagName()	टॅग नावाने एलिमेंट शोधा
document.getElementsByClassName()	वर्गाच्या नावानुसार एलिमेंट शोधा

```
document.getElementById("demo").innerHTML = "Welcome to JavaScript!";
```

वरील उदाहरणात, getElementById ही एक मेथड्स आहे,

```
</script>
```

```
</body>
```

```
</html>
```

तर innerHTML ही एक प्रॉपर्टी आहे.

### getElementById मेथड्स

HTML एलिमेंट एक्सेस करण्याचा सर्वात सामान्य पाथ म्हणजे एलिमेंटचा आयडी वापरणे. वरील उदाहरणात getElementById मेथड्स एलिमेंट सर्चण्यासाठी id="demo" वापरली आहे.

### इनरएचटीएमएल प्रॉपर्टी

एलिमेंटची कंटेंट मिळवण्याचा सर्वात सोपा पाथ म्हणजे innerHTML प्रॉपर्टीज वापरणे. इनरएचटीएमएल प्रॉपर्टीज एचटीएमएल एलिमेंटची कंटेंट मिळविण्यासाठी किंवा बदलण्यासाठी सबयुक्त आहे.

**टीप:** इनरएचटीएमएल प्रॉपर्टीचा वापर <html> आणि <body> सह कोणतेही HTML एलिमेंट मिळविण्यासाठी किंवा बदलण्यासाठी केला जाऊ शकतो.

### HTML DOM डॉक्युमेंट्स

#### HTML DOM डॉक्युमेंट्स ऑब्जेक्ट

डॉक्युमेंट्स ऑब्जेक्ट आपल्या वेब पृष्ठावरील इतर सर्व ऑब्जेक्ट्सचा मालक आहे. HTML DOM ऑब्जेक्ट मॉडेलमध्ये, डॉक्युमेंट्स ऑब्जेक्ट आपल्या वेब पृष्ठाचे प्रतिनिधित्व करतो. तुम्हाला एचटीएमएल पेजमधील ऑब्जेक्ट्स एक्सेस करायचे असल्यास, तुम्ही नेहमी डॉक्युमेंट ऑब्जेक्ट एक्सेस करण्यापासून सुरुवात करता.

एचटीएमएलमध्ये एक्सेस करण्यासाठी आणि हाताळण्यासाठी तुम्ही डॉक्युमेंट्स ऑब्जेक्टचा वापर कसा करू शकता याची काही उदाहरणे खाली दिली आहेत.

## HTML एलिमेंट बदलणे

मेथड्स	डिस्क्रिप्शन
element.innerHTML=	एलिमेंटचा इंटर्नल HTML बदला
element.attribute=	HTML एलिमेंटची स्पेशल ता बदला
element.setAttribute((attribute,value)	HTML एलिमेंटची स्पेशल ता बदला
element.style.property=	HTML एलिमेंटची स्टाइल बदला

## HTML एलिमेंट जोडणे आणि हटवणे

मेथड्स	डिस्क्रिप्शन
document.createElement()	HTML एलिमेंट तयार करा
document.removeChild()	HTML एलिमेंट काढा
document.appendChild()	HTML एलिमेंट ऑड करा
document.replaceChild()	HTML एलिमेंट पुनर्स्थित करा
document.write(टेक्स्ट )	HTML आउटपुट प्रवाहात लिहा

## इव्हेंट हँडलर जोडत आहे

मेथड्स	डिस्क्रिप्शन
document.getElementById(id).onclick=function () {code}	ऑनक्लिक इव्हेंटमध्ये इव्हेंट हँडलर कोड जोडत आहे

## JavaScript HTML DOM एलिमेंट

### HTML एलिमेंट सर्चत आहे

बऱ्याचदा, JavaScript सह, तुम्हाला HTML एलिमेंट हाताळायचे आहेत.

हे करण्यासाठी, आपल्याला प्रथम एलिमेंट शोधावे लागतील. हे करण्याचे दोन पाथ आहेत:

- आयडीद्वारे HTML एलिमेंट सर्चणे
- टॅग नावाने HTML एलिमेंट सर्चणे
- वर्गाच्या नावानुसार HTML एलिमेंट सर्चणे
- HTML ऑब्जेक्ट संग्रहाद्वारे HTML एलिमेंट सर्चणे

### आयडीनुसार HTML एलिमेंट सर्चत आहे

DOM मध्ये HTML एलिमेंट सर्चण्याचा सर्वात सोपा पाथ म्हणजे एलिमेंट आयडी वापरणे. हे उदाहरण id="demo" सह एलिमेंट सर्चते:

#### उदाहरण २

```
var x = document.getElementById("डेमो");
```

एलिमेंट आढळल्यास, मेथड्स ऑब्जेक्ट म्हणून एलिमेंट परत करेल (x मध्ये). जर एलिमेंट सापडला नाही, तर x मध्ये शून्य असेल.

### टॅग नावाने HTML एलिमेंट सर्चणे

हे उदाहरण id="main" सह एलिमेंट सर्चते आणि नंतर "main" मधील सर्व <p> एलिमेंट सर्चते:

#### उदाहरण ३

```
var x = document.getElementById("मुख्य");
```

```
var y = x.getElementsByTagName("p");
```

### वर्गाच्या नावानुसार HTML एलिमेंट सर्चणे

तुम्हाला समान वर्गाच्या नावाचे सर्व HTML एलिमेंट शोधायचे असल्यास, ही मेथड्स वापरा ElementsByClassName() मिळवा

#### उदाहरण ४

```
document.getElementsByClassName("intro");
```

वरील उदाहरण class="intro" सह सर्व एलिमेंटची लिस्ट देते.

**टीप:** वर्गाच्या नावाने एलिमेंट सर्चणे इंटरनेट एक्सप्लोरर 5,6,7 आणि 8 मध्ये कार्य करत नाही.

### HTML ऑब्जेक्ट कलेक्शनद्वारे HTML एलिमेंट सर्चणे

हे उदाहरण फॉर्म संकलनामध्ये id="frm1" सह फॉर्म एलिमेंट सर्चते आणि सर्व एलिमेंट व्हॅल्यू प्रदर्शित करते:

मेथड्स	डिस्क्रिप्शन	DOM
document.anchors	नाव स्पेशल तामधील मूल्यासह सर्व <a> परत करते	१
document.applets	सर्व <applet> एलिमेंट परत करते (HTML5 मध्ये नापसंत)	१
document.baseURI	डॉक्युमेंट्स चा परिपूर्ण आधार URI मिळवते	3
document.body	<body> एलिमेंट परत करते	१
document.cookie	डॉक्युमेंट्स ची कुकी परत करते	१
document.doctype	डॉक्युमेंट्स चा doctype परत करते	3
document.documentElement	<html> एलिमेंट परत करतो	3
document.documentMode	ब्राउझरद्वारे वापरलेला मोड परत करतो	3
document.documentURI	डॉक्युमेंट्स चा URI परत करतो	3
document.domain	डॉक्युमेंट्स सर्व्हरचे डोमेन नाव परत करते	१
document.domConfig	DOM कॉन्फिगरेशन परत करते	3
document.embeds	सर्व <embed> एलिमेंट परत करते	3
document.forms	सर्व <form> एलिमेंट परत करते	१
document.head	<head> एलिमेंट परत करते	3
document.images	सर्व <image> एलिमेंट परत करते	१
document.implementation	DOM अंमलबजावणी परत करते	3
document.inputEncoding	डॉक्युमेंट्स चे एन्कोडिंग (कॅरक्टर संच) परत करते	3
document.lastModified	डॉक्युमेंट्स अपडेट केल्याची date आणि वेळ मिळवते	3
document.links	href मधील सर्व <एरिया > आणि <a> एलिमेंटचे व्हॅल्यू मिळवते	१
document.readyState	डॉक्युमेंट्स ची (लोडिंग) स्थिती परत करते	3
document.referrer	रेफररचा URI परत करतो (लिकिंग डॉक्युमेंट्स )	१
document.scripts	सर्व <script> एलिमेंट परत करते	3
document.strictErrorChecking	त्रुटी तपासणे लागू केले असल्यास परत येते	3
document.title	<title> एलिमेंट परत करते	१
document.URL	डॉक्युमेंट्स ची संपूर्ण URL परत करते	१

### उदाहरण 5

```
var x = document.getElementById("frm1");
var text = "";
var i;
for (i = 0; i < x.length; i++) {
    text += x.elements[i].value + "<br>";
}
document.getElementById("demo").innerHTML = text;
```

खालील HTML ऑब्जेक्ट्स (आणि ऑब्जेक्ट संग्रह) देखील एक्सेस योग्य आहेत:

- document.anchors
- document.body

- document.documentElement
- document.embeds
- document.forms
- document.head
- document.images
- document.links
- document.scripts
- document.title

### HTML आउटपुट प्रवाह बदलणे

HTML DOM JavaScript ला HTML एलिमेंटची कंटेंट बदलण्याची परवानगी देतो. HTML आउटपुट प्रवाह बदलणे

JavaScript डायनॅमिक HTML कंटेंट तयार करू शकते. JavaScript मध्ये, document.write() चा वापर थेट HTML आउटपुट प्रवाहावर लिहिण्यासाठी केला जाऊ शकतो.

### उदाहरण 6

```
<!DOCTYPE html>
<html>
<body>
<script>
document.write(Date());
</script>
</body>
</html>
```

**टीप:** डॉक्युमेंट लोड झाल्यानंतर document.write() कधीही वापरू नका. ते डॉक्युमेंट्स ओवरराइडिंग करेल.

### HTML कंटेंट बदलणे

HTML एलिमेंटची कंटेंट सुधारण्याचा सर्वात सोपा पाथ म्हणजे innerHTML प्रॉपर्टीज वापरणे. HTML एलिमेंटची कंटेंट बदलण्यासाठी, हा वाक्यरचना वापरा.

```
document.getElementById(id).innerHTML = new HTML
```

हे उदाहरण <p> एलिमेंटची कंटेंट बदलते:

उदाहरण 7

```
<html>
<body>
<p id="p1">Hello World!</p>
<script>
document.getElementById("p1").innerHTML = "New text!";
</script>
</body>
</html>
```

### उदाहरण स्पष्ट केले:

- वरील HTML डॉक्युमेंट्स त id="p1" सह <p> एलिमेंट आहे
- id="p1" सह एलिमेंट मिळविण्यासाठी आम्ही HTML DOM वापरतो
- JavaScript त्या एलिमेंटची कंटेंट (innerHTML) बदलते "नवीन टेक्स्ट !" हे उदाहरण <h1> एलिमेंटची कंटेंट बदलते:

### उदाहरण 8

```
<!DOCTYPE html>
<html>
<body>
<h1 id="id1">Old Heading</h1>
<script>
var element = document.getElementById("id1");
element.innerHTML = "New Heading";
</script>
</body>
</html>
```

### उदाहरण स्पष्ट केले:

- वरील HTML डॉक्युमेंट्स त id="id1" सह <h1> एलिमेंट आहे
- id="id1" सह एलिमेंट मिळविण्यासाठी आम्ही HTML DOM वापरतो
- JavaScript त्या एलिमेंटची कंटेंट (innerHTML) "नवीन टाईटल " मध्ये बदलते गुणधर्मचे व्हॅल्यू बदलणे

### HTML स्पेशल ताचे व्हॅल्यू बदलण्यासाठी, हा वाक्यरचना वापरा.

```
document.getElementById(id).attribute = new व्हॅल्यू
```

हे उदाहरण <img> एलिमेंटच्या src स्पेशल ताचे व्हॅल्यू बदलते.

### उदाहरण ९

```
<!DOCTYPE html>
<html>
<body>

<script>
document.getElementById("Image1").src = "newflower.jpg";
</script>
</body>
</html>
```

### उदाहरण स्पष्ट केले:

- वरील HTML डॉक्युमेंट्स त id="myImage" सह <img> एलिमेंट आहे
- आम्ही id="myImage" सह एलिमेंट मिळविण्यासाठी HTML DOM वापरतो

- JavaScript त्या एलिमेंटची src स्पेशल ता "smiley.gif" वरून "landscape.jpg" मध्ये बदलते

### HTML स्टाइल बदलणे

HTML एलिमेंटची स्टाइल बदलण्यासाठी, हा वाक्यरचना वापरा.

document.getElementById(id).style.property = नवीन स्टाइल  
खालील उदाहरण <p> एलिमेंटची स्टाइल बदलते:

#### उदाहरण 10

```
<html>
<body>
<p id="p2">Hello World!</p>
<script>
document.getElementById("p2").style.color = "green";
</script>
<p>The paragraph above was changed by a script.</p>
</body>
</html>
```

**इव्हेंट वापरणे:** जेव्हा एखादी इवेंट्स घडते तेव्हा HTML DOM तुम्हाला कोड कार्यान्वित करण्याची परवानगी देतो. जेव्हा HTML एलिमेंटवर "गोष्टी घडतात" तेव्हा ब्राउझरद्वारे इव्हेंट तयार केले जातात.

- एका एलिमेंटवर क्लिक केले जाते
- पृष्ठ लोड झाले आहे
- इनपुट फील्ड बदलले आहेत

जेव्हा युजर बटणावर क्लिक करतो तेव्हा हे उदाहरण id="id1" सह HTML एलिमेंटची स्टाइल बदलते.

#### उदाहरण 11

```
<!DOCTYPE html>
<html>
<body>
<h1 id="id1">Heading1</h1>
<button type="button"
onclick="document.getElementById('id1').style.color= 'blue'">
Click Me</button>
</body>
</html>
```

### JavaScript HTML DOM इव्हेंट

एचटीएमएल डीओएम JavaScript ला एचटीएमएल इव्हेंटवर प्रतिक्रिया देण्यास अनुमती देते.

घटनांवर प्रतिक्रिया

एखादी इवेंट्स घडते तेव्हा JavaScript कार्यान्वित केले जाऊ शकते, जसे की युजर HTML एलिमेंटवर क्लिक करतो. जेव्हा युजर एलिमेंटवर क्लिक करतो तेव्हा कोड कार्यान्वित करण्यासाठी, HTML इव्हेंट स्पेशल तामध्ये JavaScript कोड ऑड करा

onclick=जावास्क्रिप्ट

#### HTML इव्हेंटची उदाहरणे:

- जेव्हा युजर माउस क्लिक करतो
- जेव्हा वेब पृष्ठ लोड होते
- प्रतिमा लोड केल्यावर
- जेव्हा माउस एखाद्या एलिमेंटवर फिरतो
- जेव्हा इनपुट फील्ड बदलले जाते
- जेव्हा HTML फॉर्म सबमिट केला जातो
- जेव्हा युजर की स्ट्रोक करतो

या उदाहरणात, जेव्हा युजर त्यावर क्लिक करतो तेव्हा <h1> एलिमेंटची कंटेंट बदलली जाते.

#### उदाहरण 12

```
<!DOCTYPE html>
<html>
<body>
<h1 onclick="this.innerHTML = 'Oops!'">Click on this text!</h1>
</body>
</html>
```

#### HTML इव्हेंट ऐट्रिब्यूट्स

HTML एलिमेंटना इव्हेंट नियुक्त करण्यासाठी तुम्ही इव्हेंट स्पेशल ता वापरू शकता.

#### उदाहरण 13

बटण एलिमेंटला ऑनक्लिक इव्हेंट नियुक्त करा:

```
<button onclick="displayDate()">Try it</button>
```

वरील उदाहरणात, बटणावर क्लिक केल्यावर displayDate नावाचे फंक्शन कार्यान्वित केले जाईल.



## HTML DOM वापरून इव्हेंट नियुक्त करा

HTML DOM तुम्हाला JavaScript वापरून HTML एलिमेंटना इव्हेंट नियुक्त करण्याची परवानगी देतो.

### उदाहरण 14

बटण एलिमेंटला ऑनक्लिक इव्हेंट नियुक्त करा:

Assign an onclick event to a button element:

```
<script>
document.getElementById("myBtn").onclick = displayDate;
</script>
```

वरील उदाहरणामध्ये, displayDate नावाचे फंक्शन id="myBtn" सह HTML एलिमेंटला नियुक्त केले आहे.

बटण क्लिक केल्यावर फंक्शन कार्यान्वित होईल.

### ऑनलोड आणि अनलोड इव्हेंट

जेव्हा युजर पेजवर एक्सेस करतो किंवा सोडतो तेव्हा ऑनलोड आणि अनलोड इव्हेंट ट्रिगर केले जातात. ऑनलोड इव्हेंटचा वापर अभ्यागताचा ब्राउझर टाइप आणि ब्राउझर आवृत्ती तपासण्यासाठी आणि इनफार्मेशन वर आधारित वेब पेजची योग्य आवृत्ती लोड करण्यासाठी केला जाऊ शकतो. ऑनलोड आणि ऑनलोड इव्हेंटचा वापर कुकीज हाताळण्यासाठी केला जाऊ शकतो.

### उदाहरण 15

```
<body onload="checkCookies()">
```

बदललेला प्रोग्रॅम

ऑनचेंज इव्हेंट सहसा इनपुट फील्डच्या प्रमाणीकरणासह वापरला जातो. खाली ऑनचेंज कसे वापरायचे याचे उदाहरण दिले आहे. जेव्हा युजर इनपुट फील्डची कंटेंट बदलतो तेव्हा अपरकेस() फंक्शन कॉल केले जाईल.

### उदाहरण 16

```
<input type="text" id="fname" onchange="upperCase()">
```

### onmouseover आणि onmouseout इव्हेंट

ऑनमाऊसओव्हर आणि ऑनमाऊसआउट इव्हेंट्सचा वापर फंक्शन ट्रिगर करण्यासाठी केला जाऊ शकतो जेव्हा युजर HTML एलिमेंटवर किंवा बाहेर माऊस करतो.

### नमाउसडाउन, ऑनमाउसअप आणि ऑनक्लिक इव्हेंट्स

ऑनमाउसडाउन, ऑनमाउसअप आणि ऑनक्लिक इव्हेंट हे सर्व माउस-क्लिकचे पार्ट आहेत. प्रथम जेव्हा माउस-बटण क्लिक केले जाते, तेव्हा ऑनमाउसडाउन इव्हेंट ट्रिगर केला जातो, त्यानंतर, जेव्हा माउस-बटण सोडले जाते, तेव्हा ऑनमाउसअप इव्हेंट ट्रिगर केला जातो, शेवटी, जेव्हा माउस-क्लिक पूर्ण होते, तेव्हा ऑनक्लिक इव्हेंट ट्रिगर केला जातो.

## onmousedown and onmouseup ऑनक्लिक इव्हेंट

जेव्हा युजर माउस बटण दाबून ठेवतो तेव्हा प्रतिमा बदला.

### ऑनलोड

जेव्हा पृष्ठ लोड करणे समाप्त होईल तेव्हा एक इशारा बॉक्स प्रदर्शित करा.

onfocus

इनपुट फील्डवर फोकस आल्यावर त्याचा बॅकग्राउंड -रंग बदला.

माऊस इव्हेंट

जेव्हा कर्सर त्यावर हलतो तेव्हा एलिमेंटचा रंग बदला.

DOM इव्हेंट लिसनर

addEventListener() मेथड्स

एक इव्हेंट श्रोता ऑड करा जो युजर बटण क्लिक करतो तेव्हा ऍक्टिव्ह होतो.

उदाहरण 17

```
<!DOCTYPE html>
<html>
<body>
<h2>JavaScript addEventListener()/h2>
<p>This example uses the addEventListener() method to
attach a click event to a button.</p>
<button id="myBtn">Try it</button>
<p id="demo"></p>
<script>
document.getElementById("myBtn").addEventListener("click",
displayDate);
function displayDate() {
document.getElementById("demo").innerHTML = Date();
}
</script>
</body>
</html>
```

### रिझल्ट :(आकृती क्रं 1)

- addEventListener() मेथड्स इव्हेंट हँडलरला स्पेसिफाइड एलिमेंटशी संलग्न करते.
- addEventListener() मेथड्स विद्यमान इव्हेंट हँडलर ओव्हरराईट न करता एलिमेंटस इव्हेंट हँडलर संलग्न करते.

Fig 1

## JavaScript addEventListener()

This example uses the addEventListener() method to attach a click event to a button.

Try it

Thu Sep 27 2018 17:10:55 GMT+0530 (India Standard Time)

- तुम्ही एका एलिमेंटमध्ये अनेक इव्हेंट हँडलर जोडू शकता.
- तुम्ही एकाच टाइप चे अनेक इव्हेंट हँडलर एका एलिमेंटमध्ये जोडू शकता, म्हणजे दोन "क्लिक" इव्हेंट.
- तुम्ही केवळ HTML एलिमेंट च नाही तर कोणत्याही DOM ऑब्जेक्टमध्ये इव्हेंट श्रोते जोडू शकता. म्हणजे विंडो ऑब्जेक्ट.
- addEventListener() मेथड्स इव्हेंट बबलिंगवर कशी प्रतिक्रिया देते हे कंट्रोल करणे सोपे करते.
- addEventListener() मेथड्स वापरताना, JavaScript चांगल्या वाचनीय (रिडेबल)तेसाठी HTML मार्कअपपासून वेगळे केले जाते आणि तुम्ही HTML मार्कअप कंट्रोल करत नसतानाही तुम्हाला इव्हेंट श्रोते जोडण्याची परवानगी देते.
- तुम्ही removeEventListener() मेथड्स वापरून इव्हेंट श्रोता सहज काढू शकता.

### ओपन सोर्स सॉफ्टवेअर

तुम्ही ही लोकप्रिय मोफत आणि ओपन सोर्स वेबसाइट बिल्लिंग टूल्स वापरून वेबसाइट तयार करू शकता. आजकाल, तुम्ही वैयक्तिक

उद्योजक असाल किंवा एखाद्या व्यावसायिक संस्थेचे प्रतिनिधीत्व करत असाल, वैयक्तिक आणि व्यावसायिक वाढीसाठी वेबसाइट आवश्यक आहे. आकर्षक वेबसाइट्स तयार करण्यासाठी संस्था खूप पैसा खर्च करत आहेत. खालील काही ओपन सोर्स वेबसाइट बिल्लिंग टूल्स आहेत जी तुम्ही प्रोग्रामिंग किंवा इंटरनेटबद्दल जास्त इनफार्मेशन नसताना तुमची वेबसाइट स्वतः तयार करण्यासाठी वापरू शकता.

### 1 वर्डप्रेस

वर्डप्रेससाठी अधिकृत वेबसाइट <https://wordpress.com> आणि <https://wordpress.org/> आहेत.

### 2 कंपोजर

Kompozer साठी अधिकृत वेबसाइट <https://www.kompozer.net> आहे.

### 3 जूमला

Joomla साठी अधिकृत वेबसाइट <https://www.joomla.org/> आहे.

### 4 Drupal

Drupal साठी अधिकृत वेबसाइट <https://www.drupal.org/> आहे.

### 5 OpenCms

OpenCms साठी अधिकृत वेबसाइट <http://www.opencms.org/en/> आहे.

## एडवान्स फॉर्मूला आणि मॅक्रो तयार करा (Create advanced formulas and macros)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- रेफरेंस टाइप
- कस्टम क्रमांक फॉर्मॅट

### एडवान्स एक्सेल

#### रेफरेंस चे टाइप

सेल संदर्भाचे दोन टाइप आहेत: रिलेटिव्ह आणि अब्सोल्युट. इतर सेलमध्ये कॉपी आणि भरल्यावर रिलेटिव्ह आणि परिपूर्ण रेफरेंस वेगळ्या मेथड्स ने वागतात. जेव्हा फॉर्मूला दुसऱ्या सेलमध्ये कॉपी केले जाते तेव्हा संबंधित रेफरेंस बदलतात. दुसरीकडे, अब्सोल्युट रेफरेंस, ते कोठेही कॉपी केले तरीही स्थिर राहतात.

#### रिलेटिव्ह रेफरेंस

डीफॉल्टनुसार, सर्व सेल रेफरेंस रिलेटिव्ह रेफरेंस असतात. मल्टिपल सेलमध्ये कॉपी केल्यावर, ते रो आणि स्तंभांच्या रिलेटिव्ह स्थितीवर आधारित बदलतात. उदाहरणार्थ, जर तुम्ही फॉर्मूला =A1+B1 रो 1 वरून रो 2 मध्ये कॉपी केले, तर फॉर्मूला =A2+B2 होईल. जेव्हा तुम्हाला अनेक रो किंवा स्तंभांमध्ये समान गणनेची पुनरावृत्ती करण्याची आवश्यकता असते तेव्हा संबंधित रेफरेंस स्पेशल तः सोयीस्कर असतात.

#### अब्सोल्युट रेफरेंस

असे काही वेळा असू शकतात जेव्हा आपण सेल भरताना सेल रेफरेंस बदलू इच्छित नाही. रिलेटिव्ह संदर्भांच्या विपरीत, कॉपी किंवा भरल्यावर परिपूर्ण रेफरेंस बदलत नाहीत. रो आणि/किंवा कॉलम स्थिर ठेवण्यासाठी तुम्ही परिपूर्ण रेफरेंस वापरू शकता.

कॉलम आणि रो च्या आधी डॉलर चिन्ह (\$) जोडून एक परिपूर्ण रेफरेंस फॉर्मूला मध्ये नियुक्त केला जातो. जर ते कॉलम किंवा रो च्या आधी असेल (परंतु दोन्ही नाही), तर त्याला मिश्र रेफरेंस म्हणून ओळखले जाते.

\$A\$2	कॉपी केल्यावर कॉलम आणि रो बदलत नाहीत
A\$2	कॉपी केल्यावर रो बदलत नाही
\$A2	कॉपी केल्यावर कॉलम बदलत नाही

तुम्ही बहुतेक सूत्रांमध्ये रिलेटिव्ह (A2) आणि परिपूर्ण (\$A\$2) फॉर्मॅट वापरात. मिश्र रेफरेंस कमी वारंवार वापरले जातात.

Microsoft Excel मध्ये फॉर्म्युला लिहिताना, खालील व्हिडिओमध्ये दाखवल्याप्रमाणे, रिलेटिव्ह, अब्सोल्युट आणि मिश्रित सेल संदर्भांमध्ये स्विक करण्यासाठी तुम्ही तुमच्या कीबोर्डवरील F4 की दाबू शकता. अचूक रेफरेंस पटकन टाकण्याचा हा एक सोपा पाथ आहे.

#### एक्सेल लिंकेज कस्टम फॉर्मॅट आणि एक्सेल प्रोटेक्शन

जर बिल्ट-इन नंबर फॉर्मॅट तुमच्या गरजा पूर्ण करत नसेल, तर तुम्ही अस्तित्वात असलेल्या नंबर फॉर्मॅटवर आधारित नवीन नंबर फॉर्मॅट तयार

करू शकता आणि कस्टम नंबर फॉर्मॅटच्या लिस्ट मध्ये जोडू शकता. उदाहरणार्थ, तुम्ही ग्राहक इनफार्मेशन असलेली स्प्रेडशीट तयार करत असल्यास, तुम्ही टेलिफोन नंबरसाठी नंबर फॉर्मॅट तयार करू शकता. टेलिफोन नंबर म्हणून फॉर्मॅट करण्यासाठी तुम्ही सेलमधील नंबरच्या स्ट्रिंगवर कस्टम नंबर फॉर्मॅट लागू करू शकता.

**महत्त्वाचे:** कस्टम क्रमांक फॉर्मॅट केवळ नंबर प्रदर्शित करण्याच्या मेथड्स वर रिझल्ट करतात आणि संख्येच्या बेसिक मूल्यावर रिझल्ट करत नाहीत. कस्टम क्रमांक फॉर्मॅट ऍक्टिव्ह वर्कबुकमध्ये स्टोर केले जातात आणि तुम्ही उघडलेल्या नवीन वर्कबुकसाठी सबलब्ध नाहीत.

#### कस्टम नंबर फॉर्मॅट तयार करा

- 1 मुख्यपृष्ठ टॅबवर, नंबर गटामध्ये, नंबर फॉर्मॅट लिस्ट च्या तळाशी असलेल्या अधिक नंबर स्वरूपांवर क्लिक करा.
- 2 फॉर्मॅट सेल डायलॉग बॉक्समध्ये, रेन्ज इंटरनल, कस्टम क्लिक करा.
- 3 टाइप लिस्ट मध्ये, तुम्ही तयार करू इच्छित असलेल्या बिल्ट-इन फॉर्मॅटची निवड करा. उदाहरणार्थ, 0.00.

तुम्ही निवडलेले नंबर फॉर्मॅट टाइप बॉक्समध्ये दिसते.

- 4 टाईप बॉक्समध्ये, तुम्हाला हवे असलेले अचूक फॉर्मॅट तयार करण्यासाठी नंबर फॉर्मॅट कोडमध्ये बदल करा. उदाहरणार्थ, 000-000-0000.

तुमचे बदल अंगभूत फॉर्मॅट बदलणार नाहीत. त्याऐवजी, तुमचे बदल नवीन कस्टम क्रमांक फॉर्मॅट तयार करतात.

- 5 तुम्ही पूर्ण केल्यावर, ओके क्लिक करा.

#### कस्टम नंबर फॉर्मॅट लागू करा

- 1 सेल किंवा सेलची रेन्ज निवडा जी तुम्ही फॉर्मॅट करू इच्छिता.
- 2 मुख्यपृष्ठ टॅबवर, नंबर गटामध्ये, नंबर फॉर्मॅट लिस्ट च्या तळाशी असलेल्या अधिक क्रमांक फॉर्मॅट क्लिक करा.
- 3 फॉर्मॅट सेल डायलॉग बॉक्समध्ये, रेन्ज इंटरनल, कस्टम क्लिक करा.
- 4 टाइप लिस्ट च्या तळाशी, तुम्ही नुकतेच तयार केलेले अंगभूत फॉर्मॅट निवडा. उदाहरणार्थ, 000-000-0000.

तुम्ही निवडलेले नंबर फॉर्मॅट टाइप बॉक्समध्ये दिसते.

- 5 ओके क्लिक करा.

## कस्टम नंबर फॉरमॅट डिलेट करा

- 1 मुख्यपृष्ठ टॅबवर, नंबर गटामध्ये, नंबर फॉरमॅट लिस्ट च्या तळाशी असलेल्या अधिक नंबर स्वरूपांवर क्लिक करा.
- 2 फॉरमॅट सेल डायलॉग बॉक्समध्ये, रेन्ज इंटरनल, कस्टम क्लिक करा.
- 3 टाइप लिस्ट मध्ये, कस्टम क्रमांक फॉरमॅट निवडा, आणि नंतर डिलेट करा क्लिक करा.

### नोट्स

- अंगभूत क्रमांकाचे फॉरमॅट हटवले जाऊ शकत नाही.
- हटवलेल्या कस्टम स्वरूपासह फॉरमॅटिंग केलेल्या कार्यपुस्तिकेतील कोणतेही सेल डीफॉल्ट सामान्य स्वरूपात प्रदर्शित केले जातील.

## एक्सेल प्रोटेक्शन

तुमच्या Excel फायलीमधील डेटामध्ये इतरांना एक्सेस करण्यापासून रोखण्यासाठी, तुमच्या Excel फाइलला पासवर्डने संरक्षित करा.

**टीप:** हा विषय केवळ फाइल-स्तरीय संरक्षणाचा समावेश करतो, आणि वर्कबुक किंवा वर्कशीट प्रोटेक्शन नाही. तुमची एक्सेल फाईल, वर्कबुक किंवा वर्कशीट संरक्षित करण्यामधील फरक जाणून घेण्यासाठी, एक्सेलमधील प्रोटेक्शन आणि सुरक्षा पहा.

- 1 फाइल > इनफार्मेशन निवडा.
- 2 प्रोटेक्ट वर्कबुक बॉक्स निवडा आणि पासवर्डसह एनक्रिप्ट निवडा. 3 पासवर्ड बॉक्समध्ये पासवर्ड प्रविष्ट करा, आणि नंतर ओके निवडा.
- 4 पासवर्ड पुन्हा प्रविष्ट करा बॉक्समध्ये पासवर्डची कन्फर्म करा, आणि नंतर ओके निवडा.

## चेतावणी

- मायक्रोसॉफ्ट विसरलेले पासवर्ड परत मिळवू शकत नाही, त्यामुळे तुमचा पासवर्ड स्पेशल तः संस्मरणीय असल्याची खात्री करा.
- तुम्ही लांबी, कॅरक्टर किंवा नंबर बाबत वापरत असलेल्या पासवर्डवर कोणतेही निर्बंध नाहीत, परंतु पासवर्ड केस-संवेदी असतात.
- क्रेडिट कार्ड क्रमांकासारखी सेन्सिटिव्ह इनफार्मेशन असलेल्या पासवर्ड-संरक्षित फाइल्सचे वितरण करणे नेहमीच सुरक्षित नसते.
- इतर वापरकर्त्यांसोबत फाइल्स किंवा पासवर्ड शेअर करताना सावधगिरी बाळगा. तुम्ही अजूनही पासवर्ड अनपेक्षित वापरकर्त्यांच्या हातात पडण्याचा धोका चालवत आहात. ते लक्षात ठेवा पासवर्डसह फाइल लॉक केल्याने तुमची फाईल दुर्भावनापूर्ण हेतूपासून संरक्षित होत नाही.

## एक्सेल टिप्स

- 1 डेटा ओळखण्यासाठी आणि समजण्यासाठी पिक्चोट टेबल वापरा.
- 2 एकापेक्षा जास्त रो किंवा कॉलम ऑड करा.

3 तुमचा डेटा सुलभ करण्यासाठी फिल्टर वापरा.

4 डुप्लिकेट डेटा पॉइंट किंवा सेट काढा.

5 रो स्तंभांमध्ये स्थानांतरित करा.

6 स्तंभांमध्ये टेक्स्ट इनफार्मेशन स्प्लिट करा.

7 साध्या गणनेसाठी ही सूत्रे वापरा.

8 तुमच्या सेलमधील नंबर ची सरासरी मिळवा.

9 डेटावर आधारित सेल ऑटोमॅटिक रंग बदलण्यासाठी कंडीशनल फॉरमॅटिंग वापरा. 10 स्पेसिफिक Excel फंक्शन ऑटोमॅटिक करण्यासाठी IF Excel फॉर्मूला वापरा.

11 एका सेलचे फॉर्म्युला कुठेही फिरले तरी ते समान ठेवण्यासाठी डॉलर सिम्बॉल्स वापरा.

12 शीटच्या एका भागातून दुसऱ्या भागात डेटा खेचण्यासाठी VLOOKUP फंक्शन वापरा. 13 क्षैतिज स्तंभांमधून डेटा काढण्यासाठी INDEX आणि MATCH सूत्रे वापरा.

14 सेलच्या कोणत्याही रेन्ज तील शब्द किंवा नंबर Excel मध्ये मोजण्यासाठी COUNTIF फंक्शन वापरा.

15 अँपरसँड वापरून सेल एकत्र करा.

16 चेकबॉक्स ऑड करा.

17 वेबसाइटला सेल हायपरलिंक करा.

18 ड्रॉप-डाउन मेनू ऑड करा.

19 फॉर्मेट पेंटर वापरा.

## पिक्चोट टेबल

सारांश डेटा सारांशित करण्यासाठी, विश्लेषण करण्यासाठी, एक्सप्लोर करण्यासाठी आणि सादर करण्यासाठी तुम्ही PivotTable वापरू शकता. PivotCharts PivotTable मधील सारांश डेटामध्ये व्हिज्युअलायझेशन जोडून PivotTables पूरक करतात आणि तुम्हाला कम्प्यारीजन, नमुने आणि ट्रेंड सहजपणे पाहण्याची परवानगी देतात.

PivotTables आणि PivotCharts दोन्ही तुम्हाला तुमच्या एंटरप्राइझमधील गंभीर डेटाबद्दल इनफार्मेशन पूर्ण निर्णय घेण्यास सक्षम करतात. PivotTables तयार करण्यासाठी तुम्ही SQL Server टेबल्स, SQL Server Analysis Services cubes, Azure Marketplace, Office Data Connection (.odc) फाइल्स, XML फाइल्स, Access डेटाबेस आणि टेक्स्ट फाइल्स सारख्या एक्सटर्नल डेटा स्रोतांशी देखील कनेक्ट करू शकता किंवा विद्यमान PivotTables वापरू शकता नवीन टेबल तयार करा.

PivotTable हा मोठ्या प्रमाणात डेटा क्लिकली सारांशित करण्याचा एक संवादी पाथ आहे. अंकीय डेटाचे तपशीलवार विश्लेषण करण्यासाठी आणि तुमच्या डेटाबद्दल अनपेक्षित प्रश्नांची उत्तरे देण्यासाठी तुम्ही PivotTable वापरू शकता. पिक्चोटटेबल स्पेशल तः यासाठी डिझाइन केलेले आहे.

- अनेक युजर -अनुकूल मार्गांनी मोठ्या प्रमाणात डेटाची क्वेरी करणे.

- नंबर तक डेटा सब-टोलिंग आणि एकत्रित करणे, रेन्ज आणि सबरेन्ज नुसार डेटा सारांशित करणे आणि कस्टम गणना आणि सूत्रे तयार करणे.
- तुमचे रिझल्ट फोकस करण्यासाठी डेटाचे स्तर वाढवणे आणि कॉम्प्रेस करणे आणि तुमच्या आवडीच्या क्षेत्रांसाठी सारांश डेटामधून तपशील सर्च घे.
- सोर्स डेटाचे भिन्न सारांश पाहण्यासाठी रो स्तंभांमध्ये किंवा स्तंभाना रो मध्ये हलवणे (किंवा "पिव्होटिंग")
- डेटाचा सर्वात सबयुक्त आणि मनोरंजक सबसंच फिल्टर करणे, क्रमवारी (सॉर्टिंग) लावणे, ग्रुप बद्ध करणे आणि कंडीशनल फॉर्मॅटिंग करणे तुम्हाला फक्त तुम्हाला हव्या असलेल्या इनफार्मेशन वर लक्ष केंद्रित करण्यास सक्षम करते.
- संक्षिप्त, आकर्षक आणि भाष्य केलेले ऑनलाइन किंवा प्रिंटेड रिपोर्ट सादर करणे.

उदाहरणार्थ, येथे डावीकडे घरगुती खर्चाची एक सोपी यादी आहे आणि उजवीकडे असलेल्या लिस्ट वर आधारित पिव्होट टेबल आहे: पिव्होट टेबल सह कार्य करण्याचे पाथ

तुम्ही PivotTable तयार केल्यानंतर त्याचा डेटा सोर्स निवडून, PivotTable फील्ड लिस्ट मधील फील्डची व्यवस्था करून आणि स्टार्टिंग सिन्टेक्स निवडून, तुम्ही PivotTable सोबत काम करत असताना खालील फंक्शन करू शकता:

### खालील गोष्टी करून डेटा एक्सप्लोर करा

- डेटा विस्तृत आणि कॉम्प्रेस करा आणि व्हॅल्यू शी संबंधित बेसिक तपशील दर्शवा.
- वर्गीकरण, फिल्टर आणि ग्रुप फील्ड आणि आयटम.
- सारांश फंक्शन बदला आणि कस्टम गणना आणि सूत्रे ऑड करा

### खालील गोष्टी करून फॉर्म लेआउट आणि फील्ड व्यवस्था बदला

- PivotTable फॉर्म बदला: संक्षिप्त, एक्सटर्नल रेखा किंवा टेबल दार.
- फील्ड ऑड करा, पुनर्रचना करा आणि काढा.
- फील्ड किंवा आयटमचा क्रम बदला.

### खालील गोष्टी करून कॉलम , रो आणि सबटोटल यांचा लेआउट बदला

- कॉलम आणि रो फील्ड हेडर चालू किंवा बंद करा किंवा रिक्त ओळी प्रदर्शित करा किंवा लपवा.
- त्यांच्या रो च्या वर किंवा खाली सबटोटल प्रदर्शित करा.
- रिफ्रेश करताना कॉलम ची रुंदी अडजस्ट करा.
- एक कॉलम फील्ड रो क्षेत्रावर किंवा रो फील्ड कॉलम क्षेत्रामध्ये हलवा.
- एक्सटर्नल रो आणि कॉलम आयटमसाठी सेल विलीन करा किंवा विलीन करा.

### पुढील गोष्टी करून रिक्त जागा आणि त्रुटींचे प्रदर्शन बदला

- एरर आणि रिक्त सेल कसे प्रदर्शित केले जातात ते बदला.
- डेटाशिवाय आयटम आणि लेबल कसे दाखवले जातात ते बदला.
- रिक्त रो प्रदर्शित करा किंवा लपवा

### खालील गोष्टी करून फॉर्मॅट बदला

- मॅन्युअली आणि कंडीशनल सेल आणि रेंज फॉर्मॅट करा.
- संपूर्ण PivotTable फॉर्मॅट स्टाइल बदला.
- फील्डसाठी नंबर फॉर्मॅट बदला.
- OLAP सर्व्हर फॉर्मॅटिंग समाविष्ट करा.

### पिव्होट टेबल आणि पिव्होट चार्ट तयार करा

तुम्ही PivotTable किंवा PivotChart साठी आधार म्हणून Excel वर्कशीटमधील डेटा वापरू शकता. डेटा पहिल्या रो मध्ये कॉलम लेबलांसह लिस्ट च्या स्वरूपात असावा, ज्याचा एकसेल फील्ड नावांसाठी वापर करेल. त्यानंतरच्या रो मधील प्रत्येक सेलमध्ये त्याच्या कॉलम शीर्षकासाठी योग्य डेटा असावा आणि तुम्ही त्याच कॉलम त डेटा टाइप मिसळू नये. उदाहरणार्थ, तुम्ही एकाच कॉलम त चलन व्हॅल्यू आणि तारखा मिक्स करू नये. याव्यतिरिक्त, डेटा रेन्ज मध्ये कोणत्याही रिक्त रो किंवा कॉलम असू नयेत.

### एक्सेल टेबल

ते आधीपासूनच लिस्ट च्या स्वरूपात आहेत आणि PivotTable सोर्स डेटासाठी चांगले उमेदवार आहेत. जेव्हा तुम्ही PivotTable रिफ्रेश करता, तेव्हा Excel टेबलमधील नवीन आणि अपडेट केलेला डेटा रिफ्रेश ऑपरेशनमध्ये ऑटोमॅटिक पणे समाविष्ट केला जातो.

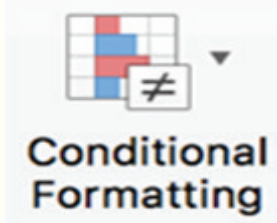
**डायनॅमिक नावाची रेन्ज वापरणे:** PivotTable अपडेट करणे सोपे करण्यासाठी, तुम्ही डायनॅमिक नावाची रेन्ज तयार करू शकता आणि ते नाव PivotTable चा डेटा सोर्स म्हणून वापरू शकता. अधिक डेटा समाविष्ट करण्यासाठी नामित रेन्ज विस्तृत झाल्यास, PivotTable रिफ्रेश केल्याने नवीन डेटा समाविष्ट होईल.

**इन्क्लुडिंग टोटल:** Excel पिव्होटटेबलमध्ये ऑटोमॅटिक सबटोटल आणि भव्य बेरीज तयार करते. जर सोर्स डेटामध्ये ऑटोमॅटिक सब बेरजे आणि तुमची भव्य बेरीज असेल डेटा टॅबवरील आऊटलाइन ग्रुपमधील सबटोटल्स कमांड वापरून तयार केलेले, तुम्ही PivotTable तयार करण्यापूर्वी सबटोटल्स आणि ग्रँड बेरीज काढून टाकण्यासाठी तीच कमांड वापरा.

### कंडिशनल फॉर्मॅटिंग

#### कंडिशनल फॉर्मॅटिंग लागू करा

- 1 सेलची रेन्ज , टेबल किंवा संपूर्ण शीट निवडा ज्यावर तुम्ही कंडीशनल फॉर्मॅटिंग लागू करू इच्छिता.
- 2 होम टॅबवर, कंडिशनल फॉर्मॅटिंग वर क्लिक करा.
- 3 खालीलपैकी एक करा:



अधोरेखित करणे	हे करा
स्पेसिफिक सेलमधील व्हॅल्यू उदाहरणे म्हणजे या आठवड्यानंतरच्या तारखा किंवा 50 आणि 100 मधील नंबर किंवा स्कोअरच्या खालच्या 10%.	सेल नियम किंवा टॉप/बॉटम नियम हायलाइट करा आणि नंतर योग्य पर्यायावर क्लिक करा.
सेल रेन्ज तील व्हॅल्यू चा संबंध. संपूर्ण सेलमध्ये रंगाचा बँड एक्सटेंडेड करा. सर्वात मोठ्या शहरांमधील किमती किंवा लोकसंख्येची कम्प्यारीजन ही उदाहरणे आहेत.	डेटा बार कडे पॉइंट करा आणि नंतर तुम्हाला हवे असलेले फिल क्लिक करा.
सेल रेन्ज तील व्हॅल्यू चा संबंध. रंग स्केल लागू करते जेथे सेलच्या रंगाची तीव्रता रेन्ज च्या वरच्या किंवा खालच्या दिशेने मूल्याचे पोजिशन प्रतिबिंबित करते. क्षेत्रांमध्ये विक्री वितरण हे एक उदाहरण आहे.	कलर स्केलकडे निर्देश करा आणि नंतर तुम्हाला पाहिजे असलेल्या स्केलवर क्लिक करा.
सेल रेन्ज ज्यामध्ये व्हॅल्यू चे तीन ते पाच ग्रुप असतात, जेथे प्रत्येक गटाचा स्वतःचा थ्रेशोल्ड असतो. उदाहरणार्थ, खाली विक्री दर्शवणारे सेल हायलाइट करण्यासाठी तुम्ही तीन चिन्हांचा संच नियुक्त करू शकता \$80,000, \$60,000 च्या खाली आणि \$40,000 च्या खाली. किंवा आपण कदाचित ऑटोमोबाईलसाठी 5-पॉइंट रेटिंग सिस्टम नियुक्त करा आणि पाच चिन्हांचा संच लागू करा.	आयकॉन सेट्सकडे इंगित करा आणि नंतर सेटवर क्लिक करा.

## पॉवर केरी

पॉवर केरी (एक्सेलमध्ये गेट आणि ट्रान्सफॉर्म म्हणून ओळखले जाते) सह, तुम्ही एक्सटर्नल डेटा आयात करू शकता किंवा कनेक्ट करू शकता आणि नंतर त्या डेटाला आकार देऊ शकता, उदाहरणार्थ कॉलम काढा, डेटा टाइप बदला किंवा टेबल्स मर्ज करा, तुमच्या गरजा पूर्ण करतात. त्यानंतर, चार्ट आणि रिपोर्ट तयार करण्यासाठी तुम्ही तुमची केरी Excel मध्ये लोड करू शकता. वेळोवेळी, तुम्ही डेटा अद्ययावत करण्यासाठी रीफ्रेश करू शकता. पॉवर केरी तीन एक्सेल ऍप्लिकेशन्सवर सबलब्ध आहे, विंडोजसाठी एक्सेल, मॅकसाठी एक्सेल आणि वेबसाठी एक्सेल. सर्व पॉवर केरी मदत विषयांच्या सारांशासाठी, एक्सेल मदतीसाठी पॉवर केरी पहा.

## पॉवर केरीचे चार टप्पे

पॉवर केरी वापरण्याचे चार टप्पे आहेत.



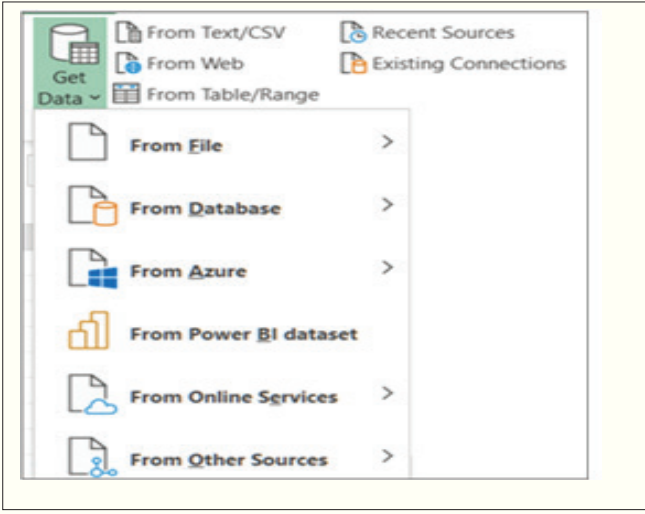
- कनेक्ट:** क्लाउडमध्ये, सेवेवर किंवा लोकल पातळीवर डेटाशी कनेक्शन करा
- ट्रान्सफॉर्म:** मूळ सोर्स अपरिवर्तित असताना, तुमच्या गरजा पूर्ण करण्यासाठी डेटाला आकार द्या

- एकत्र:** डेटामध्ये एक अद्वितीय व्ह मिळविण्यासाठी मल्टिपल स्त्रोतांकडून डेटा एकत्रित करा
- लोड:** तुमची केरी पूर्ण करा आणि ती वर्कशीट किंवा डेटा मॉडेलमध्ये लोड करा आणि वेळोवेळी रीफ्रेश करा.

पुढील प्रत्येक टप्पा अधिक तपशीलवार एक्सप्लोर करतात.

**कनेक्ट:** एक्सेल वर्कबुक सारख्या एकाच डेटा स्रोतावर किंवा क्लाउडवर विखुरलेल्या मल्टिपल डेटाबेस, फीड्स् किंवा सर्विस वर आयात करण्यासाठी तुम्ही Power Query वापरू शकता. डेटा स्रोतांमध्ये सध्याच्या वर्कबुकमधील वेब, फाइल्स, डेटाबेस, Azure किंवा अगदी Excel टेबलचा डेटा समाविष्ट आहे. Power Query सह, तुम्ही तुमची स्वतःची अनन्य परिवर्तने आणि संयोजने वापरून ते सर्व डेटा सोर्स एकत्र आणू शकता जे तुम्ही अन्यथा पाहिले नसते. एकदा आयात केल्यावर, तुम्ही रीफ्रेश करू शकता

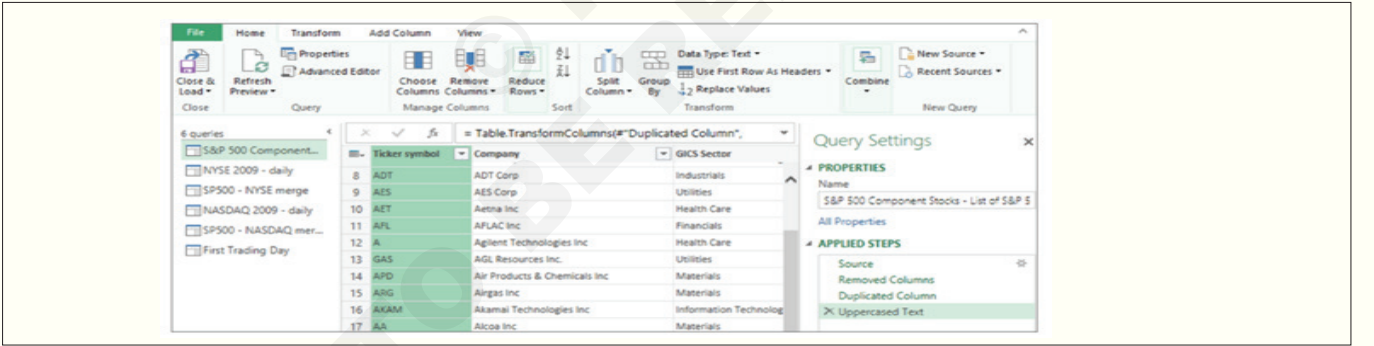
एक्सटर्नल डेटा स्त्रोतामधून जोडणे, बदल करणे आणि हटवणे यासाठी डेटा. अधिक इनफार्मेशन साठी, Excel मध्ये एक्सटर्नल डेटा कनेक्शन रीफ्रेश करा पहा.



## ट्रांसफॉर्म

डेटाचे रूपांतर करणे म्हणजे तुमच्या डेटा विश्लेषणाच्या आवश्यकता पूर्ण करण्यासाठी त्यात काही प्रकारे बदल करणे. उदाहरणार्थ, तुम्ही कॉलम काढू शकता, डेटा टाइप बदलू शकता किंवा रो फिल्टर करू शकता. यापैकी प्रत्येक ऑपरेशन डेटा ट्रांसफॉर्मेशन आहे. डेटाच्या एक किंवा अधिक संचांमध्ये परिवर्तन (आणि एकत्रीकरण) लागू करण्याच्या या प्रक्रियेला आकार देणे देखील म्हणतात.

असा विचार करा. फुलदाणी मातीच्या ढिगाच्याप्रमाणे सुरू होते जी एखाद्याला व्यावहारिक आणि सुंदर असे आकार देते. डेटा समान आहे. आपल्या गरजांसाठी योग्य आणि आकर्षक रिपोर्ट आणि डॅशबोर्ड सक्षम



**कंबाइन:** तुम्ही तुमच्या एक्सेल वर्कबुकमध्ये अनेक केरी जोडून किंवा विलीन करून एकत्र करू शकता. कोणत्याही केरीवर संलग्न आणि एकत्रीकरण ऑपरेशन्स केल्या जातात

टेबल आकारासह आणि डेटा ज्या डेटा स्रोतांमधून येतो त्यापासून स्वतंत्र असतात.

**संलग्न करा:** अपेड ऑपरेशन एक नवीन केरी तयार करते ज्यामध्ये पहिल्या केरीमधील सर्व रो असतात आणि त्यानंतर दुसऱ्या केरीमधील सर्व रो असतात. तुम्ही दोन टाइप चे संलग्न ऑपरेशन करू शकता:

- इंटरमीडिएट जोडणे :प्रत्येक संलग्न ऑपरेशनसाठी नवीन केरी तयार करते.
- इनलाइन संलग्न करा :तुम्ही अंतिम निकालापर्यंत पोहोचेपर्यंत तुमच्या विद्यमान केरीमध्ये डेटा जोडते.

करणार्या टेबल मध्ये आकार देणे आवश्यक आहे.

पॉवर केरी डेटा ट्रांसफॉर्मेशन्स सुलभ करण्यासाठी आणि प्रदर्शित करण्यासाठी पॉवर केरी एडिटर नावाची समर्पित विंडो वापरते. तुम्ही लॉक निवडून पॉवर केरी एडिटर उघडू शकता

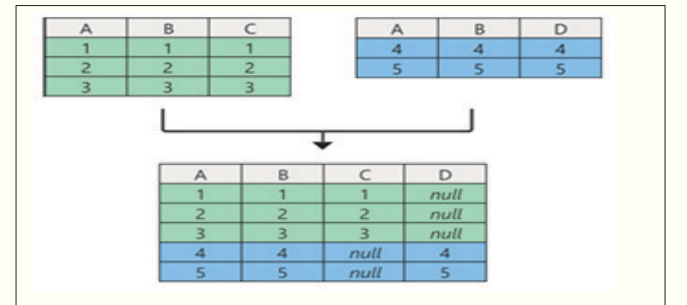
गेट अँड ट्रांसफॉर्म डेटा ग्रुपमधील गेट डेटा कमांडमधील केरी एडिटर, परंतु जेव्हा तुम्ही डेटा स्रोताशी कनेक्ट करता, नवीन केरी तयार करता किंवा केरी लोड करता तेव्हा ते देखील उघडते.

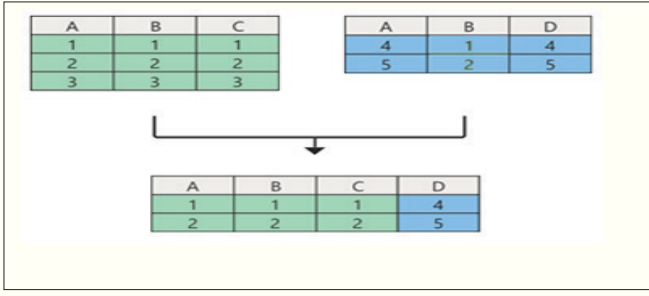
पॉवर केरी एडिटर तुम्ही डेटासह करत असलेल्या प्रत्येक ट्रांसफॉर्मेशन किंवा स्टेपला रेकॉर्ड करून आणि लेबल करून तुम्ही डेटाला लागू करत असलेल्या प्रत्येक गोष्टीचा मागोवा ठेवतो. ट्रांसफॉर्मेशन डेटा कनेक्शन असो, कॉलम काढणे, विलीन करणे किंवा डेटा टाइप बदल, तुम्ही केरी सेटिंग्ज सबखंडातील लागू चरण विभागात प्रत्येक परिवर्तन पाहू आणि सुधारित करू शकता.

युजर इंटरफेसमधून तुम्ही अनेक परिवर्तने करू शकता. प्रत्येक परिवर्तन बॅकग्राउंड त एक पाऊल म्हणून रेकॉर्ड केले जाते. तुम्ही एडवान्स संपादकातील Power Query M भाषा वापरून तुमची स्वतःची स्टेप सुधारू आणि लिहू शकता.

तुम्ही तुमच्या डेटा कनेक्शनवर लागू केलेले सर्व परिवर्तन एकत्रितपणे एक केरी बनवतात, जी मूळ (आणि न बदललेल्या) डेटा स्रोताचे नवीन प्रतिनिधित्व आहे. तुम्ही केरी रिफ्रेश केल्यावर, प्रत्येक स्टेप ऑटोमॅटिक चालते. एक्सेलमध्ये डेटा मॅन्युअली कनेक्ट आणि आकार देण्याची आवश्यकता केरी बदलतात.

**मर्ज:** मर्ज ऑपरेशन दोन विद्यमान केरींमधून नवीन केरी तयार करते. या एका केरीमध्ये प्रायमरी टेबल मधील सर्व कॉलम आहेत, एक कॉलम संबंधित टेबल च्या नेव्हिगेशन लिंक म्हणून सर्विस देत आहे. संबंधित टेबल मध्ये प्रायमरी टेबल मधील सामान्य कॉलम मूल्यापासून प्रत्येक रो शी जुळणाऱ्या सर्व रो असतात. शिवाय, तुम्ही संबंधित टेबल वरून प्रायमरी टेबल मध्ये कॉलम विस्तृत किंवा जोडू शकता. लोड





तुमच्या वर्कबुकमध्ये केरी लोड करण्याचे दोन मुख्य पाथ आहेत:

- पॉवर केरी एडिटरमधून, तुम्ही होम टॅबवरील क्लोज ग्रुपमधील क्लोज आणि लोड कमांड वापरू शकता.
- एक्सेल वर्कबुक केरी सबखंडातून (केरी आणि कनेक्शन निवडा), तुम्ही केरीवर उजवे-क्लिक करू शकता आणि लोड टू निवडा.

तुम्हाला तुमचा डेटा कसा पहायचा आहे आणि तुम्हाला डेटा कुठे लोड करायचा आहे हे निवडण्यासाठी क्वेरी ऑप्शन्स डायलॉग बॉक्स (फाइल निवडा > पर्याय आणि सेटिंग्ज > क्वेरी ऑप्शन्स निवडा) वापरून तुम्ही तुमच्या लोड ऑप्शन्सला फाइन-ट्यून करू शकता. एक डेटा मॉडेल (जे वर्कबुकमध्ये राहणाऱ्या अनेक टेबल्सचा रिलेशनल डेटा सोर्स आहे).

## एक्सेलमधील पॉवर केरीसाठी ऑप्लिकेशन सपोर्ट

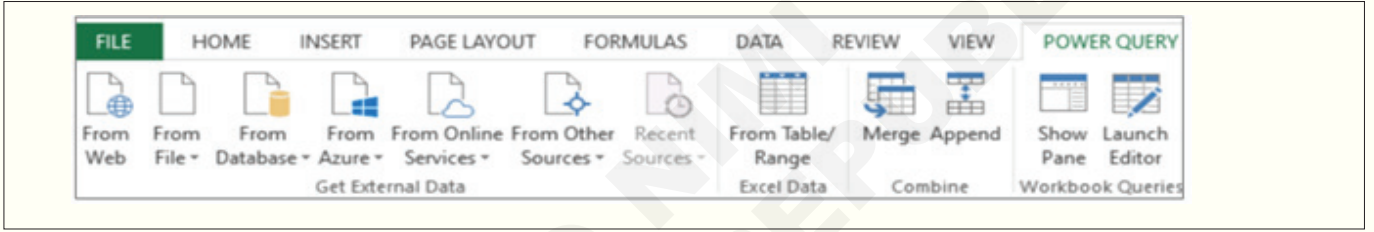
दहा वर्षांहून अधिक काळ, पॉवर केरी विंडोजसाठी एक्सेलवर समर्थित आहे. आता, एक्सेल मॅकसाठी एक्सेलवर पॉवर केरी सपोर्ट विस्तृत करत आहे आणि वेबसाठी एक्सेलला सपोर्ट जोडत आहे. याचा अर्थ आम्ही पॉवर केरी तीन प्रमुख प्लॅटफॉर्मवर सबलब्ध करून देत आहोत आणि पॉवर केरीची लोकप्रियता आणि कार्यक्षमता दर्शवितो. एक्सेल ग्राहक. Microsoft 365 रोडमॅपवर भविष्यातील घोषणा पहा आणि Microsoft 365 साठी Excel मध्ये नवीन काय आहे.

## एक्सेलमधील पॉवर केरीचा इतिहास

गेट अँड ट्रान्सफॉर्म डेटा (आता पॉवर केरी म्हटले जाते) चे एक्सेलमध्ये एकत्रीकरण गेल्या काही वर्षांमध्ये अनेक बदलांमधून गेले आहे.

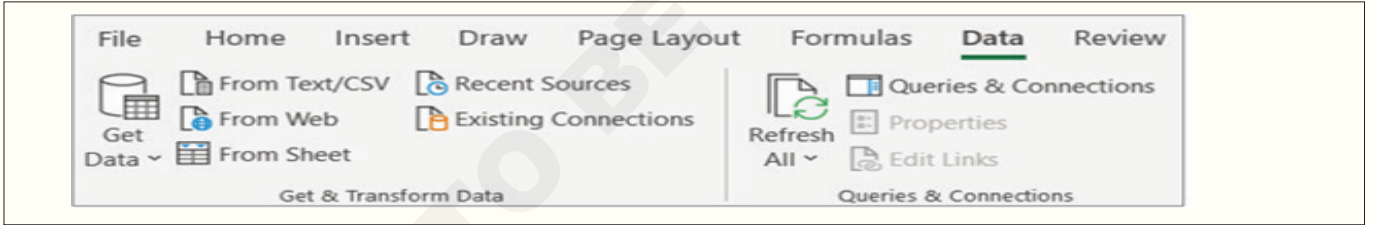
## Windows साठी Excel 2010 आणि 2013

Windows साठी Excel 2010 मध्ये, आम्ही प्रथम Power Query सादर केली आणि ती विनाव्हॅल्यु अँड-इन म्हणून सबलब्ध होती जी येथून डाउनलोड केली जाऊ शकते: Power Query अँडिन डाउनलोड करा. एकदा सक्षम केल्यावर, रिबनवरील पॉवर केरी टॅबमधून Power Query कार्यक्षमता सबलब्ध होती.



**मायक्रोसॉफ्ट 365:** डेटा इंपोर्ट आणि क्लीनिंगसाठी आम्ही Excel मध्ये प्रायमरी अनुभव म्हणून Power Query अपडेट केली आहे. तुम्ही Excel

रिबनच्या डेटा टॅबवरील डेटा गेट आणि ट्रान्सफॉर्म गटातून पॉवर केरी डेटा इंपोर्ट विझार्ड आणि टूल्समध्ये एक्सेस करू शकता.



या अनुभवामध्ये वर्धित डेटा आयात कार्यक्षमता, डेटा टॅबवर पुनर्रचना केलेल्या कमांड , नवीन केरी आणि कनेक्शन बाजूचे पॅन आणि क्रमवारी (सॉर्टिंग), डेटा टाइप बदलणे, कॉलम स्प्लिट करणे, डेटा एकत्रित करणे आणि अशाच प्रकारे शक्तिशाली मार्गानी डेटाला आकार देण्याची सतत क्षमता समाविष्ट आहे. .

या नवीन अनुभवाने गेट एक्सटर्नल डेटा ग्रुपमधील डेटा कमांड इंटरनल जुन्या, लेगसी डेटा इंपोर्ट विझार्डची जागा घेतली. तथापि, ते अद्याप Excel पर्याय डायलॉग बॉक्समधून एक्सेस केले जाऊ शकतात (फाइल निवडा > पर्याय > डेटा > लेगसी डेटा आयात विझार्ड दर्शवा).

## Windows साठी Excel 2016 आणि 2019

आम्ही मायक्रोसॉफ्ट 365 प्रमाणेच पॉवर केरी तंत्रज्ञानावर आधारित डेटा गेट आणि ट्रान्सफॉर्म अनुभव जोडला.

## Mac साठी Microsoft 365 साठी Excel

2019 मध्ये आम्ही Mac for Excel मध्ये Power Query ला सपोर्ट करण्यासाठी प्रवास सुरू केला. तेव्हापासून, आम्ही TXT, CSV, XLSX, JSON आणि XML फायलींमधील पॉवर केरी केरी रिफ्रेश करण्याची क्षमता जोडली आहे. आम्ही सध्याच्या कार्यपुस्तिकेमध्ये SQL सर्व्हरवरून आणि सारण्या आणि रेन्ज मधून डेटा रिफ्रेश करण्याची क्षमता देखील जोडली आहे.

2019 च्या ऑक्टोबरमध्ये, आम्ही विद्यमान पॉवर केरी केरी रिफ्रेश करण्याची आणि नवीन केरी तयार आणि एडिट करण्यासाठी VBA वापरण्याची क्षमता जोडली.

2021 च्या जानेवारीमध्ये, आम्ही OData आणि SharePoint स्रोतांकडील Power Query केरी रिफ्रेश करण्यासाठी सपोर्ट जोडले.



अधिक इनफार्मेशन साठी, Mac साठी Excel मध्ये Power Query वापरा पहा.

**टीप: Mac साठी Excel 2016 आणि Excel 2019 वर Power Query साठी कोणतेही सपोर्ट नाही.**

डेटा कॅटलॉग डिप्रेसेशन

डेटा कॅटलॉगसह, तुम्ही तुमच्या शेअर केलेल्या क्रेरी पाहू शकता, आणि नंतर त्यांना लोड करण्यासाठी, एडिट करण्यासाठी किंवा करंट वर्कबुकमध्ये वापरण्यासाठी निवडू शकता. हे वैशिष्ट्य हळूहळू नापसंत केले गेले:

- 1 ऑगस्ट 2018 रोजी, आम्ही नवीन ग्राहकांना डेटा कॅटलॉगवर ऑनबोर्ड करणे थांबवले.

- 3 डिसेंबर 2018 रोजी, वापरकर्ते डेटा कॅटलॉगमध्ये नवीन किंवा अपडेट केलेल्या क्रेरी शेअर करू शकले नाहीत.
- 4 मार्च 2019 रोजी, डेटा कॅटलॉगने काम करणे थांबवले. या तारखेनंतर, आम्ही तुमच्या सामायिक केलेल्या क्रेरी डाउनलोड करण्याची शिफारस केली आहे जेणेकरून तुम्ही माझा डेटा कॅटलॉग क्रेरी टास्क पेनमधील ओपन पर्याय वापरून डेटा कॅटलॉगच्या बाहेर त्यांचा वापर सुरू ठेवू शकता.

#### पॉवर क्रेरी अॅड-इन डिप्रेसेशन

2019 च्या उन्हाळ्याच्या सुरुवातीला, आम्ही Windows साठी Excel 2010 आणि 2013 साठी आवश्यक असलेले Power Query अॅड-इन अधिकृतपणे नापसंत केले. सौजन्य म्हणून, तुम्ही अजूनही अॅड-इन वापरू शकता, परंतु हे नंतरच्या तारखेला बदलू शकते.

## ई-कॉमर्स साइट्स ब्राउझ करा (E-Commerce scope and benenfits)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- ई-कॉमर्स परिभाषित करा
- पारंपारिक कॉमर्स आणि ई-कॉमर्समधील फरक स्पष्ट करा
- ई-कॉमर्सचा टाइप , व्याप्ती आणि फायदे स्पष्ट करा

### ईकॉमर्स

ई कॉमर्स ही उत्पादने किंवा सर्विसेज खरेदी आणि विक्री आणि इलेक्ट्रॉनिक मेथड्स ने निधी हस्तांतरित करण्याची प्रक्रिया आहे.

इलेक्ट्रॉनिक कॉमर्स मोबाइल कॉमर्स, इलेक्ट्रॉनिक फंड ट्रान्सफर, सप्लाय चेन मॅनेजमेंट, इंटरनेट मार्केटिंग, ऑनलाइन ट्रांझॅक्शन प्रोसेसिंग, इलेक्ट्रॉनिक डेटा इंटरचेंज (EDI), इन्व्हेंटरी मॅनेजमेंट सिस्टम आणि ऑटोमेटेड डेटा कलेक्शन सिस्टम यासारख्या तंत्रज्ञानावर आकर्षित करतो. आधुनिक इलेक्ट्रॉनिक कॉमर्स सामान्यतः ट्रांसक्शन च्या जीवन चक्राच्या किमान एक भागासाठी वर्ल्ड वाइड वेब वापरतो, जरी ते ई-मेल सारख्या इतर तंत्रज्ञानाचा देखील वापर करू शकते.

ई-कॉमर्स व्यवसाय सहसा खालीलपैकी काही किंवा सर्व मेथड्स वापरतात:

ऑनलाइन कॅटलॉगसह वेबसाइटवर Etail किंवा “व्हर्च्युअल स्टोअरफ्रंट्स” प्रदान करा, काहीवेळा “व्हर्च्युअल मॉल” मध्ये एकत्र केले जातात.

वेबसाइट्स किंवा ऑनलाइन बाजारपेठेवर खरेदी किंवा विक्री करा.

वेब संपर्क आणि सोशल मीडियाद्वारे लोकनंबर शास्त्रीय डेटा गोळा करा आणि वापरा. इलेक्ट्रॉनिक डेटा इंटरचेंज, व्यवसाय-ते-व्यवसाय डेटाची देवाणघेवाण वापरा.

ई-मेल किंवा फॅक्सद्वारे (उदाहरणार्थ, वृत्तपत्रांसह) संभाव्य आणि इंस्टाल ग्राहकांपर्यंत पोहोचा.

व्यवसाय ते व्यवसाय खरेदी आणि विक्री वापरा.

सुरक्षित व्यवसाय व्यवहार प्रदान करा.

6 ऑक्टोबर 2014 रोजी फ्लिपकार्टच्या बिग बिलियन डे सेलला सर्व ऑनलाइन विक्रीची जननी म्हणून ओळखले जात होते, परंतु मोठ्या दिवशी, गोष्टी विस्कळीत झाल्या आणि ग्राहकांना उच्च आणि कोरडे वाटले. ते पुरेशा प्रमाणात तयार नसल्याबद्दल संस्थापकांनी तत्काळ माफी मागितली असली तरी, वस्तुस्थिती अशी आहे की आपले देश मोठ्या प्रमाणावर ई-कॉमर्सकडे वळत आहेत.

स्फोटक वाढीसाठी तयार असलेल्या \$3 अब्ज क्षेत्रामधील स्वारस्य कमी करण्यासाठी यासारख्या एका घटनेने फारसे काही होणार नाही. Amazon आणि Snapdeal सारख्या Flipkart चे स्पर्धक आधीच या अनुभवातून शिकले आहेत आणि कदाचित भविष्यात अशा घटनांची पुनरावृत्ती होऊ नये म्हणून प्रयत्न करतील.

द असोसिएट चेंबर्स ऑफ कॉमर्स अँड इंडस्ट्री इन इंडिया (ASSOCHAM) नुसार, ऑनलाइन खरेदीचा जोर कायम आहे आणि 10,000 कोटी रुपयांचा टप्पा ओलांडला जाईल जो गेल्या वर्षीच्या खंडांपेक्षा 350 टक्के अधिक आहे.

### पारंपारिक कॉमर्स वि. ई कॉमर्स:

पारंपारिक कॉमर्समध्ये खरेदीदाराला एखादे उत्पादन किंवा सर्विसेज खरेदी करण्यासाठी दुकानात जावे लागते, परंतु ई कॉमर्समध्ये, उत्पादन किंवा सर्विसेज वेब साइटवर लिस्ट बद्ध केल्या जातात आणि खरेदीदार त्याचे फोटो आणि वेबवर दिलेल्या इतर तपशीलांची पडताळणी करून उत्पादन निवडतो. साइट्स आणि नंतर डेबिट/क्रेडिट कार्ड किंवा इंटरनेट बँकिंग मेथड्स ने प्राधान्याने इलेक्ट्रॉनिक मेथड्स ने किंमत भरणे. त्यानंतर कंपनीकडून ग्राहकाच्या पत्त्यावर उत्पादन पोहोचवले जाते. डिलिव्हरी शुल्क विनाव्हॅल्यु असू शकते किंवा ते उत्पादनाच्या किंमतीसह दिले जाऊ शकते.

पारंपारिक कॉमर्समध्ये ग्राहक उत्पादनाची प्रत्यक्ष पडताळणी करू शकतो, उदाहरणार्थ जर कोणी LED टीव्ही खरेदी करत असेल तर तो दुकानात जाऊन स्वतः चित्राचा दर्जा स्वतः सत्यापित करू शकतो, परंतु ई-कॉमर्स वेबसाइटमध्ये हे शक्य नाही. केवळ खरेदीदार उत्पादनाचे फोटो पाहू शकतो परंतु पिक्चर गुणवत्ता किंवा आवाज गुणवत्ता स्वतः पाहू शकत नाही.

ई-कॉमर्समध्ये, ग्राहक दुकानात प्रत्यक्ष न जाता त्याचा वेळ आणि पैसा वाचवू शकतो. परंतु उत्पादने खरेदी करण्यासाठी ग्राहकाला इंटरनेट कनेक्शन आणि डिव्हाइस आवश्यक आहे. पारंपारिक कॉमर्समध्ये खरेदीदार विक्रेत्याला प्रत्यक्ष भेटू शकतो त्यामुळे विक्रीनंतर सर्विसेज आवश्यक असल्यास, खरेदीदार विक्रेत्याशी संपर्क साधू शकतो परंतु ई-कॉमर्समध्ये खरेदीदाराला विक्रेत्याला प्रत्यक्ष भेटण्याची संधी मिळत नाही. त्यामुळे ग्राहकाने संबंधित उत्पादनाच्या सर्विसेज केंद्रांशी संपर्क साधून विक्रीपश्चात सेवेची व्यवस्था करावी.

उत्पादनांच्या किंमती हा देखील पारंपारिक व्यापारात एक एलिमेंट आहे कारण उत्पादने दुकानात साठवली पाहिजेत आणि दुकान चालवण्यासाठी कर्मचारी देखील नियुक्त केले जातात, दुकानासाठी भाडे किंवा ते विक्रेत्याच्या मालकीचे असावे म्हणून किंमत जास्त असते. ई-कॉमर्समध्ये उत्पादने खरेदीसाठी वाहतूक केली जात नाहीत त्यामुळे दुकानाची देखभाल करण्याचे ओव्हरहेड उत्पादनाच्या किंमतीत जोडले जात नाही. त्यामुळे ई-कॉमर्स साइट्समध्ये उत्पादने स्वस्त आहेत.

जरी काही लोक ई-कॉमर्स साइट्सचा लाभ घेऊ शकतात. बहुतेक शहरी भागात राहणाऱ्यांना याचा फायदा होतो. कारण ही सर्विस सामान्यतः शहरी भागांपुरतीच मर्यादित असते कारण गावकरी इंटरनेट सक्षम उपकरणे वापरण्यास आणि ऑनलाइन व्यवहार करण्यासाठी सुसज्ज नसतात. म्हणूनच ई-कॉमर्स साइट्स प्रथम खरेदीदारांना उत्पादन पाठवले जाऊ शकते की नाही हे सत्यापित करण्यासाठी खरेदीदारांचा पिन कोड क्रमांक तपासतात.

भविष्यात तंत्रज्ञानाच्या प्रगतीमुळे ही सर्विस देशभरात सबलब्ध होऊ शकते.

फिल्पकार्ट, अॅमेझॉन इत्यादीसारख्या ई-कॉमर्स साइट्सच्या आगमनाने पारंपारिक दुकानाची विक्री कमी झाली आहे कारण बहुतेक शहरी खरेदीदार आता उत्पादने खरेदी करण्यासाठी ई-कॉमर्स साइट्सना प्राधान्य देत आहेत ते त्यांच्या घरी आरामात बसून उत्पादन खरेदी करू शकतात.

ग्राहकाला दुकानापेक्षा ई-कॉमर्स साइटवर अधिक उत्पादने पाहण्याची संधी मिळू शकते. त्यामुळे पर्याय अधिक आहे आणि ई-कॉमर्स साइट्स देखील खरेदीदारांच्या घरी उत्पादन पुरवतात जी दुकाने सहसा करत नाहीत.

### ई कॉमर्सचा टाइप आणि त्याची व्याप्ती:

ई-कॉमर्स चार रेन्ज मध्ये विपार्ट ले जाऊ शकते, जे व्यवसाय ते व्यवसाय b2b, व्यवसाय ते ग्राहक b2c, सरकार ते व्यवसाय g2b आणि सरकार ते नागरिक g2c आहेत.

चार मुख्य एरिया आहेत ज्यात कंपनी ईकॉमर्स करतात ही एरिया आहेत:

- थेट विपणन, विक्री आणि सर्विस .
- ऑनलाइन बँकिंग आणि बिलिंग.
- इनफार्मेशन चे सुरक्षित वितरण.
- व्हॅल्यू साखळी व्यापार आणि कॉर्पोरेट खरेदी.
- सरकारला कर विवरणपत्र भरणे.

ई-कॉमर्सचे एरिया खूप विस्तृत आहे. ई-कॉमर्सचे अनेक ऍप्लिकेशन्स आहेत जसे की होम बँकिंग, इलेक्ट्रॉनिक मॉल्समध्ये खरेदी करणे, स्टॉक खरेदी करणे, नोकरी सर्च पे, लिलाव करणे, जगभरातील व्यावसायिक भागीदारांसह इलेक्ट्रॉनिक मेथड्स ने कोलायब्रेशन करणे आणि ग्राहक सर्विस प्रदान करणे. विविध ई-कॉमर्स ऍप्लिकेशन्सची अंमलबजावणी लोक, सार्वजनिक धोरण आणि विपणन/जाहिरात आणि पुरवठा साखळी लॉजिस्टिक यासारख्या चार प्रमुख सपोर्ट रेन्ज वर अवलंबून असते. याशिवाय पायाभूत सुविधा पुरवणे आवश्यक आहे. प्रत्येक संस्थेतील ई-कॉमर्स मॅनेजमेन्ट सह ऍप्लिकेशन आणि पायाभूत सुविधा व्यवस्थित करते. संबंध स्पष्ट करण्यासाठी मी B2C ईकॉमर्सच्या बाबतीत ऍप्लिकेशन खाली स्पष्ट केले आहे.

### ई कॉमर्सचे फायदे:

- ई कॉमर्स लोकांना वेळ किंवा अंतराच्या अडथळ्यांशिवाय व्यवसाय करण्यास अनुमती देते. कोणीही इंटरनेटवर कधीही लॉग इन करू शकतो, मग तो दिवस असो वा रात्र असो आणि माऊसच्या एका क्लिकवर आपल्या इच्छेनुसार काहीही खरेदी किंवा विक्री करू शकतो.

- वेब साईटवरून घेतलेल्या ऑर्डरसाठी थेट विक्रीची किंमत पारंपारिक माध्यमांच्या (किरकोळ, कागदावर आधारित) पेक्षा कमी आहे, कारण ऑनलाइन इलेक्ट्रॉनिक खरेदी ऑर्डर प्रक्रियेदरम्यान कोणताही मानवी संवाद नसतो. तसेच, इलेक्ट्रॉनिक विक्री प्रक्रिया त्रुटी दूर करते, तसेच अभ्यागतांसाठी जलद आणि अधिक सोयीस्कर आहे.

- ई कॉमर्स स्पेसिफिक उत्पादनांसाठी आदर्श आहे. अशा उत्पादनांचे ग्राहक सहसा कमी असतात. परंतु मोठ्या बाजारपेठेत म्हणजे इंटरनेट, अगदी स्पेसिफिक उत्पादने देखील व्यवहार्य व्हॉल्यूम निर्माण करू शकतात.

- ई कॉमर्सचा आणखी एक महत्त्वाचा फायदा म्हणजे ते व्यवसाय करण्याचे सर्वात स्वस्त साधन आहे.

- बाजारातील दैनंदिन दबावाने त्यांची स्पर्धात्मक स्थिती सुधारण्यासाठी कंपन्यांना गुंतवणूक करण्याच्या संधी कमी करण्यात त्यांची भूमिका बजावली आहे. परिपक्व बाजारपेठ, वाढलेल्या स्पर्धामुळे गुंतवणुकीसाठी सबलब्ध पैशांची रक्कम कमी झाली आहे. जर विक्री किंमत वाढवता येत नसेल आणि उत्पादित किंमत कमी करता येत नसेल तर फरक हा व्यवसाय चालविण्याच्या मेथड्स त असू शकतो. ईकॉमर्सने खर्च कमी करून समाधान प्रदान केले आहे.

- खरेदीदारांच्या दृष्टीकोनातून देखील ईकॉमर्स बरेच मूर्त फायदे देते.

- 1 खरेदीदारांच्या वर्गीकरण्याच्या वेळेत कपात.
  - 2 चांगले खरेदीदार निर्णय.
  - 3 बीजक आणि ऑर्डरमधील विसंगतीचे निराकरण करण्यात कमी वेळ घालवला जातो. 4 पर्यायी उत्पादने खरेदी करण्याच्या वाढीव संधी.
- व्यवसायाला 'ई कॉमर्स सक्षम' बनवण्याचा धोरणात्मक फायदा असा आहे की, यामुळे डिलिव्हरी वेळ, मजुरीचा खर्च आणि खालील क्षेत्रांमध्ये होणारा खर्च कमी होण्यास मदत होते:

- 1 डॉक्युमेंट्स तयार करणे
- 2 त्रुटी सर्च पे आणि सुधारणे
- 3 सलोखा
- 4 मेल तयार करणे
- 5 दूरध्वनी कॉलिंग
- 6 क्रेडिट कार्ड मशीन
- 7 डेटा एंट्री
- 8 ओव्हरटाइम
- 9 पर्यवेक्षण खर्च

- ई-कॉमर्सच्या ऑपरेशनल फायद्यांमध्ये व्यवसाय प्रक्रिया पूर्ण करण्यासाठी लागणारा वेळ आणि कर्मचारी दोन्ही कमी करणे आणि इतर संसाधनांवरील ताण कमी करणे समाविष्ट आहे. या सर्व फायद्यांमुळेच एखादी व्यक्ती ईकॉमर्सच्या शक्तीचा यूजेस करू शकते आणि ई बिझनेस सोल्यूशन प्रदात्यांद्वारे सबलब्ध केलेल्या शक्तिशाली ई कॉमर्स सोल्यूशन्सचा वापर करून व्यवसायाला ई व्यवसायात रूपांतरित करा.

**ई-कॉमर्स साइटवर व्यवहार करा (Undertake transactions on an e-commerce site)**

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- वेगवेगळ्या ई-कॉमर्स साइट्सचे स्पष्टीकरण द्या
- ऑनलाइन कॅटलॉग, शॉपिंग कार्ट आणि चेक आउट पेजेज समजावून सांगा
- पेमेंट, ऑर्डर प्रक्रिया आणि अधिकृतता स्पष्ट करा, शुल्क परत करा
- इतर पेमेंट पर्याय स्पष्ट करा

**विविध ई कॉमर्स साइट्स**

जगातील सर्वात लोकप्रिय ई कॉमर्स साइट्सपैकी काही आहेत: क्रेट आणि बॅरल, सिमेंटेक, अॅमवे, मायक्रोसॉफ्ट, अॅमेझॉन, एचपी इ.

भारतात, सुरुवातीच्या काही अडथळ्यांनंतर, ई कॉमर्स हळूहळू वेग घेत आहे. काही लोकप्रिय वेब साइट्स आहेत:

Amazon, Flipkart, Jabong, Naaptol इ.

**ऑनलाइन कॅटलॉग, शॉपिंग कार्ट आणि चेकआउट पेजेज**

ऑनलाइन कॅटलॉग ही वेबसाइटवर विक्रीसाठी दिलेल्या उत्पादनांची यादी असते. खरेदीदार उत्पादनांमधून ब्राउझ करून आणि त्याला/तिला अनुकूल असलेले उत्पादन निवडून उत्पादन निवडतो.

खरेदी कार्ट म्हणजे शेवटी किंमत देण्यापूर्वी खरेदीदाराने निवडलेल्या उत्पादनांनी भरलेली बादली. हे किरकोळ स्टोरमध्ये वापरले जाते परंतु हीच संकल्पना वेब साईटमध्ये व्हर्च्युअल बकेट बनवून अंमलात आणली गेली आहे जी खरेदीदाराने निवडलेले उत्पादन दर्शवते.

इच्छित उत्पादन निवडल्यानंतर खरेदीदार उत्पादनांची किंमत देण्यासाठी शेवटी चेकआउटवर क्लिक करतो.

**पेमेंट आणि ऑर्डर प्रक्रिया**

चेकआउटवर क्लिक केल्यानंतर एक पेमेंट पर्याय दिसतो ज्यामध्ये साधारणपणे COD (कॅश ऑन डिलिव्हरी), इंटरनेट बँकिंग, डेबिट कार्ड,

क्रेडिट कार्ड आणि इतर विविध पर्याय असतात. खरेदीदार पेमेंटसाठी योग्य पर्याय निवडतो आणि यशस्वी पेमेंट केल्यानंतर, ते खरेदीदाराला सूचित केले जाते आणि शेवटी उत्पादने खरेदीदाराला पत्त्यावर वितरित केली जातात.

**अधिकृतता आणि फी चार्जबॅक**

अधिकृतता किंवा अधिकृतता हे सर्वसाधारणपणे इनफार्मेशन सुरक्षितता आणि संगणक सुरक्षितता आणि स्पेशल तः एक्सेस नियंत्रणाशी संबंधित संसाधनांवर एक्सेस अधिकार स्पेसिफाइड करण्याचे कार्य आहे.

चार्जबॅक म्हणजे खरेदीदाराकडून किंमत वजा केल्यावर पैसे परत देणे. हे विविध परिस्थितींमध्ये घडते. उदाहरणार्थ, समजा एखाद्याने irtc वेब साइटवर तिकीट खरेदी केले आणि तिकिटाची किंमत यशस्वीरित्या भरली, परंतु नंतर तिकीट आरक्षित झाले नाही. नंतर शुल्क परत येईल आणि पैसे ग्राहकाला परत केले जातील.

**इतर पेमेंट पर्याय**

वरील चर्चा केलेल्या पेमेंट पर्यायांशिवाय पेमेंटचे इतर काही पाथ आहेत जसे की मोबाईल पेमेंट. अलीकडे Airtel Money किंवा Vodafone mpesa इत्यादी मोबाईल पेमेंट मेथड्स विकसित झाल्या आहेत ज्यामुळे व्यक्ती त्यांच्या मोबाईलद्वारे देखील पैसे देऊ शकतात.

विविध ई कॉमर्स व्यवहार जसे की युटिलिटी बिले भरणे, वेब साइट्सवरून खरेदी करणे, रिचार्ज करणे इत्यादी मोबाईलद्वारे करता येतात.

## ओळखण्यासाठी ई-कॉमर्स साइट ब्राउझ करा (E-Commerce security issues and payment gateway)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- ई-कॉमर्स सुरक्षा समस्या स्पष्ट करा
- पेमेंट गेटवे समजावून सांगा.

### ई कॉमर्स सुरक्षा समस्या

ई-कॉमर्स सुरक्षा म्हणजे ई-कॉमर्स मालमत्तेचे अनधिकृत एक्सेस , वापर, बदल किंवा विनाश यापासून प्रोटेक्शन . सुरक्षा वैशिष्ट्ये सुरक्षित सिस्टमची हमी देत नाहीत, तरीही सुरक्षित सिस्टम तयार करण्यासाठी ते आवश्यक आहेत.

ई-कॉमर्सच्या वाढीमुळे नवीन पिढीशी संबंधित सुरक्षा धोके निर्माण झाली आहेत, परंतु कोणत्याही ई-कॉमर्स सिस्टिम ने चार अविभाज्य आवश्यकता पूर्ण केल्या पाहिजेत:

- प्रायव्हसी - देवाणघेवाण केलेली इनफार्मेशन अनधिकृत पक्षांकडून ठेवली पाहिजे.
- अखंडता - देवाणघेवाण केलेल्या इनफार्मेशन मध्ये बदल किंवा छेडछाड केली जाऊ नये.
- प्रमाणीकरण - प्रेषक आणि प्राप्तकर्ता दोघांनीही त्यांची ओळख एकमेकांना सिद्ध करणे आवश्यक आहे आणि
- नॉन-रिड्युएशन - देवाणघेवाण केलेली इनफार्मेशन खरोखरच प्राप्त झाली याचा पुरावा आवश्यक आहे.

### सुरक्षा धोके

तांत्रिक हल्ले हे सर्वात आव्हानात्मक टाईप च्या सुरक्षा तडजोडीपैकी एक आहेत ज्यांना ई-कॉमर्स प्रदात्याला तोंड द्यावे लागते. तांत्रिक हल्ल्यांचे गुन्हेगार, आणि स्पेशल त: डिनायल-ऑफ-सर्व्हिस हल्ले, स्पेशल त: बँका, क्रेडिट कार्ड पेमेंट गेटवे, मोठे ऑनलाइन किरकोळ विक्रेते आणि लोकप्रिय सोशल नेटवर्किंग साइट्स यासारख्या हाय-प्रोफाइल वेब सर्व्हरवर होस्ट केलेल्या साइट्स किंवा सर्विस ना टारगेट करतात.

### सर्विस हल्ले नाकारणे

डिनायल ऑफ सर्व्हिस (DoS) हल्ल्यांमध्ये सर्व्हर, नेटवर्क किंवा वेबसाइटची सामान्य गतिविधी लकवा बनवण्याचा समावेश असतो. DoS हल्ल्यांपासून बचाव करणे ही आज इंटरनेटवरील सर्वात आव्हानात्मक सुरक्षा समस्यांपैकी एक आहे. या हल्ल्यांना आळा घालण्यात एक मोठी अडचण म्हणजे हल्ल्याचा सोर्स सर्व्हर, कारण ते आक्रमणाचे खरे मूळ सर्व्हर प्यासाठी अनेकदा चुकीचे किंवा बनावट IP सोर्स एड्रेस वापरतात.

युनायटेड स्टेट्स कॉम्प्युटर इमर्जन्सी रेडिनेस टीम सर्विस नाकारण्याच्या हल्ल्याची लक्षणे परिभाषित करते:

- असामान्यपणे धीमे नेटवर्क कार्यप्रदर्शन
- स्पेसिफिक वेब साइटची अनुपलब्धता
- कोणत्याही वेब साइटवर एक्सेस करण्यास असमर्थता

- प्राप्त झालेल्या स्पॅम ईमेलच्या संख्येत नाटकीय वाढ.

DoS हल्ले यासह अनेक वेगवेगळ्या मार्गांनी अंमलात आणले जाऊ शकतात:

### ICMP flood (Smurf Attack)

जिथे गुन्हेगार मोठ्या संख्येने आयपी पॅकेट पाठवतील ज्याचा सोर्स एड्रेस पिडीत व्यक्तीचा एड्रेस असल्याचे खोटे केले जाईल. नेटवर्कची बँडविड्थ त्वरीत वापरली जाते, कायदेशीर पॅकेट्स त्यांच्या डेस्टिनेशन स्थानापर्यंत जाण्यापासून प्रतिबंधित करते.

### टीयरड्रॉप हल्ला

टीयरड्रॉप अटॅकमध्ये टारगेट मशीनवर ओव्हर लॉपिंग, ओव्हर साइज, पेलोडसह मँगलड आयपीचे तुकडे पाठवणे समाविष्ट असते. विविध ऑपरेटिंग सिस्टिमच्या TCP/IP फ्रॅगमेंटेशन री असेंबली कोडमधील बगमुळे तुकड्यांना अयोग्यरित्या हाताळले जाते, परिणामी ते क्रॅश होतात.

### पलशिंग

कायमस्वरूपी नकार-ऑफ-सर्व्हिस (PDOS) म्हणूनही ओळखले जाणारे आक्रमण म्हणजे सिस्टमला इतके वाईट रीतीने नुकसान होते की त्याला हार्डवेअर बदलणे किंवा पुनर्स्थापित करणे आवश्यक आहे. गुन्हेगार पीडिताच्या हार्डवेअरच्या रिमोट मॅनेजमेंट इंटरफेसमधील सुरक्षा त्रुटींचा गैरफायदा घेतात, मग ते राउटर, प्रिंटर किंवा इतर नेटवर्किंग हार्डवेअर असो. या त्रुटींमुळे आक्रमणकर्त्याला डिव्हाइस फर्मवेअरला सुधारित, करपेट किंवा सडोष फर्मवेअर प्रतिमेवर दूरस्थपणे 'अपडेट' करण्यासाठी दरवाजा मोकळा होतो, त्यामुळे डिव्हाइसला वीट बसते आणि मूळ उद्देशासाठी ते कायमचे निरुपयोगी बनवणे.

### वितरित नकार-ऑफ-सर्व्हिस हल्ले

डिस्ट्रिब्युटेड डिनायल ऑफ सर्व्हिस (DDoS) हल्ले हे आयटी व्यवस्थापकांसाठी सर्वात मोठी सुरक्षा भीती आहे. काही मिनिटांत, हजारो असुरक्षित संगणक कायदेशीर रहदारी गुदमरून पीडित वेबसाइटला पूर देऊ शकतात. डिस्ट्रिब्युटेड डिनायल ऑफ सर्व्हिस अटॅक (DDoS) तेव्हा उद्भवते जेव्हा मल्टिपल तडजोड सिस्टिम लक्षित सिस्टिम ची बँड रुंदी किंवा रिसोर्सेस , सामान्यतः एक किंवा अधिक वेब सर्व्हरमध्ये पूर येतात. सर्वात प्रसिद्ध DDoS हल्ले फेब्रुवारी 2000 मध्ये झाले ज्यात Yahoo, Buy.com, eBay, Amazon आणि CNN सारख्या वेबसाइटवर हल्ला करण्यात आला आणि प्रत्येकी अनेक तासांपर्यंत पोहोचण्यायोग्य राहिल्या.

### ब्रूट फोर्स अटॅक

ब्रूट फोर्स अटॅक ही क्रिप्टोग्राफिक स्कीमला मोठ्या संख्येने शक्यता वापरून पराभूत करण्याची एक मेथड्स आहे; उदाहरणार्थ, संदेश डिक्रिप्ट

करण्यासाठी की स्पेसमध्ये संभाव्य की मोठ्या संख्येने. ब्रूट फोर्स अटॅक, जरी कमी तंत्रज्ञानाचे फॉरमॅट मानले जात असले तरी ती भूतकाळातील गोष्ट नाही.

## नॉन टेक्निकल अटॅक

फिशिंग ही इलेक्ट्रॉनिक कम्युनिकेशन मध्ये विश्वासाह संस्था म्हणून मुखवटा घालून युजर नाव, पासवर्ड आणि क्रेडिट कार्ड तपशील यासारखी सेन्सिटिव्ह इनफार्मेशन मिळविण्याचा प्रयत्न करण्याची गुन्हेगारी फसवणूक प्रक्रिया आहे. फिशिंग घोट्याळे सामान्यतः पीडित व्यक्तीला सेन्सिटिव्ह इनफार्मेशन ची विनंती करणारी कायदेशीर संस्था म्हणून 'फसव्या' ईमेलसह ईमेल करून चालते. जेव्हा पीडित व्यक्ती ईमेलमध्ये एम्बेड केलेल्या लिंकचे अनुसरण करते तेव्हा त्यांना कायदेशीर संस्थांच्या वेबसाइटच्या विस्तृत आणि अत्याधुनिक डुप्लिकेटवर आणले जाते. फिशिंग हल्ले सामान्यतः बँक खरेदीदार, ऑनलाइन लिलाव साइट (जसे की eBay), ऑनलाइन किरकोळ विक्रेते (जसे की amazon) आणि सर्विस प्रदाते (जसे की PayPal) टारगेट करतात. सामुदायिक बँकरच्या मते, अलीकडच्या काळात सायबर गुन्हेगार नैसर्गिक आपत्तीच्या वेळी धर्मादाय संस्था म्हणून त्यांच्या हल्ल्यांच्या वेळेत अधिक परिष्कृत झाले आहेत.

सोशल इंजिनीरिंग -सोशल इंजिनीरिंग ही कृती करण्यासाठी किंवा गोपनीय इनफार्मेशन उघड करण्यासाठी लोकांना हाताळण्याची कला आहे. सोशल इंजिनीरिंग तंत्रांमध्ये बहाणा करणे (जेथे फसवणूक करणारा पीडित व्यक्तीला इनफार्मेशन सांगण्यासाठी एक आविष्कृत परिस्थिती तयार करतो), इंटरएक्टिव्ह व्हॉईस रेकॉर्डिंग (IVR) किंवा फोन फिशिंग (जेथे फसवणूक करणारा पीडित व्यक्तीला फोनवर सेन्सिटिव्ह इनफार्मेशन प्रकट करतो) आणि ट्रोजनसह आमिष दाखवतो. घोडे (जेथे फसवणूक करणारा बळी लादण्यासाठी 'आमिष' देतो

सिस्टममध्ये मालवेअर). सोशल इंजिनीरिंग हे ई-कॉमर्स सुरक्षेसाठी एक गंभीर धोका बनले आहे कारण ते सर्व णे आणि सामना करणे कठीण आहे कारण त्यात 'मानवी' एलिमेंटचा समावेश आहे ज्यास हार्डवेअर किंवा सॉफ्टवेअरसारखे पॅच केले जाऊ शकत नाही, जरी कर्मचारी प्रशिक्षण आणि शिक्षण काही प्रमाणात हा हल्ला थोपवू शकतात.

## सुरक्षित कसे रहावे

### सुरक्षित वेब साइट्सवर खरेदी करा

तुमच्या संगणकावरून ऑनलाइन व्यापाऱ्याच्या संगणकावर इनफार्मेशन हस्तांतरित करण्यासाठी सुरक्षित साइट्स एन्क्रिप्शन तंत्रज्ञान वापरतात. एन्क्रिप्शन तुम्ही पाठवलेली इनफार्मेशन, जसे की तुमचा क्रेडिट कार्ड नंबर, संगणक हॅकर्सना मार्गात ती मिळवण्यापासून रोखण्यासाठी स्कॅम्बल करते. कायदेशीर एक्सेस विशेषाधिकार असलेले एकमेव लोक कोड अनस्कॅम्बल करू शकतात. तुम्ही सुरक्षित साइटवर व्यवहार करता तेव्हा तुम्ही कसे सांगू शकता ते येथे आहे:

जर तुम्ही तुमच्या स्क्रीनच्या शीर्षस्थानी वेब साइटचा एड्रेस ("अॅड्रेस बार") प्रदर्शित केला आहे असे पाहिले तर, तुम्हाला <https://> दिसले पाहिजे. "http" नंतर प्रदर्शित होणारा "s" संकेतस्थळ सुरक्षित असल्याचे सूचित करतो. बऱ्याचदा, जोपर्यंत तुम्ही वेब साइटवरील ऑर्डर पेजवर जात नाही तोपर्यंत तुम्हाला "s" दिसत नाही.

वेबसाइट सुरक्षित आहे की नाही हे निर्धारित करण्याचा दुसरा पाथ म्हणजे तुमच्या स्क्रीनच्या अॅड्रेस बारवर प्रदर्शित केलेला बंद पॅडलॉक पाहणे.

ते कुलूप उघडे असल्यास, ती सुरक्षित साइट नाही असे समजावे. अर्थात, जर व्यापारी बिनधास्तपणे डेटा स्टोर करत असेल तर सुरक्षित चॅनेलवर तुमचा डेटा प्रसारित करणे तुमच्यासाठी फारसे महत्त्वाचे नाही. व्यापारी डेटा एन्क्रिप्टेडमध्ये स्टोर करतो की नाही हे सर्व ण्याचा प्रयत्न करा

फॉर्म हॅकर घुसखोरी करण्यास सक्षम असल्यास, तो तुमचा क्रेडिट डेटा आणि इतर वैयक्तिक इनफार्मेशन मिळवू शकत नाही. व्यापार्याच्या संगणकावरील तुमचा वैयक्तिक डेटा कसा सुरक्षित ठेवतो हे जाणून घेण्यासाठी त्याची प्रायव्हसी आणि सुरक्षा धोरणे वाचा याची खात्री करा.

## आपण ऑर्डर करण्यापूर्वी वेबसाइटवर रिसर्च करा

तुम्हाला आधीच माहीत असलेल्या कंपन्यांसोबत व्यवसाय करा. कंपनी अपरिचित असल्यास, त्यांची उत्पादने खरेदी करण्यापूर्वी तुमचा गृहपाठ करा. तुम्ही एखाद्या अज्ञात कंपनीकडून एखादी वस्तू खरेदी करण्याचे ठरवल्यास, कंपनी विश्वासाह आहे की नाही हे जाणून घेण्यासाठी स्वस्त ऑर्डरसह सुरुवात करा.

विश्वासाह कंपन्यांनी त्यांचा प्रत्यक्ष व्यवसाय एड्रेस आणि किमान एक फोन नंबर, खरेदीदार सर्विस किंवा ऑर्डर लाइनची जाहिरात करावी. फोन नंबरवर कॉल करा आणि व्यवसाय कायदेशीर आहे की नाही हे निर्धारित करण्यासाठी प्रश्न विचारा. जरी तुम्ही तासांनंतर कॉल केला तरीही, बऱ्याच कंपन्यांकडे "लाइव्ह" उत्तर देणारी सर्विस आहे, स्पेशल तः त्यांना ऑर्डर चुकवायची नसल्यास. व्यापारी परत केलेला माल आणि तक्रारी कशा हाताळतात ते विचारा. ते पूर्ण returns देते की फक्त स्टोर क्रेडिट देते ते शोधा.

तुम्ही बेटर बिझनेस ब्युरो किंवा डिस्ट्रिक्ट अॅटर्नी ऑफिस किंवा अॅटर्नी जनरल सारख्या सरकारी ग्राहक प्रोटेक्शन एजन्सीद्वारे कंपनीचे रिसर्च न देखील करू शकता. कदाचित लिस्ट बद्ध शहरात राहणारे मित्र किंवा कुटुंबातील सदस्य कंपनीच्या वैधतेची पडताळणी करू शकतात. लक्षात ठेवा, कोणीही वेबसाइट तयार करू शकते.

## पेमेंट गेटवे

पेमेंट गेटवे ही एक ई-कॉमर्स ऍप्लिकेशन सर्विस प्रदाता सर्विस आहे जी ई-व्यवसाय, ऑनलाइन किरकोळ विक्रेते, ब्रिक्स आणि क्लिक्स किंवा पारंपारिक वीट आणि मोटारसाठी क्रेडिट कार्ड पेमेंट अधिकृत करते.

हे बहुतांश रिटेल आउटलेटमध्ये स्थित फिजिकल पॉइंट ऑफ सेल टर्मिनलच्या समतुल्य आहे. पेमेंट गेटवे क्रेडिट कार्ड क्रमांकांसारखी सेन्सिटिव्ह इनफार्मेशन एन्क्रिप्ट करून क्रेडिट कार्ड तपशीलांचे प्रोटेक्शन करतात, हे सुनिश्चित करण्यासाठी की इनफार्मेशन खरेदीदार आणि व्यापारी आणि व्यापारी आणि पेमेंट प्रोसेसर दरम्यान सुरक्षितपणे पास केली जाते.

पेमेंट गेटवे पेमेंट पोर्टल (जसे की वेबसाइट, मोबाइल फोन किंवा इंटरएक्टिव्ह व्हॉईस रिस्पॉन्स सर्विस ) आणि फ्रंट एंड प्रोसेसर किंवा अधिग्रहण बँक यांच्या दरम्यान इनफार्मेशन चे हस्तांतरण सुलभ करते.

## व्यवहार प्रक्रिया

- जेव्हा खरेदीदार पेमेंट गेटवे-सक्षम व्यापाऱ्याकडून उत्पादन ऑर्डर करतो, तेव्हा पेमेंट गेटवे ट्रान्सक्शन वर प्रक्रिया करण्यासाठी विविध फंक्शन करतो.

## इनफार्मेशन सुरक्षा आणि धोक्यांचे ओवरविह (Overview of information security and threats)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- इनफार्मेशन सुरक्षा आणि त्याच्या बेसिक तत्वांचे डिस्क्रिप्शन करा
- इनफार्मेशन सुरक्षा आणि सायबर सुरक्षा यांच्यातील संबंधांचे डिस्क्रिप्शन करा
- इनफार्मेशन सुरक्षिततेतील प्रमुख आव्हानांचे डिस्क्रिप्शन करा
- इनफार्मेशन सुरक्षिततेच्या फायद्यांचे डिस्क्रिप्शन करा
- इनफार्मेशन सुरक्षा लागू करण्याच्या मेथड्स स्पष्ट करा.

### परिचय

इनफार्मेशन सुरक्षा म्हणजे इनफार्मेशन आणि इनफार्मेशन प्रणालींचे अनधिकृत एक्सेस , वापर, प्रकटीकरण, व्यत्यय, बदल किंवा विनाश यापासून प्रायव्हसी , अखंडता आणि उपलब्धता प्रदान करण्यासाठी प्रोटेक्शन . ही एक सामान्य संज्ञा आहे जी डेटा कोणत्याही स्वरूपाची पर्वा न करता वापरली जाऊ शकते (इलेक्ट्रॉनिक, फिजिकल , इ.)

सरकार, सैन्य, कॉर्पोरेशन, वित्तीय संस्था, रुग्णालये आणि प्राइवेट व्यवसाय त्यांचे कर्मचारी, ग्राहक, उत्पादने, रिसर्च न आणि आर्थिक स्थिती याबद्दल मोठ्या प्रमाणात गोपनीय इनफार्मेशन स्टोर करतात. यातील बरीचशी इनफार्मेशन आता इलेक्ट्रॉनिक संगणकांवर संकलित, प्रक्रिया आणि स्टोर केली जाते आणि नेटवर्कवरून इतर संगणकांवर प्रसारित केली जाते.

एखाद्या व्यवसायाच्या ग्राहकांबद्दल किंवा वित्त किंवा नवीन उत्पादन लाइनबद्दलची गोपनीय इनफार्मेशन एखाद्या स्पर्धक किंवा हॅकरच्या हातात पडल्यास, व्यवसाय आणि त्याच्या ग्राहकांना मोठ्या प्रमाणावर, भरून न येणारे आर्थिक नुकसान होऊ शकते, कंपनीच्या प्रतिष्ठेला हानी पोहोचू शकते. गोपनीय इनफार्मेशन चे प्रोटेक्शन करणे ही एक व्यावसायिक आवश्यकता आहे आणि बर्बाद बाबतीत नैतिक आणि कायदेशीर आवश्यकता देखील आहे.

**इनफार्मेशन आश्वासन** : जेव्हा गंभीर समस्या उद्भवतात तेव्हा डेटा गमावला जाणार नाही याची खात्री करण्याची कृती म्हणजे इनफार्मेशन आश्वासन. या समस्यांमध्ये समाविष्ट आहे परंतु इतकेच मर्यादित नाही: नैसर्गिक आपत्ती, संगणक/सर्व्हर खराब होणे, फिजिकल चोरी किंवा डेटा गमावण्याची शक्यता असलेल्या इतर कोणत्याही इवेंट्स . आमच्या आधुनिक युगात बहुतेक इनफार्मेशन संगणकावर स्टोर केली जात असल्याने, इनफार्मेशन ची हमी सामान्यतः आयटी सुरक्षा तज्ञांद्वारे हाताळली जाते. इनफार्मेशन चे आश्वासन प्रदान करण्याच्या सर्वात सामान्य पद्धतींपैकी एक म्हणजे नमूद केलेल्या समस्यांपैकी एखादी समस्या उद्भवल्यास डेटाचा ऑफ-साइट बॅकअप घेणे.

**इनफार्मेशन सुरक्षिततेची बेसिक तत्त्वे**: सीआयए ट्रायड हे सुरक्षा धोरण विकासासाठी एक प्रसिद्ध मॉडेल आहे, ज्याचा वापर समस्या एरिया आणि

आवश्यक उपाय ओळखण्यासाठी केला जातो. इनफार्मेशन सुरक्षिततेसाठी. प्रायव्हसी , अखंडता आणि उपलब्धता (CIA) हे संस्थेतील इनफार्मेशन सुरक्षिततेसाठी धोरणांचे पाथ दर्शन करण्यासाठी डिझाइन केलेले मॉडेल आहे. प्रायव्हसी , अखंडता आणि सबलब्धतेचा CIA त्रिकूट इनफार्मेशन सुरक्षेच्या केंद्रस्थानी आहे. क्लासिक इनफार्मेशन सिक्युरिटी ट्रायडचे सदस्य - प्रायव्हसी , अखंडता आणि उपलब्धता - यांना सुरक्षा प्रॉपर्टीज , प्रॉपर्टीज , सुरक्षा उद्दिष्टे, बेसिक पैलू इनफार्मेशन निकष, गंभीर इनफार्मेशन वैशिष्ट्ये आणि बेसिक बिल्डिंग ब्लॉक्स म्हणून देखील संबोधले जाते.

### गुप्तता

प्रायव्हसी हा नियमांचा एक संच आहे जो इनफार्मेशन वर एक्सेस मर्यादित करतो. प्रायव्हसी मुळे सेन्सिटिव्ह इनफार्मेशन चुकीच्या लोकांपर्यंत पोहोचण्यापासून प्रतिबंधित करते, तसेच योग्य लोकांना ती मिळू शकते याची खात्री करून घेते. प्रायव्हसी चे प्रोटेक्शन करणे हे इनफार्मेशन साठी योग्य एक्सेस स्तर परिभाषित करणे आणि लागू करणे यावर अवलंबून असते. कोणाला त्यात एक्सेस असावा आणि ती किती सेन्सिटिव्ह आहे याद्वारे आयोजित केलेल्या स्वतंत्र युनिटमध्ये इनफार्मेशन विभक्त करून हे केले जाऊ शकते.

### सचोटी

अखंडता ही खात्री आहे की इनफार्मेशन विश्वासार्ह, सातत्यपूर्ण आणि तिच्या संपूर्ण जीवनचक्रात अचूक आहे. याचा अर्थ असा की डेटा अनधिकृत किंवा न सापडलेल्या मेथड्स ने बदलला जाऊ शकत नाही. डेटा ट्रांझिटमध्ये बदलला जाऊ नये आणि अनधिकृत लोकांकडून डेटा बदलला जाऊ शकत नाही याची खात्री करण्यासाठी पावले उचलली पाहिजेत (उदाहरणार्थ, प्रायव्हसी चा भंग झाल्यास). याव्यतिरिक्त, इलेक्ट्रोमॅग्नेटिक पल्स (EMP) किंवा सर्व्हर क्रॅश यांसारख्या गैर-मानवी-उद्भवलेल्या घटनांमुळे डेटामध्ये होणारे कोणतेही बदल सर्चण्यासाठी काही माध्यमे असणे आवश्यक आहे. अनपेक्षित बदल झाल्यास, प्रभावित डेटा त्याच्या योग्य स्थितीत रिस्टोरार करण्यासाठी बॅकअप प्रत सबलब्ध असणे आवश्यक आहे.

## उपलब्धता

उपलब्धता ही अधिकृत लोकांद्वारे इनफार्मेशन साठी तयार प्रवेशाची हमी आहे. कोणत्याही इनफार्मेशन सिस्टिम चा उद्देश पूर्ण करण्यासाठी, जेव्हा आवश्यक असेल तेव्हा इनफार्मेशन सबलब्ध असणे आवश्यक आहे. याचा अर्थ इनफार्मेशन स्टोर करण्यासाठी आणि त्यावर प्रक्रिया करण्यासाठी वापरल्या जाणाऱ्या संगणकीय सिस्टिम , तिचे प्रोटेक्शन करण्यासाठी वापरल्या जाणाऱ्या सुरक्षा कंट्रोल आणि त्यामध्ये एक्सेस करण्यासाठी वापरल्या जाणाऱ्या कॅम्प्युनिकेशन चॅनेल योग्यरित्या कार्यरत असणे आवश्यक आहे.

पॉवर आउटेज, हार्डवेअर अयशस्वी आणि सिस्टिम अपग्रेडमुळे सर्विस व्यत्यय टाळण्यासाठी, उच्च उपलब्धता सिस्टिम नेहमी सबलब्ध राहण्याचे उद्दिष्ट ठेवते. सर्व हार्डवेअरची काटेकोरपणे देखभाल करून, आवश्यकतेनुसार हार्डवेअरची दुरुस्ती करून, रिडंडंसी आणि फेलओव्हरचे काही मोजमाप प्रदान करून, पुरेशी कॅम्प्युनिकेशन बँडविड्थ प्रदान करून आणि बाटलीच्या गळ्याची इवेंट्स रोखून, आणीबाणीच्या बँकअप पॉवर सिस्टिमची अंमलबजावणी करून, सर्व आवश्यक सिस्टीमसह विद्युत चालू ठेवून उपलब्धता सुनिश्चित केली जाते. अपग्रेड, आणि डिनायल-ऑफ-सर्व्हिस (DoS) हल्ल्यांसारख्या दुर्भावनापूर्ण कृतींपासून प्रोटेक्शन .

वर नमूद केलेल्या तीन सदस्यांव्यतिरिक्त, प्रमाणिकता आणि नॉन-रिपिडिएशन हे देखील CIA मॉडेलचे सदस्य मानले जातात.

## सत्यता

प्रमाणिकता ही डेटा, व्यवहार, संप्रेषणे किंवा कागदपत्रे खरी असल्याची खात्री करण्याची प्रक्रिया आहे. यात सहभागी असलेले पक्ष खरे आहेत याची पडताळणी करणेही सत्यतेसाठी महत्त्वाचे आहे. काही इनफार्मेशन सुरक्षा प्रणालींमध्ये प्रमाणीकरण वैशिष्ट्ये समाविष्ट आहेत जसे की "डिजिटल स्वाक्षरी", जे पुरावा देतात की मेसेज केलेला डेटा खरा आहे आणि योग्य स्वाक्षरी की असलेल्या एखाद्याने पाठवला होता.

## नाकारणे

नाकारणे म्हणजे एखाद्या व्यक्तीचा करारातील त्याच्या जबाबदाऱ्या पूर्ण करण्याचा हेतू. हे असेही सूचित करते की ट्रांसक्शन तील एक पक्ष असे म्हणू शकत नाही की त्यांना व्यवहार प्राप्त झाला नाही किंवा दुसरा पक्ष व्यवहार पाठविल्याचे नाकारू शकत नाही.

हे लक्षात घेणे महत्त्वाचे आहे की क्रिप्टोग्राफिक सिस्टीम सारखे तंत्रज्ञान नॉन-रिड्युएशन प्रयत्नांमध्ये मदत करू शकते, ही संकल्पना मुळात कायदेशीर संकल्पना आहे. उदाहरणार्थ, संदेश प्रेषकाच्या प्राइवेट की सह स्वाक्षरी केलेल्या डिजिटल स्वाक्षरीशी जुळतो हे दाखवण्यासाठी पुरेसे नाही, आणि अशा प्रकारे केवळ प्रेषक संदेश पाठवू शकला असता आणि इतर कोणीही ते संक्रमणामध्ये बदलू शकले नसते. कथित प्रेषक त्या बदल्यात डिजिटल स्वाक्षरी अल्गोरिदम असुरक्षित किंवा सदोष असल्याचे प्रदर्शित करू शकतो किंवा त्याच्या स्वाक्षरी की तडजोड केली गेली आहे असा आरोप करू शकतो किंवा सिद्ध करू शकतो. या उल्लंघनांची चूक प्रेषकाची स्वतःची असू शकते किंवा नसू शकते आणि अशा दाव्यामुळे प्रेषकाच्या दायित्वापासून ओपन ता होऊ शकते किंवा नसू शकते, परंतु प्रतिपादन हा

दावा अमान्य करेल की स्वाक्षरी अनिवार्यपणे सत्यता आणि सचोटी सिद्ध करते आणि अशा प्रकारे खंडन टाळते.

आम्हाला जवळजवळ अमर्याद स्वातंत्र्य देणाऱ्या सर्व क्रियाकलापांसह, रिस्क आहेत. इंटरनेट कोणालाही सहज सबलब्ध असल्यामुळे ते धोकादायक ठिकाण असू शकते. तुम्ही कोणाशी व्यवहार करत आहात किंवा तुम्ही कशात आहात हे जाणून घ्या. शिकारी, सायबर गुन्हेगार, गुंड आणि भ्रष्ट व्यवसाय अविचारी पाहुण्यांचा फायदा घेण्याचा प्रयत्न करतील.

## इंटरनेट, इंटरनेट आणि सुरक्षा

जरी इंटरनेट आणि इंटरनेटमधील फरक तंत्रज्ञानाच्या दृष्टीने फारसा नसला तरी, इनफार्मेशन चे प्रसारण संस्थात्मक दृष्टिकोनातून पूर्णपणे भिन्न आहे.

इंटरनेट आणि इतर नेटवर्क आणि इनफार्मेशन सिस्टिम यांच्यातील इनफार्मेशन सुरक्षा धोके ऐवजी समान आहेत. इंटरनेट इनफार्मेशन चॅनेल म्हणून इंटरनेटचा वापर इनफार्मेशन सुरक्षिततेच्या महत्त्वावर भर देतो. इंटरनेट इंटरनेटवर ठेवलेल्या मालमत्तेमुळे संभाव्य गैरवापरकर्त्यांचे स्वारस्य वाढू शकते. त्यामुळे, गुप्तता, अखंडता आणि इनफार्मेशन ची उपलब्धता धोक्यात आणणाऱ्या विविध धोक्यांपासून इंटरनेट आणि त्यांच्याद्वारे प्रसारित होणारा डेटा आणि इनफार्मेशन यांचे प्रोटेक्शन करणे हे अत्यंत महत्त्वाचे आहे विचार.

इंटरनेटचा एक पार्ट म्हणून इंटरनेट वापरणे गंभीर धोका निर्माण करते, कारण इंटरनेट मूळतः असुरक्षित आहे. परिणामी, वापरकर्त्यांनी स्पेशल तः सावधगिरी बाळगली पाहिजे त्यांचे कॅम्प्युनिकेशन एन्क्रिप्ट करताना. अनुकरण (स्पूफिंग), प्रत्युत्तर (रॅपिड फायर), संदेश कॅटेंट मध्ये बदल (सुपरझॉपिंग), सर्विस उपलब्धता प्रतिबंध आणि ऍक्टिव्ह आणि निष्क्रिय वायरटॅपिंग हे सर्वात दुर्भावनापूर्ण धोके आहेत. वायरटॅपिंग, उदाहरणार्थ, मजबूत एन्क्रिप्शन पद्धतींद्वारे कॅम्प्युनिकेशन एन्क्रिप्शनची अंमलबजावणी न केल्यास, एखाद्या संस्थेशी संबंधित धोरणात्मक ज्ञान बाहेरील लोकांच्या हातात जाईल अशी परिस्थिती उद्भवू शकते.

हॅकर साधने, जरी इंटरनेटसाठी विकसित केली असली तरी, इंटरनेटवर देखील वापरण्यायोग्य आहेत. ते सॉफ्टवेअर किंवा हार्डवेअर आधारित किंवा दोन्हीचे संयोजन असू शकतात. त्यांच्या अधिकृत वापरामध्ये इंटरनेटवरील इनफार्मेशन सुरक्षा कमकुवतपणा सर्च णे आणि दुरुस्त करणे समाविष्ट आहे. तथापि, ते अशा कॅम्प्युनिकेशन प्रणालींना हॅक करण्यास आणि त्यांना एक्सेस करण्यास अधिकृत नसलेल्या इनफार्मेशन मध्ये एक्सेस करण्यास देखील सक्षम करतात.

## इनफार्मेशन सुरक्षा आणि सायबर सुरक्षा यांच्यातील संबंध

इनफार्मेशन सुरक्षा, पूर्वीच्या विभागांमध्ये नमूद केलेली इनफार्मेशन आणि इनफार्मेशन सिस्टिम चे अनधिकृत एक्सेस , वापर, प्रकटीकरण, व्यत्यय, बदल किंवा विनाश यापासून प्रायव्हंसी , अखंडता आणि उपलब्धता प्रदान करण्यासाठी प्रोटेक्शन आहे.

दुसरीकडे सायबरसुरक्षा ही सायबर हल्ल्यांपासून सायबरस्पेसच्या वापराचे प्रोटेक्शन किंवा प्रोटेक्शन करण्याची क्षमता म्हणून परिभाषित केली जाऊ शकते. सायबरस्पेस म्हणजे "ज्या वातावरणात संगणक नेटवर्कवर कॅम्प्युनिकेशन होते."



सायबर सुरक्षेमध्ये सायबर डोमेन किंवा क्षेत्रामध्ये (किंवा सायबरस्पेस) सुरक्षा-संबंधित कोणतीही गोष्ट समाविष्ट असते. इनफार्मेशन सुरक्षेमध्ये इनफार्मेशन किंवा इनफार्मेशन सिस्टिम ची सुरक्षितता समाविष्ट आहे ती कोणत्याही क्षेत्रात उद्भवते (उदा., फिजिकल जगामध्ये उघड होण्याचा धोका). सायबर क्षेत्रात घडणाऱ्या कोणत्याही गोष्टीमध्ये इनफार्मेशन आणि इनफार्मेशन प्रणालींचे प्रोटेक्शन समाविष्ट असते, त्यामुळे तुम्ही असा निष्कर्ष काढू शकता की इनफार्मेशन सुरक्षा हा सायबर सुरक्षेचा एक सुपर-सेट आहे. (पिक्चर 1) काहीवेळा दोन संज्ञा परस्पर बदलून वापरल्या जातात.

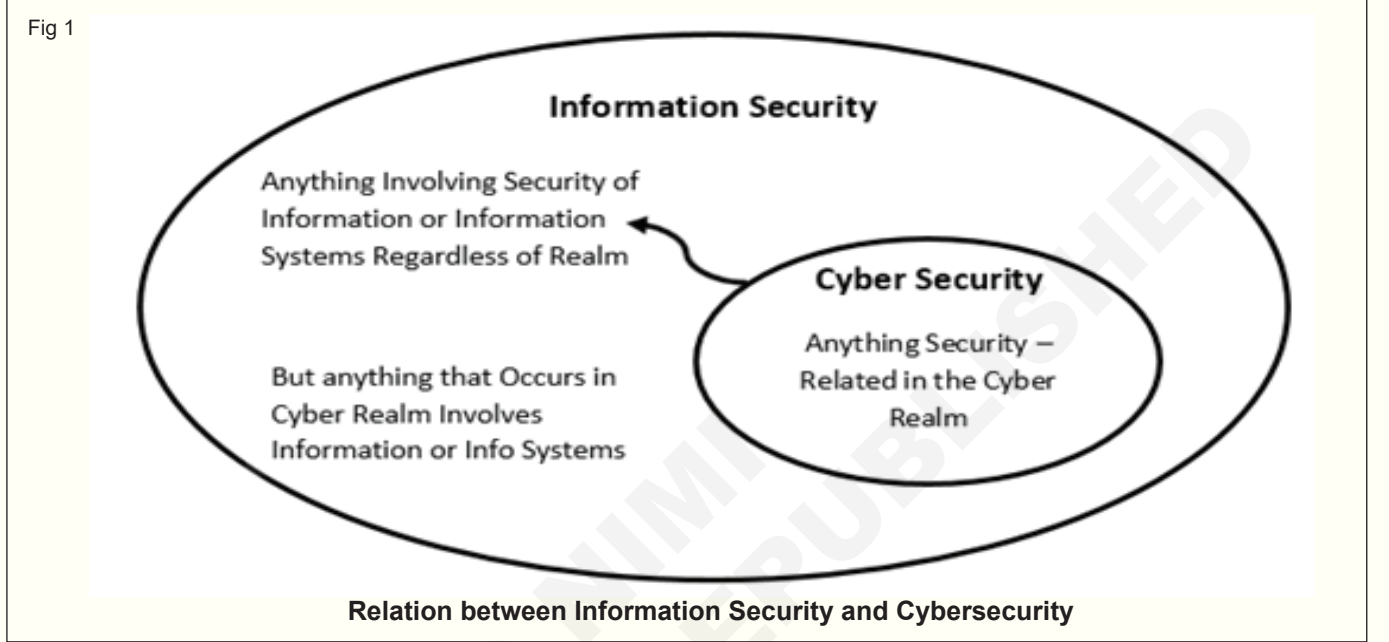
## इनफार्मेशन सुरक्षा मधील प्रमुख आव्हाने

### 1 आयटी सुरक्षिततेला कमी प्राधान्य दिले जाते

- संस्थेने IT सुरक्षा मेथड्स लागू करण्यावर योग्य लक्ष केंद्रित केलेले नाही.

### 2 Ad hoc सुरक्षा एडमिनिस्ट्रेशन

- इनफार्मेशन सुरक्षा मॅनेजमेन्ट सिस्टिम (ISMS) किंवा संरचित शासन यंत्रणेची अनुपस्थिती. (आकृती क्रं 1)



### 3 भूमिका आणि जबाबदाऱ्यांमध्ये अस्पष्टता

- SSO मधील वेगवेगळ्या खेळाडूंच्या (व्यवसाय, SSO मधील संघ, इ.) भूमिका आणि जबाबदाऱ्यांवर संदिग्धता आहेत. सिंगल साइन-ऑन (SSO) ही ऍक्सेस कंट्रोलची प्रॉपर्टी आहे मल्टिपल संबंधित, परंतु स्वतंत्र सॉफ्टवेअर सिस्टम. या मालमतेसह युजर एकदाच लॉग इन करतो आणि प्रत्येक सिस्टममध्ये पुन्हा लॉग इन करण्यास प्रॉम्प्ट न करता सर्व सिस्टममध्ये ऍक्सेस मिळवतो

### 4 कर्तव्यांचे अपुरे पृथक्करण

- SSO मध्ये ओव्हरलॅपिंग आणि सामायिक जबाबदाऱ्या कर्तव्यांमध्ये योग्य लेयर चे पार्टिशन लागू करणे कठीण करते.

### 5 सुरक्षा आवश्यकतांची विविध व्याख्या

- स्टॅण्डर्ड व्याख्येच्या अनुपस्थितीत, वेगवेगळ्या व्यक्ती आणि संघांचे स्वतःचे स्पष्टीकरण असते.

### 6 रिस्क पातळी कमी करण्याची प्रवृत्ती

- संघ अड्मिनिस्ट्रेटिव्ह प्रक्रियेच्या कठोरतेला बायपास करण्यासाठी 'रिस्क पातळी' कमी करण्याची प्रवृत्ती दर्शवितात.

### 7 मल्टिपल विक्रेते

- अथक स्पर्धा आणि असुरक्षिततेच्या भावनेमुळे विक्रेत्यांमध्ये जबाबदारीची वाटणी करण्याची अनिच्छा आणि विक्रेत्यांमध्ये कमी किंवा कोणतेही सहकार्य नाही.

### 8 व्यवसाय/ऑपरेशन्स अनेक भौगोलिक क्षेत्रांमध्ये पसरलेले आहेत

- संस्था जगभरात पसरलेल्या अनेक ठिकाणांवर आधारित आहे आणि कार्य करते.

### 9 प्रशिक्षणाचा/जागरूकतेचा अभाव

- सुरक्षा पद्धतींबाबत अपुरे प्रशिक्षण आणि जागरूकता.

### इनफार्मेशन सुरक्षिततेचे फायदे:

- विविध सायबर गुन्हांच्या एक्सटेर्नल धोक्यांपासून होणारा रिझल्ट कमी करण्यासाठी अनधिकृत प्रवेशापासून नेटवर्क, संगणक आणि डेटाचे प्रोटेक्शन करा
- तांत्रिक, मॅनेजमेन्ट, अड्मिनिस्ट्रेटिव्ह आणि परिचालन कंट्रोल लागू करण्यासाठी सुधारित इनफार्मेशन सुरक्षा आणि व्यवसाय सातत्य मॅनेजमेन्ट, जो धोका कमी करण्याचा सर्वात किफायतशीर पाथ आहे.
- इनफार्मेशन सुरक्षा व्यवस्थेमध्ये सुधारित पार्ट धारकांचा आत्मविश्वास.

- योग्य सुरक्षा नियंत्रणांसह सुधारित कंपनी क्रेडेन्शियल्स संस्था इंटरनल पार्ट धारक आणि एक्सटर्नल विक्रेत्यांमध्ये विश्वासाहता आणि विश्वास सुधारेल. विश्वासाहता आणि विश्वास हे व्यवसाय जिंकण्यासाठी महत्त्वाचे एलिमेंट आहेत.
- व्यत्यय आल्यास जलद पुनर्प्राप्ती वेळा

## संस्थेमध्ये IS लागू करण्याचे तंत्र

### इनफार्मेशन सुरक्षा लागू करण्यासाठी साधने ओळखणे

एक यशस्वी इनफार्मेशन सुरक्षा धोरण कॉर्पोरेशनला अनेक फायदे प्रदान करते. अंमलबजावणी करण्यायोग्य धोरणे हे सुनिश्चित करतात की असुरक्षा ओळखल्या जातात आणि त्यांचे निराकरण केले जाते. याचा रिझल्ट व्यवसायातील सातत्य राखण्यात आणि आयटी पायाभूत सुविधा मजबूत करण्यात होतो. जेव्हा संपूर्ण संस्थेतील कर्मचारी सुरक्षा धोरणाचे पालन करतात, याची खात्री करून संस्थेमध्ये तसेच ग्राहक, भागीदार आणि विक्रेत्यांसह इनफार्मेशन सुरक्षितपणे सामायिक केली जाते, तेव्हा धोका कमी होतो.

### 1 प्रभावी इनफार्मेशन सुरक्षा धोरण तयार करण्याची पहिली स्टेप म्हणजे इनफार्मेशन मालमत्तेचे असेसमेंट करणे आणि त्या मालमत्तेला धोका ओळखणे.

संस्थेतील काही प्रॉपर्टी इतरांपेक्षा अधिक मौल्यवान असतील, परंतु आर्थिक व्हॅल्यू हा एकमेव एलिमेंट असू नये. मालमत्तेचे मौद्रिक व्हॅल्यू आणि आंतरिक व्हॅल्यू दोन्ही निश्चित करणे तिची किंमत अचूकपणे मोजण्यासाठी आवश्यक आहे. मालमत्तेच्या आर्थिक मूल्याची गणना करण्यासाठी, एखाद्या संस्थेने मालमत्तेचा डेटा, नेटवर्क किंवा सिस्टमशी कोणत्याही प्रकारे तडजोड केली असल्यास त्याच्या प्रभावाचा विचार केला पाहिजे. आंतरिक मूल्याची गणना करण्यासाठी, एखाद्या संस्थेने विश्वासाहता, प्रतिष्ठा आणि मुख्य पार्ट धारकांसोबतच्या संबंधांवर सुरक्षिततेच्या घटनेचा इफेक्ट विचारात घेणे आवश्यक आहे.

### 2 धोरण तयार करणे हे संस्थांना जोखमीचे असेसमेंट करण्यासाठी आहे

मालमत्तेची आणि धमक्यांची ओळख झाल्यानंतर, धोरण तयार करण्याची पुढील स्टेप म्हणजे संस्थांनी जोखमीचे असेसमेंट करणे. हे असेसमेंट संस्थेला इनफार्मेशन संरक्षित, अतिसंरक्षित किंवा पुरेसे संरक्षित आहे की नाही हे ठरवू देते. या रिस्क व्हॅल्यू कनाचे उद्दिष्ट एखाद्या संस्थेला अनावश्यक जोखमीच्या समोर न आणता खर्च कमी

करणे हे असले पाहिजे. सुरक्षा धोरण प्रभावीपणे अंमलात आल्यावर हे असेसमेंट संसाधनांचे योग्य वाटप निश्चित करण्यात मदत करेल.

### इनफार्मेशन सुरक्षा फ्रेमवर्क

इनफार्मेशन सुरक्षा फ्रेमवर्क संस्थेसाठी सुरक्षा धोरण आणि मेथड्स इंस्टाल करते. धोरणे सुरक्षेवर रिझल्ट करणाऱ्या बाबींवर पाथ दर्शन देतात ज्यांचे पालन संस्थेच्या सदस्यांनी करणे अपेक्षित आहे. सुरक्षा धोरण सर्व हार्डवेअर, सॉफ्टवेअर, डेटा, इनफार्मेशन, नेटवर्क, वैयक्तिक संगणकीय उपकरणे, सपोर्ट कर्मचारी आणि संस्थेतील वापरकर्त्यांना लागू होते.

IT सुरक्षा सिस्टिम कार्य करण्यासाठी, सर्व पार्ट धारकांना सामील करून एक सु-परिभाषित फ्रेमवर्क विकसित करणे आवश्यक आहे आणि ते सबयुक्त होण्यासाठी वेळोवेळी अपडेट करणे आवश्यक आहे. इनफार्मेशन सुरक्षा फ्रेमवर्क डेटा हाताळणे आणि मालमत्तेची ओळख, सुरक्षा आवश्यकता निश्चित करणे, रिस्क असेसमेंट, कंट्रोल व्हॅल्यू मापन, कंट्रोल अंमलबजावणी, प्रक्रिया देखरेख आणि अद्यतनाच्या स्वरूपात सिस्टिम /प्रक्रियेच्या अंमलबजावणीचा विचार करण्यासाठी मॅनेजमेन्ट प्रक्रियेस सुलभ करते.

इनफार्मेशन सुरक्षा फ्रेमवर्कमध्ये सामान्यतः वापरल्या जाणाऱ्या संज्ञा आहेत:

- धोरण: सामान्य मॅनेजमेन्ट स्टेटमेंट
- मानके: स्पेसिफिक अनिवार्य कंट्रोल
- पाथ दर्शक तत्त्वे: शिफारसी / सर्वोत्तम मेथड्स
- प्रक्रिया: स्टेप बाय स्टेप सूचना

सुरक्षा फ्रेमवर्क प्रत्यक्षात अंमलात आणण्यासाठी सविस्तर क्रियाकलाप, अनुसरण्याचा सराव, करावयाच्या सुधारात्मक कृती खालील आकृतीमध्ये दर्शविल्या आहेत. (पिक्चर 2).

फ्रेमवर्कमधील प्रत्येक हेड इंटरनल प्रमुख हेड आणि सराव खाली दर्शविल्याप्रमाणे आहेत.

अशा प्रकारे सुरक्षा फ्रेमवर्क हे तंत्रज्ञान बनते जे सुरक्षा धोरणांना ट्रांसक्शन त बदलते. योजना, कृती करा आणि तपासा या चार स्टेप्स च्या चक्राद्वारे ते साध्य होते. द पीपीटी ट्रायड, म्हणजे. हे साध्य करण्यासाठी लोक, प्रक्रिया आणि तंत्रज्ञान यांना समान महत्त्व देणे आवश्यक आहे.

नवीन तंत्रज्ञान आणि नवीन नेटवर्क सुरक्षा फ्रेमवर्कमध्ये जोडू शकतात आणि सुरक्षा खर्च अधिक अंदाजे आणि मॅनेज करण्यायोग्य बनतात.

Fig 2



ओळखा	प्रोटेक्शन करा	शोधा	प्रतिसाद द्या	पुनर्प्राप्त करा
<ul style="list-style-type: none"> <li>मालमत्ता व्यवस्थापन</li> <li>व्यवसाय वातावरण</li> <li>शासन</li> <li>जोखीम</li> <li>जोखीम व्यवस्थापन धोरण</li> </ul>	<ul style="list-style-type: none"> <li>प्रवेश नियंत्रण कार्यक्रम</li> <li>जागरूकता आणि प्रशिक्षण</li> <li>डेटा सुरक्षा</li> <li>माहिती संरक्षण आणि प्रक्रिया</li> <li>देखभाल मूल्यांकन</li> <li>संरक्षणात्मक तंत्रज्ञान</li> </ul>	<ul style="list-style-type: none"> <li>विसंगती आणि</li> <li>सुरक्षा सतत देखरेख</li> <li>शोध प्रक्रिया</li> <li>शमविणे</li> <li>सुधारणा</li> </ul>	<ul style="list-style-type: none"> <li>प्रतिसाद नियोजन</li> <li>संप्रेषण</li> <li>विश्लेषण</li> </ul>	<ul style="list-style-type: none"> <li>पुनर्प्राप्ती नियोजन</li> <li>सुधारणा</li> <li>संप्रेषण</li> </ul>

# सुरक्षा धोक्यांचे ओवरविव्ह (Overview of security threats)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- सुरक्षा धोक्यांचे आणि त्याचे टाईप डिस्क्रिप्शन करा
- धोके ओळखण्याच्या पद्धतींचे डिस्क्रिप्शन करा
- धमक्यांचा सिस्टिम वर कसा रिझल्ट होतो ते स्पष्ट करा
- धमक्यांच्या स्त्रोतांचे डिस्क्रिप्शन करा
- धोके ओळखण्यासाठी आणि कमी करण्यासाठी सर्वोत्तम पद्धतींचे डिस्क्रिप्शन करा.

## परिचय

धोका ही कोणतीही परिस्थिती किंवा इवेंट्स आहे ज्यामध्ये सिस्टिम किंवा क्रियाकलाप नष्ट करणे, प्रकटीकरण करणे आणि डेटामध्ये बदल करणे किंवा सर्विस नाकारणे या स्वरूपात हानी पोहोचवण्याची क्षमता आहे. धोका म्हणजे हानी होण्याची शक्यता. धोक्याच्या सबस्थितीचा अर्थ असा नाही की तो अपरिहार्यपणे वास्तविक हानी करेल.

धमक्यांशी संबंधित काही सामान्य संज्ञा आणि त्यांचे डिस्क्रिप्शन खालीलप्रमाणे आहे:

### 1 अनधिकृत एक्सेस

परवानगी किंवा अधिकारांशिवाय इनफार्मेशन किंवा प्रणालींचा प्रयत्न किंवा यशस्वी एक्सेस.

### 2 सायबर हेरगिरी

सेन्सिटिव्ह इनफार्मेशन मिळविण्यासाठी संगणकाच्या वापराद्वारे हेरगिरीची कृती, ज्यामध्ये गुप्त एक्सेस किंवा कंपनी किंवा सरकारी नेटवर्कचे 'हॅकिंग' समाविष्ट आहे.

### 3 मालवेअर

दुर्भावनापूर्ण सॉफ्टवेअरसाठी सामूहिक संज्ञा, जसे की व्हायरस, वर्म्स आणि ट्रोजन; गुन्हेगारी, व्यावसायिक किंवा विध्वंसक हेतूसाठी सिस्टिम आणि इनफार्मेशन मध्ये घुसखोरी करण्यासाठी डिझाइन केलेले.

### 4 डेटा लीकेज

सेन्सिटिव्ह कंपनी किंवा वैयक्तिक इनफार्मेशन जाणूनबुजून किंवा अपघाती नुकसान, चोरी किंवा उघड.

### 5 मोबाइल डिव्हाइस हल्ला

मोबाइल डिव्हाइसेस आणि त्यांच्याद्वारे स्टोर किंवा प्रक्रिया केलेल्या इनफार्मेशन वर दुर्भावनापूर्ण हल्ला किंवा अनधिकृत एक्सेस; बिनतारी किंवा फिजिकल ताब्याद्वारे केले जाते.

### 6 सोशल इंजिनीरिंग

कंपनीची इनफार्मेशन किंवा प्रणालींमध्ये एक्सेस करण्यासाठी, सेन्सिटिव्ह इनफार्मेशन उघड करण्यासाठी फोन, ईमेल, ऑनलाइन किंवा वैयक्तिकरित्या इतरांना फसवणे आणि हाताळणे.

### 7 इनसाइडर

सेन्सिटिव्ह कंपनीची इनफार्मेशन चोरण्याचा, फसवणूक करण्याचा

किंवा कंपनीच्या सिस्टिम किंवा इनफार्मेशन चे नुकसान करण्याचा मालिशियस हेतू असलेला कर्मचारी किंवा कर्मचारी.

### 8 फिशिंग

सोशल इंजिनीरिंग चा एक टाईप, ज्यामध्ये प्राप्तकर्त्याकडून फसवणूक करून सेन्सिटिव्ह इनफार्मेशन काढणे, सामान्यतः सिस्टिममध्ये एक्सेस मिळवणे किंवा ओळख चोरी करणे या उद्देशाने कायदेशीर दिसणारे ईमेल पाठवणे समाविष्ट आहे.

### 9 सिस्टिम तडजोड

दुर्भावनापूर्ण व्यक्ती किंवा 'हॅकर्स' द्वारे आक्रमण करून ताब्यात घेतलेली सिस्टिम, सामान्यतः एक किंवा अधिक असुरक्षिततेच्या शोषणाद्वारे, आणि नंतर अनेकदा इतर प्रणालींवर हल्ला करण्यासाठी वापरली जाते.

### 9 स्पॅम

अनेक व्यक्तींना मोठ्या प्रमाणात पाठवलेले नको त्या ईमेल, सामान्यतः व्यावसायिक फायद्यासाठी, परंतु मालवेअर पसरवण्यासाठी वाढत्या प्रमाणात.

### 10 सर्विस नाकारणे

सिस्टीमवर हेतुपुरस्सर किंवा अनावधानाने केलेला हल्ला आणि त्यावर स्टोर केलेली इनफार्मेशन, सिस्टिम अनुपलब्ध आणि अधिकृत वापरकर्त्यांसाठी एक्सेस करण्यायोग्य नाही.

### 11 ओळख चोरी

एखाद्या अनोळखी व्यक्तीच्या वैयक्तिक इनफार्मेशन ची चोरी, एखाद्या व्यक्तीची ओळख फसवणूक करून गुन्हा करण्यासाठी, सहसा आर्थिक फायद्यासाठी.

## धमक्यांच्या प्रकार

सुरक्षा धोक्यांचे अनेक प्रकारे वर्गीकरण केले जाऊ शकते. काही लोकप्रिय वर्गीकरणे खालीलप्रमाणे आहेत:

1 अत्याधुनिकतेच्या आधारावर, सुरक्षा धोक्यांना तीन रेन्ज मध्ये वर्गीकृत केले जाऊ शकते.

- साध्या पहिल्या पिढीतील धमक्या हे जेनेरिक व्हायरस-टाईप चे हल्ले आहेत जे वापरकर्ते संक्रमित ई-मेल आणि अस्पष्ट फाइल संलग्नक उघडतात.

- दुसऱ्या पिढीतील धमक्या अधिक अत्याधुनिक आहेत आणि मोठ्या समस्या निर्माण करतात. ऑटोमॅटिक साधनांसह तयार केलेले, हे जंत मानवी संवादाशिवाय असुरक्षिततेवर हल्ला करतात. नवीन बळींची प्रतिकृती, ओळख आणि लक्ष्यीकरण ऑटोमॅटिक आहे.
  - तिसऱ्या पिढीतील धमक्या मिश्रित धमक्या आहेत, सामान्य आहेत आणि व्हायरस, ट्रोजन आणि ऑटोमेशन समाविष्ट करतात. हे वर्म्स हायपर प्रसारासाठी लक्ष्ये पूर्व-संकलित करतात, ज्ञात असुरक्षा शोषण करतात आणि लपलेल्या असुरक्षा लक्षित वापर सक्षम करतात. ते वायरलेस लिंक्स, व्हर्च्युअल प्रायव्हेट नेटवर्क्स आणि फायरवॉल आणि घुसखोरी सर्च सिस्टिम सारख्या परिमितीच्या आतल्या संरक्षणास देखील टारगेट करतात.
- 2 OWASP (ओपन वेब ऍप्लिकेशन्स सिक्युरिटी प्रोजेक्ट) नुसार सर्वात वरचे धोके खालीलप्रमाणे आहेत:
- इंजेक्शन
  - क्रॉस साइट स्क्रिप्टिंग (CSS)
  - तुटलेली प्रमाणीकरण आणि सत्र मॅनेजमेंट
  - असुरक्षित थेट ऑब्जेक्ट रेफरेन्स
  - क्रॉस - साइट रिक्वेस्ट फोर्जरी (CSRF)
  - सुरक्षा चुकीची कॉन्फिगरेशन
  - असुरक्षित क्रिप्टोग्राफिक स्टोरेज
  - URL एक्सेस प्रतिबंधित करण्यात अयशस्वी
  - अपुरे वाहतूक स्तर प्रोटेक्शन
  - अप्रमाणित पुनर्निर्देशन आणि अग्रेषित
- 3 वापरल्या जाणाऱ्या शोषणाच्या प्रकारांनुसार मायक्रोसॉफ्टचे वर्गीकरण (किंवा आक्रमणकर्त्यांची प्रेरणा) म्हणजे. STRIDE सिस्टिम . STRIDE संक्षिप्त रूप खालीलपैकी प्रत्येक रेन्ज च्या पहिल्या अक्षरापासून तयार केले आहे.
- स्पूफिंग ओळख. ओळख स्पूफिंगचे उदाहरण म्हणजे बेकायदेशीरपणे एक्सेस करणे आणि नंतर दुसऱ्या यूजर ची प्रमाणीकरण इनफार्मेशन वापरणे, जसे की युजर नाव आणि संकेतशब्द.
  - डेटाशी छेडछाड. डेटा टॅम्परिंगमध्ये डेटाचे दुर्भावनापूर्ण बदल समाविष्ट आहेत.
  - खंडन. नाकारण्याच्या धमक्या अशा वापरकर्त्यांशी संबंधित आहेत जे इतर पक्षांना अन्यथा सिद्ध करण्याचा कोणताही पाथ नसताना कृती करण्यास नकार देतात - उदाहरणार्थ, युजर अशा सिस्टिममध्ये बेकायदेशीर ऑपरेशन करतो ज्यामध्ये प्रतिबंधित ऑपरेशन्स ट्रेस करण्याची क्षमता नसते.
  - इनफार्मेशन प्रकटीकरण. इनफार्मेशन उघड करण्याच्या धमक्यांमध्ये इनफार्मेशन उघड करणे समाविष्ट आहे ज्यांना त्यात एक्सेस नसावा.

- सर्विस नाकारणे. सर्विस नाकारणे (DoS) हल्ले वैध वापरकर्त्यांना सर्विस नाकारतात.
- विशेषाधिकाराची उन्नती. या टाइप च्या धोक्यात, विशेषाधिकार नसलेल्या यूजर ला विशेषाधिकार प्राप्त होतो आणि त्याद्वारे संपूर्ण सिस्टिम शी तडजोड किंवा नष्ट करण्यासाठी पुरेसा एक्सेस असतो.

### तंत्रज्ञानावर आधारित धमक्या

- 1 WWW तंत्रज्ञानावर आधारित धमक्या
- 2 ब्राउझर सॉफ्टवेअरमध्ये नवीन वैशिष्ट्ये
- 3 ब्राउझर सॉफ्टवेअर टेस्टिंग आवृत्त्या
- 4 सर्व्हर सॉफ्टवेअर
- 5 CGI स्क्रिप्ट
- 6 कुकीज
- 7 युनिक्स आणि TCP/IP साधनांवर आधारित 7 धमक्या
- 8 फायरवॉल मॅनेजमेंट त अडचणी
- 9 क्रिप्टोग्राफिक सॉफ्टवेअरचा वापर
- 10 हॅकर साधने
- 11 इतर सॉफ्टवेअर आधारित धमक्या
- 12 इंटरनेट ऍप्लिकेशन सॉफ्टवेअर
- 13 जावा भाषा
- 14 ActiveX
- 15 कम्युनिकेशन वर आधारित धमक्या
- 16 व्हायरसवर आधारित धमक्या
- 17 मानवी क्रियाकलापांवर आधारित धमक्या

### इनफार्मेशन सुरक्षा धोक्यांची ओळख

इनफार्मेशन सुरक्षा मॅनेजमेंट कार्यक्रमाचे यश संस्थेच्या इनफार्मेशन सिस्टिम ला असलेल्या धोक्यांच्या अचूक ओळखीवर आधारित आहे. इनफार्मेशन सुरक्षा धोक्यांची ओळख सुरक्षा नियोजकांसाठी एक आवश्यक पहिली स्टेप आहे. योग्य धोका आणि असुरक्षितता ओळखण्यात सुरक्षा टेस्टिंग आणि तपासणी यांचा समावेश असावा, जे उपकरणे योग्यरित्या कार्य करत आहेत, आवश्यकतेनुसार सहज सबलब्ध आहेत आणि कर्मचारी इन्फार्मेशन च्या वापरामध्ये निपुण आहेत याची खात्री करण्यासाठी आणि प्रोत्साहन देण्यासाठी सज्ज आहेत.

हे पूर्ण करण्यासाठी, सिस्टमने एक टेस्टिंग प्रोग्रॅम डिझाईन केला पाहिजे जो केवळ सुरक्षिततेच्या सध्याच्या स्थितीचे मुल्यांकन करत नाही तर प्रशिक्षणाद्वारे कर्मचार्यांची प्रभावीता सुधारण्यासाठी देखील वापरला जाऊ शकतो.

सुरक्षा धोके ओळखण्याच्या दोन प्रमुख मेथड्स आहेत **तपासणी आणि स्कॅनिंग**.

प्रोबिंग म्हणजे संगणक सिस्टिम तील ज्ञात किंवा संधाव्य कमकुवत पॉइंट द्वारे संगणक आणि त्याच्या फाइल्समध्ये एक्सेस मिळवण्याचा प्रयत्न. ही नेटवर्कच्या स्थितीबद्दल काही शिकण्याच्या उद्देशाने केलेली कृती आहे.

स्कॅनिंग ही सर्व फायली किंवा नेटवर्क एलिमेंटमधून काहीतरी असामान्य सर्चण्याच्या उद्देशाने जाण्याची एक मेथड्स आहे. फाइल स्कॅनिंग वापरकर्ते व्हायरस आणि इतर दुर्भावनापूर्ण कॅटॅगरी साठी दूरस्थपणे डाउनलोड करण्याचा किंवा उघडण्याचा प्रयत्न करणाऱ्या फाइल्सची तपासणी करते. फाइल स्कॅनिंग पॉलिसी अंमलबजावणीसाठी काही इनफार्मेशन देते.

फाइल स्कॅनिंगचे 2 टाइप आहेत. ते एकत्र वापरले जाऊ शकतात.

- एडवॉन्स ओळख व्हायरस, ट्रोजन हॉर्स, वर्म्स आणि इतरांसह ज्ञात आणि उदयोन्मुख धोके सर्चण्यासाठी तंत्र लागू करते.
- अँटी-व्हायरस स्कॅनिंग व्हायरस-संक्रमित फाइल्स ओळखण्यासाठी अँटी-व्हायरस परिभाषा फाइल्स वापरते.

नेटवर्क स्कॅनिंग ही नेटवर्क सुरक्षा व्हॅल्यु कनाच्या उद्देशाने नेटवर्कवरील ऍक्टिव्ह होस्ट ओळखण्याची एक प्रक्रिया आहे. स्कॅनिंग प्रक्रिया, जसे की पिंग स्वीप आणि पोर्ट स्कॅन, इंटरनेटवर ऍक्टिव्ह असलेल्या लाइव्ह होस्ट्सना कोणते IP एड्रेस मॅप करतात याबद्दलची इनफार्मेशन परत करा आणि कोणतीही संशयास्पद क्रियाकलाप इ.

### सिस्टिम वर हल्ला करण्याच्या धमक्यांची मोडस ऑपरेंडी

सामान्यतः डिव्हाइसवर हल्ला 3 टप्प्यात होतो:

- यजमानाचा संसर्ग,
- त्याच्या ध्येयाची पूर्तता, आणि
- मालवेअरचा इतर प्रणालींमध्ये प्रसार.

धमक्या अनेकदा संक्रमित उपकरणांद्वारे ऑफर केलेल्या संसाधनांचा वापर करतात. ते ब्लूटूथ किंवा इन्फ्रारेड सारख्या आउटपुट डिव्हाइसेसचा वापर

करतात, परंतु यूजर च्या ओळखीच्या लोकांना संक्रमित करण्यासाठी अँड्रेस बुक किंवा त्या व्यक्तीचा ईमेल एड्रेस देखील वापरू शकतात. ओळखीच्या व्यक्तीने पाठवलेल्या डेटावर दिलेल्या विश्वासाचा ते गैरफायदा घेतात.

### इन्फेक्शन

इन्फेक्शन हे साधन मध्ये येण्यासाठी धोक्याचा वापर केला जातो. तो एकतर आधी सादर केलेल्या दोषांपैकी एक वापरू शकतो किंवा यूजर च्या चुकीचा वापर करू शकतो. संसर्ग परवानगी मागू शकतात, किंवा भोळसट यूजर शी संवाद साधू शकतात किंवा या दोनपैकी काहीही करू शकत नाहीत आणि थेट सिस्टिमवर हल्ला करू शकतात.

### ध्येयाची पूर्तता

एकदा धोक्याने एखाद्या डिव्हाइसला संक्रमित केले की ते त्याचे उद्दिष्ट पूर्ण करण्याचा प्रयत्न करेल, जे सहसा खालीलपैकी एक असते: हार्डवेअरचे नुकसान, सेवेला नकार (DoS), आर्थिक नुकसान, नुकसान डेटा आणि/किंवा, डिव्हाइस आणि लपविलेले नुकसान इ.

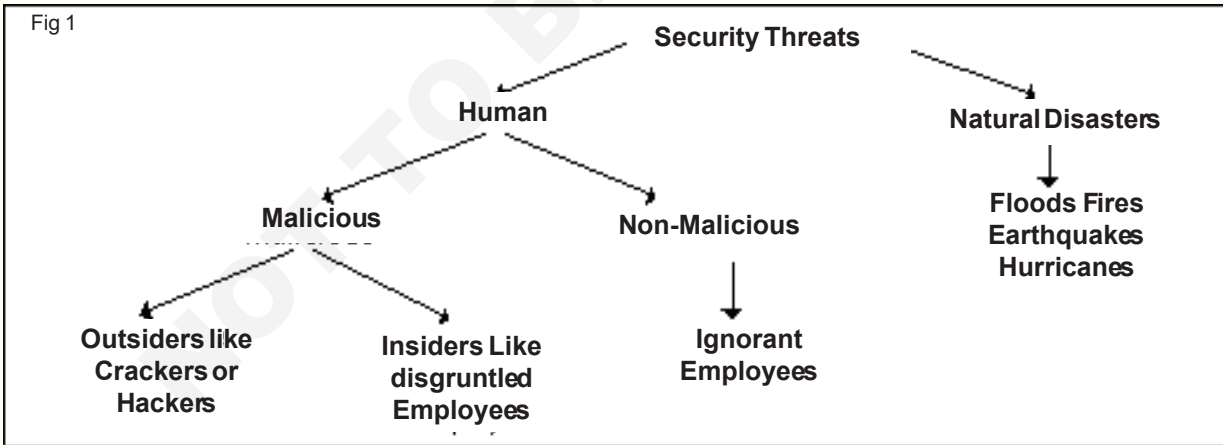
### इतर प्रणालींमध्ये पसरणे

एकदा धोक्याने एखाद्या इन्फेक्शन ला संक्रमित केले की, ते नेहमी एक किंवा दुसऱ्या मार्गाने पसरण्याचे उद्दिष्ट ठेवते.

हे नेटवर्क, वायर्ड किंवा वायरलेस, इंटरनेट, वाय-फाय, ब्लूटूथ आणि इन्फ्रारेड लाइट वापरून जवळच्या उपकरणांद्वारे आणि सामायिक उपकरणांद्वारे पसरू शकते.

### धोक्याचे सोर्स (पिव्चर 1)

धोक्यांचे प्रायमरी सोर्स कर्मचारी/आतील व्यक्ती, दुर्भावनापूर्ण हॅकर्स, नैसर्गिक आपत्ती, फॉरेन विरोधक आणि प्रतिकूल हल्ले आहेत. बऱ्याच प्रकरणांमध्ये, धमक्यांचे सोर्स ओव्हरलॅप होऊ शकतात. उदाहरणार्थ, परकीय शत्रू किंवा असंतुष्ट कर्मचार्यांद्वारे प्रतिकूल हल्ले केले जाऊ शकतात.



**नैसर्गिक आपत्ती:** भूकंप, चक्रीवादळ, पूर, वीज आणि आग यामुळे संगणक सिस्टिम चे गंभीर नुकसान होऊ शकते. इनफार्मेशन गमावली जाऊ शकते, वेळ कमी होऊ शकतो किंवा उत्पादकता कमी होऊ शकते आणि हार्डवेअरचे नुकसान इतर आवश्यक सर्विस मध्ये व्यत्यय आणू शकते. नैसर्गिक आपत्तींविरुद्ध काही सुरक्षा उपाय लागू केले जाऊ शकतात. आपत्ती पुनर्प्राप्ती योजना आणि आकस्मिक योजना तयार करणे हा सर्वोत्तम पाथ आहे.

### मानवी धमक्या

**A कर्मचारी/आतील व्यक्ती :** हेतुपुरस्सर आणि आकस्मिक चुका आणि कर्मचारी आणि आतल्या लोकांच्या दुर्भावनापूर्ण कृत्यांमुळे दूरसंचार उद्योगात मोठ्या प्रमाणात नुकसान आणि तोटा होतो.

असंतुष्ट कर्मचारी संगणक सिस्टिम वर गैरटाइप आणि तोडफोड दोन्ही घडवू शकतात. सार्वजनिक आणि प्राइवेट दोन्ही क्षेत्रातील त्यांच्या

नोक-यांमधून काढून टाकलेल्या कर्मचार्यांनी महत्त्वपूर्ण संस्थात्मक ज्ञान असलेल्या व्यक्तींचा एक ग्रुप तयार केला आहे जो संभाव्य सिस्टम एक्सेस राखून ठेवू शकतो. संकेतशब्द अवैध करून आणि वेळेवर सिस्टम खाती हटवून सिस्टम व्यवस्थापक हा धोका मर्यादित करू शकतात.

### मालिशियस हॅकर्स

दुर्भावनापूर्ण धमक्यांमध्ये असंतुष्ट किंवा दुर्भावनापूर्ण कर्मचार्यांचे इंटरनल हल्ले आणि केवळ एखाद्या संस्थेला हानी पोहोचवू आणि व्यत्यय आणू पाहणाऱ्या नोकरदारांद्वारे बाहेरील हल्ले यांचा समावेश होतो. दुर्भावनापूर्ण हल्लेखोरांना सामान्यतः सिस्टमवरील हल्ल्यासाठी स्पेसिफिक ध्येय, उद्दिष्ट किंवा हेतू असतो.

सर्विस नाकारणे (DoS) अटॅक टूल्स वापरून सर्विस आणि व्यवसाय ऑपरेशन्सची सातत्य व्यत्यय आणणे ही उद्दिष्टे असू शकतात. त्यांना इनफार्मेशन चोरायची असेल किंवा लॅपटॉप कॉम्प्युटरसारखे हार्डवेअर देखील चोरायचे असेल. हॅकर्स स्पर्धकांना सबयोगी पडणारी इनफार्मेशन विकू शकतात.

सर्वात धोकादायक हल्लेखोर हे सहसा आतले (किंवा पूर्वीचे आतले) असतात, कारण त्यांना आधीच अस्तित्वात असलेले अनेक कोड आणि सुरक्षा उपाय माहित असतात. आतील व्यक्तींची स्पेसिफिक उद्दिष्टे आणि उद्दिष्टे असण्याची शक्यता असते आणि त्यांना सिस्टममध्ये कायदेशीर एक्सेस असतो. कर्मचारी हे संस्थेच्या संगणक आणि एप्लिकेशन शी सर्वात परिचित असलेले लोक आहेत आणि कोणत्या कृतींमुळे सर्वात जास्त नुकसान होऊ शकते हे त्यांना माहित असण्याची शक्यता असते. आतील व्यक्ती व्हायरस, ट्रोजन हॉर्स किंवा वर्म्स लावू शकतात आणि ते फाइल सिस्टमद्वारे ब्राउझ करू शकतात.

तथापि, असंतुष्ट करंट कर्मचारी प्रत्यक्षात माजी कर्मचार्यांपेक्षा जास्त नुकसान करतात. संगणक-संबंधित कर्मचारी तोडफोडीच्या सामान्य उदाहरणांमध्ये हे समाविष्ट आहे:

- डेटा बदलणे
- डेटा हटवत आहे
- लॉजिक बॉम्बसह डेटा किंवा प्रोग्राम नष्ट करणे
- क्रेडेंशियल सिस्टम
- डेटा ओलिस ठेवणे
- हार्डवेअर सुविधा नष्ट करणे
- चुकीच्या मेथड्स ने डेटा प्रविष्ट करणे

### फॉरेन विरोधक (Adversaries)

संगणक घुसखोर क्रियाकलाप आंतरराष्ट्रीय लेयर वर घडले आहेत, आंतरराष्ट्रीय गेटवेद्वारे घुसखोरीच्या प्रयत्नांची नंबर चिंताजनक दराने वाढत आहे. असे काही संकेत मिळाले आहेत की परदेशातील जमिनीखालील संगणक राजकीय अजेंडा पार पाडतात. काहीवेळा एका देशाच्या गुप्तचर सर्विस इतर देशांच्या कम्युनिकेशन सिस्टमला थेट टारगेट करतात, घुसतात किंवा तडजोड करतात.

### बाहेरील हल्लेखोर किंवा 'फटाके'

लोक सहसा "फटाकांना" "हॅकर्स" म्हणून संबोधतात. हॅकर हा शब्द अशा लोकांचा रेफरेन्स घेतो जे एकतर अशा सिस्टीममध्ये एक्सेस करतात ज्यासाठी त्यांच्याकडे कोणतीही अधिकृतता नाही किंवा त्यांच्याकडे कायदेशीर एक्सेस नसलेल्या सिस्टीमवर जाणूनबुजून मर्यादा ओलांडतात. सिस्टममध्ये एक्सेस मिळवण्याच्या सामान्य पद्धतींमध्ये पासवर्ड क्रॅक करणे, ज्ञात सुरक्षा कमकुवतपणाचे शोषण करणे, नेटवर्क स्पूफिंग आणि सोशल इंजिनीरिंग यांचा समावेश होतो.

### प्रतिकूल हल्ले

प्रतिकूल हल्ल्यांद्वारे, नेटवर्कच्या सबलब्धतेवर रिझल्ट करणे शक्य आहे. समन्वित आण्विक हल्ले, मर्यादित/असमन्वित आण्विक हल्ले, आण्विक अपघात, दहशतवाद, इलेक्ट्रॉनिक युद्ध, तोडफोड आणि राष्ट्रांच्या नेटवर्कवर नागरी अव्यवस्था यासारख्या प्रतिकूल हल्ल्यांचा प्रायमरी रिझल्ट म्हणजे व्यत्यय आणि सर्विस नाकारणे. अशा आपत्ती वितरित सर्विस च्या वेळेवर आणि गुणवत्तेवर रिझल्ट करतात.

### B NON मालिशियस कर्मचारी

केवळ हल्लेखोरच एखाद्या संस्थेला हानी पोहोचवू शकत नाहीत. डेटा अखंडतेला प्रायमरी धोका अज्ञान वापरकर्त्यांकडून येतो. हे अधिकृत वापरकर्ते आहेत ज्यांना ते करत असलेल्या कृतींबद्दल इनफार्मेशन नसते. त्रुटी आणि वगळणे मौल्यवान डेटा गमावू, नुकसान किंवा बदलू शकतात. वापरकर्ते, डेटा एंट्री क्लर्क, सिस्टम ऑपरेटर आणि प्रोग्रामर वारंवार अनावधानाने चुका करतात ज्यामुळे सुरक्षा समस्यांना थेट आणि अप्रत्यक्षपणे. काहीवेळा त्रुटी ही धमकी असते, जसे की डेटा एंट्री त्रुटी किंवा प्रोग्रामिंग त्रुटी जी सिस्टम क्रॅश करते. इतर प्रकरणांमध्ये, त्रुटी असुरक्षा निर्माण करतात.

### धमक्या ओळखण्यासाठी वापरल्या जाणाऱ्या सर्वोत्तम मेथड्स किंवा पाथ दर्शक तत्त्वे

धमक्या/हल्ले त्यांच्या प्रकारानुसार यशस्वीपणे ओळखण्यासाठी विविध यंत्रणा आणि मेथड्स वापरल्या जाऊ शकतात. दुसऱ्या शब्दांत, धोक्याच्या आधारावर, तुम्ही त्यांना ओळखण्यासाठी आणि त्यानुसार वर्गीकरण करण्यासाठी स्पेसिफिक तंत्रे वापरू शकता. खालील सर्वात सामान्य मेथड्स आहेत:

- विसंगती सर्च प्याच्या साधनांचा वापर
- प्रवाह-आधारित विश्लेषण वापरून नेटवर्क टेलीमेट्री
- घुसखोरी सर्च णे आणि घुसखोरी प्रतिबंधक प्रणालींचा वापर (IDS/IPS)
- नेटवर्क एलिमेंट लॉगचे विश्लेषण करणे (म्हणजे वेगवेगळ्या नेटवर्क डिव्हाइसेसवरून SYSLOG, अकाउंटिंग रेकॉर्ड, ऍप्लिकेशन लॉग, सिंपल नेटवर्क मॅनेजमेंट प्रोटोकॉल (SNMP) इ.

### धोके कमी करण्यासाठी वापरल्या जाणाऱ्या सर्वोत्तम मेथड्स किंवा पाथ दर्शक तत्त्वे

खालील काही मेथड्स आहेत ज्यांची अंमलबजावणी केल्यावर संस्थेची इनफार्मेशन, डेटा आणि विश्वासाहतेच्या सुरक्षिततेला धोका कमी होण्याची शक्यता असते.

## सुरक्षा जागरूकता प्रशिक्षण

- बहुतेक सुरक्षेचे उल्लंघन प्रत्यक्षात कंपन्यांमध्ये असंतुष्ट किंवा निष्काळजी कर्मचाऱ्यांकडून होते. त्यामुळे तुमच्या कंपनीतील प्रत्येकाला शिक्षित करा जेणेकरून ते विविध सुरक्षा धोके ओळखण्यात मदत करू शकतील.
- कर्मचारी ईमेल फिशिंग आणि स्फूफिंग हल्ले सर्च प्यात आणि ओळखण्यात सक्षम असावेत. त्यांना एनक्रिप्टेड नसलेली सेन्सिटिव्ह इनफार्मेशन स्टोर , पाठवू किंवा कॉपी न करण्याचे प्रशिक्षण दिले पाहिजे.
- जोपर्यंत त्यांना प्रेक्षकांबद्दल 100% खात्री नसते तोपर्यंत त्यांना फोनवर सेन्सिटिव्ह इनफार्मेशन शेअर करू नये हे माहित असले पाहिजे.
- कर्मचाऱ्यांना सुरक्षा धोरणे आणि पद्धतींचे प्रशिक्षण द्या. आणि त्यांना अद्ययावत करण्याचे सुनिश्चित करा आणि वारंवार पुन्हा प्रशिक्षण द्या.
- प्रशिक्षण कंटेंट ने कॉर्पोरेट धोरणांचे रिव्यु केले पाहिजे आणि कर्मचाऱ्यांमधील कोणत्याही संशयास्पद किंवा दुर्भावनापूर्ण वर्तनासाठी स्पष्टपणे तपशीलवार रिझल्ट केले पाहिजेत. त्यांना विविध सुरक्षा धोरणांचे प्रशिक्षण दिले पाहिजे, यासह:
  - स्वीकार्य वापर आणि अस्वीकार्य वापर
  - सामान्य वापर आणि मालकी
  - सुरक्षा आणि मालकीची इनफार्मेशन
  - ब्लॉगिंग आणि सोशल मीडिया
  - अंमलबजावणी म्हणजे. पालन न केल्याबद्दल शिस्तभंगाची कारवाई
  - सामाजिक माध्यमे
  - तुमची स्वतःची डिव्हाइस धोरणे आणा
    - i डेटा धोरण
    - ii मोबाइल डिव्हाइस मॅनेजमेंट (MDM) धोरण
    - iii मोबाइल डिव्हाइस सपोर्ट धोरण
    - iv कंपनीने जारी केलेल्या उपकरणांबाबत धोरणे
    - v तोटा आणि चोरी
    - vi कर्मचारी समाप्ती धोरण
    - vii सुरक्षा इवेंट्स मॅनेजमेंट

## अँटी-व्हायरस आणि अँटी-मालवेअर प्रोटेक्शन

- मालवेअरच्या सामान्य प्रकारांमध्ये हे समाविष्ट आहे: वर्म्स, की लॉगर्स, व्हिडिओ फ्रेम ग्रॅबर्स, rootकिट्स आणि ट्रोजन हॉर्स.
- चांगले अँटीव्हायरस प्रोग्राम इंस्टाल करा, अपडेट करा, शेड्यूल करा आणि चालवा.
- मालवेअर धोक्यांचा सामना करण्यासाठी "एंड पॉइंट सिक्युरिटी" धोरण स्वीकारा. एंडपॉइंट सिक्युरिटी ही इनफार्मेशन सुरक्षा संकल्पना

आहे ज्याचा अर्थ असा आहे की नेटवर्कवरील प्रत्येक डिव्हाइस (किंवा एंडपॉइंट) स्वतःच्या सुरक्षिततेसाठी जबाबदार आणि सक्षम असावे.

- तुमचे अँटी-मालवेअर सोल्यूशन कोणतेही असो, ते संलग्न व्हायरससाठी ईमेल स्कॅन केले पाहिजे, संक्रमणासाठी रिअल टाइममध्ये फाइल्सचे निरीक्षण करा आणि प्रत्येक फाइलचे कसून स्कॅन करा.

## डेटा एन्क्रिप्शन

- डेटा एन्क्रिप्शन इनफार्मेशन सुरक्षिततेचा एक शक्तिशाली पार्ट आहे. एन्क्रिप्शन तुमचा डेटा ऍक्सेस केल्यानंतरही त्याचे प्रोटेक्शन करते.

## पॅचिंग

- तुमच्या संगणक सिस्टिम ला धोका कमी करण्यासाठी पॅचिंग आवश्यक आहे. सिस्टिम आणि ऍप्लिकेशन्समधील सुरक्षा छिद्रांचे निराकरण करण्यासाठी पॅचेस सहसा सोडले जातात. तुम्ही चालवलेल्या सर्व ऑपरेटिंग सिस्टीम आणि ऍप्लिकेशन्स पॅच केलेले असल्याची खात्री करा. सर्व नेटवर्क उपकरणांवर नवीनतम फर्मवेअर अपडेट इंस्टाल करा.

## एक्सेस कंट्रोल

- वाढीव सुरक्षिततेसाठी, कर्मचाऱ्यांना (आणि भागीदारांना) त्यांना आवश्यक असलेल्या डेटामध्ये एक्सेस द्या. यात फिजिकल आणि लॉजिकल एक्सेस दोन्ही समाविष्ट आहेत.
- कमीत कमी विशेषाधिकार देऊन सुरुवात करा. त्यानंतर तुम्ही आवश्यकतेनुसार अनधिकृत डेटामध्ये एक्सेस करण्याची परवानगी देण्यासाठी विशेषाधिकार वाढवू शकता.

## मोबाइल उपकरणे

- लॅपटॉप, स्मार्टफोन आणि टॅब्लेट यांनी आजच्या कामगारांची उत्पादकता आणि गतिशीलता वाढवली आहे. पण त्यासोबतच उत्पादकताही येते. हरवलेले किंवा चोरीला गेलेले लॅपटॉप आणि इतर मोबाइल उपकरणे हे डेटा भंगाचे प्रमुख कारण आहेत.
- ऑटो-लॉक सक्षम करा किंवा सर्व डिव्हाइसेसमध्ये एक्सेस करण्यासाठी पासवर्ड आवश्यक आहे.

## देखरेख

- तुमचा व्यवसाय कोणत्याही विकृतीसाठी सिस्टिम आणि नेटवर्क डिव्हाइसेसचे निरीक्षण करण्यासाठी सेट केल्याची खात्री करा.
- सर्व ठिकाणे किंवा पायाभूत सुविधा - नेटवर्क, सिस्टीम आणि युजर क्रियाकलाप यांच्याकडून इनफार्मेशन संकलित करा आणि परस्परसंबंधित करा.
- फक्त फायरवॉल किंवा IPS वरील क्रियाकलाप अवरोधित करू नका. लॉग इन करा, त्याचे रिव्यु करा आणि त्यातून शिका.
- तुमच्या व्यवसायातील यूजर च्या क्रियाकलापांचे परीक्षण करण्यासाठी कंटेंट फिल्टरिंग इंस्टाल करा. कर्मचाऱ्यांकडून इंटरनेटचा गैरवापर करण्याचा सर्वात सामान्य टाइप म्हणजे नको त्या साइट सर्फ करणे



## फायरवॉल

- फायरवॉल नियम आणि धोरणे कॉन्फिगर करा कारण फायरवॉल ही कोणत्याही हल्ल्यापासून (नेटवर्क किंवा होस्ट) संरक्षणाची पहिली ओळ आहे. हे सार्वजनिक नेटवर्क आणि प्राइवेट नेटवर्क दरम्यान अडथळा कार्य करते.

## रिमोट बॅकअप

- तुमच्या डेटाचा नियमितपणे दूरस्थ स्थानावर बॅकअप घ्या. बॅकअप हे संगणकीय क्षेत्रातील सर्वात दुर्लक्षित क्षेत्रांपैकी एक आहे आणि त्यामुळे तुमच्या व्यवसायाला रिस्क कमी करण्याच्या सर्वात मोठ्या संधीपैकी एक आहे.
- बर्खाचदा, व्यवसाय हॅकर्स किंवा मालवेअरपासून डेटा सुरक्षित करण्यासाठी गुंतवणूक करतात, परंतु नंतर नैसर्गिक कारणांमुळे डेटा

फिजिकल रित्या नष्ट केला जातो. जर डेटा अस्तित्वात नसेल, तर तो बाहेरील धोक्यांपासून सुरक्षित करण्यात काही अर्थ नाही.

## सुरक्षा व्हॅल्यु आणि एक्सेस टेस्टिंग

- तुमचा व्यवसाय सुरक्षित करण्यासाठी तुम्ही सतर्क राहणे आवश्यक आहे. तुमच्या व्यवसायाच्या इनफार्मेशन शी तडजोड करण्याचा पुढील पाथ सर्च गारे चुकीचे हेतू असलेले लोक नेहमीच असतात
- नवीन रिस्क ओळखण्यासाठी वार्षिक किंवा अधिक चांगले, त्रैमासिक असुरक्षा व्हॅल्यु मापन करा. सतत बदलणारे सुरक्षा वातावरण नेहमीच नवीन धोके निर्माण करत असते.
- तुमच्या व्यवसायाला लागू होणारे नवीन धोके ओळखा आणि इतर कोणीतरी ते सर्चण्यापूर्वी त्यांचे निराकरण करा.
- दर तीन वर्षांनी औपचारिक इनफार्मेशन सुरक्षा रिस्क असेसमेंट करा, जे आजकाल बहुतेक उत्पादनांचे जीवन चक्र आहे

## इनफार्मेशन सुरक्षा भेद्यता आणि रिस्क मॅनेजमेंट (Information security vulnerabilities and Risk Management)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- सुरक्षा असुरक्षा डिस्क्रिप्शन करा
- भेद्यतेच्या प्रकारांचे डिस्क्रिप्शन करा
- धमक्यांचा सिस्टिम वर कसा रिझल्ट होतो ते स्पष्ट करा
- असुरक्षा व्हॅल्यु मापन साधने आणि तंत्रांचे डिस्क्रिप्शन करा
- सुरक्षा भेद्यता कमी करण्यासाठी सर्वोत्तम पद्धतींचे डिस्क्रिप्शन करा
- धोका, असुरक्षा आणि रिस्क यांच्यातील संबंधांचे डिस्क्रिप्शन करा
- विविध धमकी एजंट्सचे डिस्क्रिप्शन करा
- रिस्क मॅनेजमेंट च्या उद्देशाचे डिस्क्रिप्शन करा
- रिस्क मॅनेजमेंट च्या प्रकारांचे डिस्क्रिप्शन करा
- ICT पुरवठा साखळी मॅनेजमेंट तील रिस्क स्पष्ट करा.

## परिचय

संगणक सुरक्षेमध्ये, असुरक्षा ही एक सॉफ्टवेअर, हार्डवेअर, प्रक्रियात्मक किंवा मानवी कमजोरी आहे जी आक्रमणकर्त्याला सिस्टिमची इनफार्मेशन आश्वासन कमी करण्यास अनुमती देते. भेद्यता ही एक कमकुवतता आहे जी आक्रमणकर्त्याला संगणक किंवा नेटवर्कमध्ये एक्सेस करण्यासाठी सर्च त असलेले खुले दार प्रदान करू शकते आणि वातावरणातील संसाधनांमध्ये अनधिकृत एक्सेस आहे. हे सुरक्षा उपायाची अनुपस्थिती किंवा कमकुवतपणा दर्शवते ज्याचा यूजेस केला जाऊ शकतो.

## धोका, धमकी एजंट, शोषण, रिस्क आणि असुरक्षितता.

- धमकी इनफार्मेशन किंवा सिस्टिम साठी कोणताही संभाव्य धोका आहे. धोका असा आहे की कोणीतरी, किंवा काहीतरी, स्पेसिफिक असुरक्षा ओळखेल आणि ती कंपनी किंवा व्यक्तीविरुद्ध वापरेल.
- असुरक्षिततेचा फायदा घेणाऱ्या एलिमेंटला a म्हणून संबोधले जाते धमकी एजंट.

- शोषण असुरक्षिततेचा फायदा घेण्याचे आणि सिस्टिम किंवा नेटवर्कचा फायदा घेण्यासाठी ते वापरण्याचे साधन आहे.
- धोका :- धोका एजंट असुरक्षिततेचा आणि संबंधित व्यवसाय प्रभावाचा फायदा घेण्याची शक्यता आहे.

दुसऱ्या शब्दांत, धोका म्हणजे ज्यापासून आपण प्रोटेक्शन करण्याचा प्रयत्न करत आहोत, असुरक्षितता ही आपल्या संरक्षणाच्या प्रयत्नांमधील कमकुवतपणा किंवा अंतर आहे.

## इनफार्मेशन सुरक्षा भेद्यता का अस्तित्वात आहे?

कोड किंवा नेटवर्किंग प्रोटोकॉलमधील शोषणांमुळे भेद्यता अस्तित्वात आहे. ऑपरेटिंग सिस्टिम बनवण्यासाठी लाखो कोडची आवश्यकता असते आणि काहीवेळा असुरक्षा त्यामध्ये आढळू शकतात. फायरवॉलमध्ये अनेक पोर्ट उघडे असल्यास, घुसखोर अनधिकृत मेथड्स ने नेटवर्कमध्ये एक्सेस करण्यासाठी एक वापरण्याची शक्यता जास्त असते. असुरक्षितता "हार्डवेअर, सॉफ्टवेअर किंवा प्रक्रियेतील त्रुटी किंवा कमकुवतपणाचा

रिझल्ट असू शकतो ज्यामुळे सिस्टिम तडजोड होऊ शकते". हे एक कमकुवतपणा, डिझाइन किंवा अंमलबजावणी त्रुटीचे अस्तित्व आहे ज्यामुळे संगणक सिस्टिम , नेटवर्क, एप्लिकेशन किंवा प्रोटोकॉलच्या सुरक्षिततेची तडजोड करणारी अनपेक्षित, अनिष्ट इवेंट्स घडू शकते.

## A तांत्रिक भेद्यता एक टाइप

बऱ्याच सॉफ्टवेअर सुरक्षा भेद्यता वर्गाच्या एका छोट्या संचामध्ये येतात: • बफर ओव्हरफ्लो

- अप्रमाणित इनपुट
- शर्यतीची परिस्थिती
- एक्सेस -कंट्रोल समस्या
- प्रमाणीकरण, अधिकृतता, किंवा मध्ये कमकुवतपणा क्रिप्टोग्राफिक मेथड्स

### बफर ओव्हरफ्लो

बफर ओव्हरफ्लो, जो असुरक्षिततेचा एक प्रमुख सोर्स मानला जातो, जेव्हा एखादा एप्लिकेशन बफरच्या शेवटच्या पलीकडे (किंवा, कधीकधी, सुरुवातीपूर्वी) डेटा लिहिण्याचा प्रयत्न करतो तेव्हा उद्भवतो. बफर ओव्हरफ्लोमुळे ॲप्लिकेशन्स क्रॅश होऊ शकतात, डेटाशी तडजोड होऊ शकते आणि ॲप्लिकेशन ज्या सिस्टिमवर चालत आहे त्या सिस्टीमशी तडजोड करण्यासाठी पुढील विशेषाधिकार वाढीसाठी ॲटॅक वेक्टर देऊ शकतात. बफर ओव्हरफ्लो हल्ले सामान्यतः स्टॅक, हीप किंवा दोन्हीशी तडजोड करून होतात.

### अप्रमाणित इनपुट

सामान्य नियमानुसार, डेटा वाजवी किंवा वैध आहे याची खात्री करण्यासाठी तुम्ही तुमच्या प्रोग्रामद्वारे प्राप्त झालेले सर्व इनपुट तपासले पाहिजेत. ज्या प्रकरणांमध्ये अवैध डेटाला परवानगी दिली जाते किंवा स्वीकारली जाते, अशा टाइप ची फाइल वाचण्याचा प्रयत्न करणारा एक सामान्य प्रोग्राम चुकीच्या आकाराचा बफर वाटप करण्याचा प्रयत्न करतो, ज्यामुळे हीप ओव्हरफ्लो हल्ला किंवा इतर समस्या उद्भवण्याची शक्यता असते. या कारणास्तव, तुम्ही तुमचा इनपुट डेटा काळजीपूर्वक तपासला पाहिजे. ही प्रक्रिया सामान्यतः इनपुट प्रमाणीकरण किंवा विवेक तपासणी म्हणून ओळखली जाते.

अविश्वासू स्रोताकडून तुमच्या प्रोग्रामला मिळालेले कोणतेही इनपुट हे हल्ल्याचे संभाव्य टारगेट आहे. हॅकर्स प्रोग्राममधील इनपुटच्या प्रत्येक स्रोताकडे पाहतात आणि त्यांना कल्पना करू शकतील अशा प्रत्येक टाइप च्या विकृत डेटामध्ये पास करण्याचा प्रयत्न करतात. प्रोग्राम क्रॅश झाल्यास किंवा अन्यथा गैरवर्तन केल्यास, हॅकर नंतर समस्येचे शोषण करण्याचा पाथ सर्च प्याचा प्रयत्न करतो.

### शर्यतीच्या अटी

शर्यतीची स्थिती उद्भवते जेव्हा अनुप्रयोगातील नियमित प्रोग्रामिंग कॉलची जोडी नियमानुसार अभिप्रेत असलेल्या अनुक्रमिक मेथड्स ने कार्य करत नाही. हा सॉफ्टवेअरमधील एक टाइमिंग इव्हेंट आहे जो कॉल योग्य क्रमाने न केल्यास सुरक्षा असुरक्षा बनू शकतो. प्रोग्रामच्या योग्य कार्यासाठी अंमलबजावणीचा योग्य क्रम आवश्यक असल्यास, हा एक बग आहे.

जर एखादा आक्रमणकर्ता दुर्भावनापूर्ण कोड घालण्यासाठी, फाइलनाव बदलण्यासाठी किंवा अन्यथा प्रोग्रामच्या सामान्य ऑपरेशनमध्ये व्यत्यय आणण्यासाठी परिस्थितीचा फायदा घेऊ शकत असेल, तर शर्यतीची स्थिती ही एक सुरक्षा भेद्यता आहे. आक्रमणकर्ते काहीवेळा कोडच्या प्रक्रियेतील लहान वेळेच्या अंतराचा फायदा घेऊन ऑपरेशन्सच्या क्रमामध्ये व्यत्यय आणू शकतात, ज्याचा ते नंतर शोषण करतात.

सॉफ्टवेअर डेव्हलपमेंटमध्ये, वापरण्याच्या वेळेची तपासणी (TOCTOU किंवा TOCTOU, "टॉक टू" असे उच्चारले जाते) हा सॉफ्टवेअर बगचा एक क्लास आहे जो एखाद्या कंडिशनची तपासणी (जसे की सुरक्षा क्रेडेन्शियल) आणि वापरादरम्यान सिस्टिममधील बदलांमुळे होतो. परिणामांची. हे शर्यतीच्या स्थितीचे एक उदाहरण आहे.

### इंटरप्रोसेस कम्युनिकेशन (IPC)

इंटरप्रोसेस कम्युनिकेशन (IPC) हा प्रोग्रामिंग इंटरफेसचा एक संच आहे जो प्रोग्रामरला ऑपरेटिंग सिस्टिममध्ये एकाच वेळी चालू असलेल्या विविध प्रोग्राम प्रक्रियांमधील क्रियाकलाप समन्वयित करण्यास अनुमती देतो. हे एका प्रोग्रामला एकाच वेळी अनेक वापरकर्त्यांच्या विनंत्या हाताळण्यास अनुमती देते. इंटरप्रोसेस कम्युनिकेशनसाठी वापरलेले हे मेसेजिंग प्रोटोकॉल अनेकदा हल्ल्यासाठी असुरक्षित असतात.

**रिमोट प्रोसिजर कॉल (RPC)** ही इंटरप्रोसेस कम्युनिकेशन मेकॅनिझम आहे जी एका होस्टवर चालणाऱ्या प्रोग्रामला रिमोट होस्टवर कोड रन करण्यास अनुमती देते.

### असुरक्षित फाइल ऑपरेशन्स

वेळ-चेक-चेक-टाईम-ऑफ-वापर समस्यांव्यतिरिक्त, इतर अनेक फाइल ऑपरेशन्स असुरक्षित आहेत. प्रोग्रामर बहुतेकदा एखाद्या फाइलच्या मालकी, पोजिशन किंवा गुणधर्मांबद्दल गृहीत धरतात जे कदाचित खरे नसतील. उदाहरणार्थ, तुम्ही असे गृहीत धरू शकता की तुम्ही तुमच्या प्रोग्रामद्वारे तयार केलेल्या फाइलवर नेहमी लिहू शकता. तथापि, आपण ती तयार केल्यानंतर आक्रमणकर्ता त्या फाइलवरील परवानग्या किंवा ध्वज बदलू शकत असल्यास, आणि आपण लेखन ऑपरेशननंतर रिझल्ट कोड तपासण्यात अयशस्वी झाल्यास, फाइलमध्ये छेदछाड केली गेली आहे हे आपल्याला आढळणार नाही.

### B मूळ असुरक्षिततेचे टाइप

मूळ भेद्यतेची उदाहरणे आहेत:

- सँडबॉक्सिंग मेकॅनिझममधील भेद्यता ज्यामुळे अविश्वासू बायकोडला सुरक्षा व्यवस्थापकाने लादलेल्या निर्बंधांना टाळता येते
- Java क्लास लायब्ररीमधील भेद्यता ज्यावर एप्लिकेशन त्याच्या सुरक्षिततेसाठी अवलंबून असतो

### सुरक्षा भेद्यता समजून घेणे

#### सॉफ्टवेअर किंवा प्रोटोकॉल डिझाइनमधील त्रुटी

सॉफ्टवेअर डिझाइनमधील बेसिक चुका आणि उपेक्षा हे डिझाइनच्या असुरक्षिततेचे कारण आहे. डिझाइनमधील त्रुटींमुळे सॉफ्टवेअर सुरक्षित नसल्यामुळे ते उच्च स्तरीय असुरक्षिततेचे प्रकरण बनते.

संगणक नेटवर्क प्रोटोकॉलवर अवलंबून असतात जे रनटाइमच्या वेळी देवाणघेवाण केलेले संदेश, त्यांचे फॉरमॅट आणि स्ट्रक्चर स्पेसिफाइड करतात. प्रोटोकॉल वेगवेगळ्या प्रोटोकॉल स्टॅकशी जोडलेले आहेत, उदा., TCP/IP, किंवा भिन्न मॉडेल्स, उदा., OSI, आणि अधोस्पेसिफाइड सुरक्षा असलेले बरेच प्रोटोकॉल अजूनही ट्रांसक्शन त आहेत. सदोष प्रोटोकॉलमुळे उद्भवलेल्या काही असुरक्षा खाली डिस्क्रिप्शन केल्या आहेत:

A TCP/IP. TCP/IP प्रोटोकॉल स्टॅकमध्ये काही कमकुवत गुण आहेत जे परवानगी देतात:

- **स्पूफिंग:** जेव्हा एखादा मालिशियस पक्ष नेटवर्क होस्टवर हल्ला करण्यासाठी, डेटा चोरण्यासाठी, मालवेअर पसरवण्यासाठी किंवा एक्सेस कंट्रोल बायपास करण्यासाठी नेटवर्कवरील दुसऱ्या डिव्हाइसची किंवा यूजर ची तोतयागिरी करतो तेव्हा स्पूफिंग हल्ला होतो. हे पूर्ण करण्यासाठी मालिशियस पक्ष वापरू शकतात असे अनेक टाइप चे स्पूफिंग हल्ले आहेत. ते IP अॅड्रेस स्पूफिंग अॅटॅक्स, ARP (अॅड्रेस रिझोल्यूशन प्रोटोकॉल) स्पूफिंग अॅटॅक्स, DNS सर्व्हर स्पूफिंग अॅटॅक्स इ.
- **टेलनेट प्रोटोकॉल:** टेलनेटचा वापर Microsoft Windows 2000 आणि Unix चालवणाऱ्या प्रणालींचे मॅनेजमेंट करण्यासाठी केला जाऊ शकतो. टेलनेट क्लायंटचा वापर Microsoft सिस्टीम वरून UNIX सिस्टीमशी कनेक्ट करताना आणि त्याउलट, युजर नावे आणि संकेतशब्द स्पष्ट टेक्स्ट मध्ये प्रसारित केले जातात त्यामुळे सुरक्षा असुरक्षा निर्माण होते.
- **फाइल ट्रान्सफर प्रोटोकॉल (FTP):** फाइल ट्रान्सफर प्रोटोकॉल वापरकर्त्यांना रिमोट सिस्टमशी कनेक्ट करण्याची आणि फायली पुढे-मागे हस्तांतरित करण्याची परवानगी देतो. रिमोट कॉम्प्युटरशी कनेक्शन इंस्टाल करण्याचा पार्ट म्हणून, FTP प्रमाणीकरणासाठी युजर नाव आणि पासवर्ड संयोजनावर अवलंबून असते. FTP चा वापर टेलनेट प्रोटोकॉलच्या वापराप्रमाणेच सुरक्षा समस्या निर्माण करतो कारण FTP वर टाइप केलेले पासवर्ड नेटवर्कवर साध्या टेक्स्ट मध्ये प्रसारित केले जातात, प्रत्येक पॅकेटमध्ये एक कॅरक्टर . ही पॅकेट्स रोखली जाऊ शकतात.
- **प्रोटोकॉल आणि सॉफ्टवेअर कसे अंमलात आणले जातात यामधील कमकुवतता**  
जरी एखादे प्रोटोकॉल चांगले डिझाइन केलेले असले तरी, ते ज्या प्रकारे लागू केले जाते त्यामुळे ते असुरक्षित असू शकते. उदाहरणार्थ, इलेक्ट्रॉनिक मेलसाठी प्रोटोकॉल अशा प्रकारे लागू केला जाऊ शकतो जो घुसखोरांना पीडिताच्या मशीनच्या मेल पोर्टशी कनेक्ट करण्याची परवानगी देतो आणि मशीनला सेवेद्वारे अभिप्रेत नसलेले कार्य करण्यास मूर्ख बनवतो. या टाइप ची भेद्यता घुसखोरांना पीडिताच्या सिस्टमवरील खात्यात एक्सेस न करता रिमोट साइटवरून पीडिताच्या मशीनवर हल्ला करण्यास सक्षम करते.
- सॉफ्टवेअर रिलीझ होण्यापूर्वी ओळखल्या गेलेल्या त्रुटीमुळे सॉफ्टवेअर असुरक्षित असू शकते. या टाइप च्या भेद्यतेमध्ये सबवर्गाची विस्तृत रेन्ज आहे, ज्याचे घुसखोर अनेकदा त्यांच्या स्वतः च्या हल्ल्याच्या साधनांचा

वापर करून शोषण करतात जसे की फाइल प्रवेशामध्ये शर्यतीची परिस्थिती, डेटा कंटेंट आणि आकाराची अस्तित्वात नसलेली तपासणी इ.

## सिस्टम आणि नेटवर्क कॉन्फिगरेशनमधील कमकुवतपणा

- सिस्टम एडमिनिस्ट्रेटर आणि वापरकर्ते नेटवर्क कॉन्फिगरेशनमधील डीफॉल्ट सेटिंग्ज बदलण्याकडे दुर्लक्ष करू शकतात किंवा ते नेटवर्कला असुरक्षित ठेवणाऱ्या मार्गाने ऑपरेट करण्यासाठी त्यांची सिस्टम सेट करू शकतात.
- एसिक्रोनस ट्रान्सफर मोड (ATM). “मॅनहोल मॅनिप्युलेशन” म्हणून ओळखल्या जाणाऱ्या सुरक्षेची तडजोड केली जाऊ शकते - नेटवर्क केबल्स आणि भूमिगत पार्किंग गॅरेज आणि लिफ्ट शाफ्टमधील कनेक्शनमध्ये थेट एक्सेस .
- फ्रेम रिले. एटीएमच्या समस्येप्रमाणेच.
- सबकरण एडमिनिस्ट्रेशन . स्विच आणि राउटर सहजपणे HTTP इंटरफेसद्वारे किंवा कमांड लाइन इंटरफेसद्वारे मॅनेज केले जातात. कमकुवत संकेतशब्द (उदाहरणार्थ, सार्वजनिक संकेतशब्द) वापरण्यासाठी जोडलेले, हे काही तांत्रिक ज्ञान असलेल्या कोणालाही डिव्हाइसवर कंट्रोल ठेवण्याची परवानगी देते.
- मोडेम. मॉडेम “फायरवॉल” ला बायपास करते जे नेटवर्कला बाहेरील घुसखोरांपासून संरक्षित करते. मॉडेम टेलिफोन नंबर ओळखण्यासाठी “वॉर डायलर” टूल आणि कमकुवत पासवर्ड तोडण्यासाठी “पासवर्ड क्रॅकर” टूल वापरणारा हॅकर सिस्टममध्ये एक्सेस मिळवू शकतो.
- वेब किंवा क्लाउड ऍप्लिकेशन्समधील कमकुवतता

जेव्हा एखादी संस्था त्यांचे महत्त्वपूर्ण ऍप्लिकेशन्स आणि डेटा क्लाउड कॉम्प्युटिंग वातावरणात हलवण्यास तयार असते तेव्हा अनेक महत्त्वपूर्ण भेद्यता विचारात घेतल्या पाहिजेत, या भेद्यता खाली डिस्क्रिप्शन केल्या आहेत:

### A सत्र रायडिंग आणि अपहरण

सेशन रायडिंग म्हणजे हॅकर्सने लक्षित यूजर च्या वतीने वेब ऍप्लिकेशनला फक्त ईमेल पाठवून किंवा यूजर ला खास तयार केलेल्या वेबसाइटला भेट देण्यासाठी फसवून कमांड पाठवण्याचा रेफरेन्स दिला जातो. सत्र रायडिंग यूजर चा डेटा हटवते, ऑनलाइन व्यवहार जसे की बोली किंवा ऑर्डर कार्यान्वित करते, इंटरनेटद्वारे इंटरनेट सिस्टमला स्पॅम पाठवते आणि सिस्टम तसेच नेटवर्क कॉन्फिगरेशन बदलते किंवा फायरवॉल देखील उघडते.

### B व्हर्चुअल मशीन एस्केप

VM एस्केप ही एक भेद्यता आहे जी अतिथी-स्तरीय VM ला त्याच्या होस्टवर हल्ला करण्यास सक्षम करते. या असुरक्षा इंटरनल आक्रमणकर्ता VM वर कोड चालवतो जो त्याच्या आत चालणाऱ्या OS ला बाहेर पडू देतो आणि थेट हायपरवाइजरशी संवाद साधू देतो.

## C विश्वासार्हता आणि सेवेची उपलब्धता

क्लाउड स्टोरेज इन्फ्रास्ट्रक्चर बऱ्याच काळासाठी खाली जाऊ शकते, ज्यामुळे डेटा गमावला जाऊ शकतो आणि वेब सर्विस सह एक्सेस समस्या उद्भवू शकतात.

## D असुरक्षित क्रिप्टोग्राफी

हल्लेखोर कोणतीही क्रिप्टोग्राफिक यंत्रणा किंवा अल्गोरिदम डीकोड करू शकतात कारण त्यांना हॅक करण्याच्या मुख्य मेथड्स सर्च ल्या जातात.

## E डेटा प्रोटेक्शन आणि पोर्टेबिलिटी

जरी क्लाउड सर्विस क्लायंट आणि प्रदात्यामधील कराराच्या आधारावर ऑफर केल्या जातात परंतु करार संपुष्टात आल्यावर काय होईल आणि क्लायंट यापुढे चालू ठेवू इच्छित नाही.

## F विक्रेता लॉक-इन

ही भेद्यता अपरिपक्व प्रदाते आणि नवीन व्यवसाय मॉडेल्समुळे उद्भवते ज्यामुळे अपयश आणि व्यवसायातून बाहेर जाण्याचा धोका वाढतो.

## G इंटरनेट अवलंबित्व

क्लाउड कॉम्प्युटिंग हे इंटरनेटवर अवलंबून असलेले तंत्रज्ञान आहे जेथे वापरकर्ते वेब ब्राउझरद्वारे सर्विस मध्ये एक्सेस करत आहेत. इंटरनेट सबलब्ध नसल्यास किंवा सर्विस बंद असल्यास, वापरकर्त्यांच्या सिस्टम आणि ऑपरेटिंग्जचे काय होईल जे अत्यंत गंभीर आहेत आणि हेल्थकेअर आणि बँकिंग सिस्टम सारख्या 24 तास चालविण्याची आवश्यकता आहे.

## ऑनलाइन ई-ट्रांसक्शन तील कमकुवतपणा

ऑनलाइन पेमेंट सिस्टमच्या सुरक्षेवर होणार्या हल्ल्यांच्या संख्येत आणि प्रकारात समान वाढ झाल्यामुळे ऑनलाइन व्यवहारांमध्ये प्रचंड वाढ झाली आहे. यापैकी काही हल्ल्यांनी असुरक्षा वापरल्या आहेत ज्या वेबसाइट्सद्वारे वापरल्या जाणार्या पुन्हा वापरता येण्याजोग्या तृतीय-पक्ष एलिमेंटमध्ये प्रकाशित केल्या आहेत, जसे की शॉपिंग कार्ट सॉफ्टवेअर. इतर हल्ल्यांमध्ये SQL इंजेक्शन किंवा क्रॉस-साइट स्क्रिप्टिंग सारख्या कोणत्याही वेब ऍप्लिकेशनमध्ये सामान्य असलेल्या भेद्यता वापरल्या गेल्या आहेत.

ऑनलाइन व्यवहारांमध्ये सामान्य टाइप च्या भेद्यता म्हणजे SQL इंजेक्शन, क्रॉस-साइट स्क्रिप्टिंग, इनफार्मेशन प्रकटीकरण, पाथ प्रकटीकरण, किमतीमध्ये फेरफार आणि बफर ओव्हरफ्लो.

या असुरक्षिततेचे यशस्वी शोषण केल्याने परिणामांची विस्तृत रेन्ज मिळू शकते. इनफार्मेशन आणि पाथ प्रकटीकरण असुरक्षा सामान्यतः स्टार्टिंग टप्पे म्हणून कार्य करतील ज्यामुळे पुढील शोषण होईल. एसक्यूएल इंजेक्शन किंवा किमतीत फेरफार हल्ले वेबसाइटला अपंग करू शकतात, प्रायव्हसी शी तडजोड करू शकतात आणि सर्वात वाईट परिस्थितीत ई. व्यापार व्यवसाय पूर्णपणे बंद.

अशा असुरक्षिततेचे एक मुख्य कारण हे आहे की वेब ऍप्लिकेशन डेव्हलपर बहुतेक वेळा सुरक्षित प्रोग्रॅमिंग तंत्राशी निपुण नसतात.

## ब्राउझर सुरक्षा आणि कुकीज आणि पॉप-अपची भूमिका

सुरक्षा भेद्यतेमुळे कुकीचा डेटा हॅकरद्वारे वाचला जाऊ शकतो, यूजर च्या डेटामध्ये एक्सेस मिळवण्यासाठी वापरला जातो किंवा कुकी ज्या वेबसाइटशी संबंधित आहे त्या वेबसाइटवर (यूजर च्या क्रेडेन्शियल्ससह) एक्सेस मिळवण्यासाठी वापरला जातो.

## पॉप-अप जाहिराती किंवा पॉप-अप

पॉपअप विंडोमधले "सुरक्षा" धोके म्हणजे फिशिंग, नको त्या वेब साइट्सवर अडकणे इ.

## ब्राउझर, वेब ऍप्लिकेशन्स, ओएस आणि स्मार्टफोनमधील सुरक्षा

**छिद्र:** सुरक्षेच्या परिभाषेत, छिद्र म्हणजे सॉफ्टवेअर किंवा ऑपरेटिंग सिस्टमच्या असुरक्षिततेचा रेफरेन्स घेतो ज्याचा यूजेस संगणक सिस्टिम किंवा नेटवर्कच्या संपूर्ण सुरक्षिततेशी तडजोड करण्यासाठी केला जाऊ शकतो.

- ऑपरेटिंग सिस्टिमच्या भेद्यता म्हणजे लिनक्स कर्नलला प्रभावित करणाऱ्या; किंवा मायक्रोसॉफ्ट, ऍपल किंवा मालकीच्या युनिक्स विक्रेत्याने उत्पादित केलेल्या ऑपरेटिंग सिस्टमसह पाठवलेले आणि विक्रेत्याद्वारे ऑपरेटिंग सिस्टमचा पार्ट म्हणून परिभाषित केलेले एलिमेंट .
- ब्राउझर भेद्यता म्हणजे वेब ब्राउझरचा पार्ट म्हणून परिभाषित केलेल्या एलिमेंटना प्रभावित करणारे. यामध्ये Mozilla Firefox आणि Google Chrome सारख्या तृतीय-पक्ष ब्राउझरसह Windows Internet Explorer आणि Apple's Safari सारख्या ऑपरेटिंग सिस्टिमसह शिप करणारे वेब ब्राउझर समाविष्ट आहेत.
- ऍप्लिकेशन भेद्यता म्हणजे ऑपरेटिंग सिस्टिम विक्रेते आणि इतर विक्रेत्यांद्वारे प्रकाशित एलिमेंटसह इतर सर्व एलिमेंटवर रिझल्ट करणारे. ओपन सोर्स एलिमेंटमधील भेद्यता जे लिनक्स डिस्ट्रीब्युशन सह पाठवू शकतात (जसे की X विंडो सिस्टम, GNOME डेस्कटॉप वातावरण, GIMP, आणि इतर) ऍप्लिकेशन असुरक्षा मानल्या जातात.

## वेब ऍप्लिकेशन्समधील सुरक्षा होल्स

OWASP (ओपन वेब ऍप्लिकेशन्स सिक्युरिटी प्रोजेक्ट) रेन्ज तील शीर्ष 10 धोक्यांची यादी खालीलप्रमाणे आहे.

इंजेक्शन (Sqli -> SQL इंजेक्शन), तुटलेली प्रमाणीकरण आणि सत्र मॅनेजमेंट, XSS (क्रॉस साइट स्क्रिप्टिंग), असुरक्षित थेट ऑब्जेक्ट रेफरेन्स, सुरक्षा चुकीची कॉन्फिगरेशन, सेन्सिटिव्ह डेटा एक्सपोजर, गहाळ फंक्शन लेव्हल ऍक्सेस कंट्रोल, क्रॉस साइट रिक्वेस्ट फोर्जरी (CSRF किंवा XSRF), ज्ञात असुरक्षा आणि अप्रमाणित पुनर्निर्देशन आणि फॉरवर्डसह एलिमेंट वापरणे.

## OS मध्ये सुरक्षा होल्स

UNIX OS मधील काही भेद्यता म्हणजे Setuid समस्या, ट्रोजन हॉर्स आणि टर्मिनल टूबल

Windows OS मधील काही भेद्यता म्हणजे पासवर्ड, पीअर टू पीअर फाइल शेअरिंग, मायक्रोसॉफ्ट आउटलुक आणि Outlook Express मधील एम्बेडेड ऑटोमेशन वैशिष्ट्यांमधील भेद्यता ज्यामुळे रॉग कोडची अंमलबजावणी होऊ शकते.

LINUX OS मधील काही भेद्यता गहाळ परवानगी तपासण्या, अननिशिअल डेटा आणि मेमरी गैरमॅनेजमेन्ट आहेत.

स्मार्टफोनमधील काही भेद्यता म्हणजे डिव्हाइस गमावणे किंवा चोरी झाल्यामुळे डेटा लीक होणे, डेटाचे अनावधानाने प्रकटीकरण, बंद केलेल्या स्मार्टफोनवरील हल्ले, फिशिंग हल्ले, स्पायवेअर हल्ले, नेटवर्क स्फूफिंग हल्ले, पाळत ठेवणे (निगराणीखाली युजर ) हल्ले, डायलरवेअर हल्ले (पैसे चोरणे) , आर्थिक मालवेअर हल्ले (क्रेडेन्शियल चोरणे) आणि नेटवर्क गर्दी.

### असुरक्षितता असेसमेंट साधने आणि तंत्रे

असुरक्षा व्हॅल्यु मापन हा एक सुरक्षा एक्सरसाईस आहे जो कमकुवतपणा ओळखतो, असुरक्षा ओळखतो आणि त्यांची गणना करतो आणि नेटवर्क, ऍप्लिकेशन्स आणि सिस्टममधील सुरक्षा दायित्वांबद्दलच्या शोधांवर रिपोर्ट देतो.

### असुरक्षा व्हॅल्यु मापन याद्वारे असुरक्षा सर्च ते:

- सुरक्षा तंत्रज्ञान
  - VA स्कॅनर, उपकरणे आणि सॉफ्टवेअर
- उपाय तंत्रज्ञान
  - पॅच मॅनेजमेंट सिस्टम्स (WSUS, SCCM, LanDesk, VMWare अपडेट मॅनेजर)

### असुरक्षितता व्हॅल्यु कनामध्ये प्रामुख्याने खालील तीन स्टेप्स चा समावेश होतो:

- इनफार्मेशन गोळा करणे आणि सर्च ज्यात नेटवर्क स्कॅनिंग, पोर्ट स्कॅनिंग, डिरेक्टरी सर्विस आणि DNS झोन आणि नोंदणी यांचा समावेश आहे.
- प्रगणना ज्यामध्ये यजमान आणि ओएस, पोर्ट, सर्विस आणि त्यांच्या आवृत्त्या, इनफार्मेशन आणि SNMP समुदाय समाविष्ट आहेत
- दुर्बलता ओळखणे, भेद्यता ओळखणे, रिपोर्ट तयार करणे आणि उपाय साधने वापरणे यांचा समावेश असलेला सर्च .

### असुरक्षितता व्हॅल्यु साधने

असुरक्षा व्हॅल्यु मापन साधने विविध स्तरांवर आढळलेल्या असुरक्षिततेचा रिझल्ट सर्च तात, ओळखतात, मोजतात. बहुतांश भेद्यता व्हॅल्यु मापन साधने नेटवर्किंग आणि नेटवर्क उपकरणे (स्विच, फायरवॉल, प्रिंटर इ.) तसेच सर्व्हर, डेस्कटॉप आणि पोर्टेबल संगणकांसह अनेक नेटवर्क नोड्स स्कॅन करण्यास सक्षम आहेत.

नेटवर्क स्कॅनर, होस्ट स्कॅनर, डेटाबेस स्कॅनर, वेब ऍप्लिकेशन स्कॅनर, मल्टीलेव्हल स्कॅनर, ऑटोमेटेड पेनिट्रेशन टेस्ट टूल्स आणि व्हलनेरेबिलिटी स्कॅन कन्सोलिडेटर्स ही कॉमन व्हलनेरेबिलिटी असेसमेंट टूल्स आहेत.

### असुरक्षिततेचे शोषण करण्याचे तंत्र

असुरक्षितता माजी साठी शोषण केले जाऊ शकते. पॅकेट स्निफरच्या वापराने. बनावट पत्त्यांसह पॅकेट तयार करण्यासाठी इतर साधने वापरली जातात; या साधनांचा एक वापर म्हणजे डिनायल-ऑफ-सर्व्हिस अटॅक अशा प्रकारे माउंट करणे जे हल्ल्याचा सोर्स लपवते. घुसखोर संगणक एड्रेस देखील "स्फूफ" करतात, त्यांची खरी ओळख लपवतात आणि यशस्वीरित्या कनेक्शन बनवतात ज्यांना अन्यथा परवानगी दिली जाणार नाही. अशाप्रकारे, ते संगणकांमधील विश्वासाहर्ष संबंधांचे शोषण करतात.

ट्रोजन, व्हायरस, वर्म्स, लॉजिक बॉम्ब, फिशिंग, फॉरवर्डिंग आणि शहरी दंतकथा सामायिक करणे, नायजेरियन घोटाळ्यांना प्रतिसाद देणे इत्यादीद्वारे सर्वात सामान्य शोषण होते.

### असुरक्षा दूर करण्यासाठी तंत्र

प्रभावी उपायांसाठी सतत प्रक्रियांची आवश्यकता असते ज्यांना एकत्रितपणे असुरक्षा मॅनेजमेन्ट म्हणतात. असुरक्षा मॅनेजमेंट द्वारे परिभाषित केलेल्या प्रक्रिया आणि संबंधित तंत्रज्ञान संस्थांना नेटवर्क सुरक्षा भेद्यता कार्यक्षमतेने सर्च प्यात आणि निराकरण करण्यात मदत करते. या प्रक्रियांचा मेथड्स शीर वापर व्यवसाय प्रणालींना वारंवार होणार्या व्हायरस, वर्म्स आणि इतर नेटवर्क-जनित हल्ल्यांपासून प्रोटेक्शन करतो.

असुरक्षितता मॅनेजमेंट च्या निरंतर प्रक्रियेमध्ये सुरक्षा धोरणे आणि कंट्रोल तयार करणे, मालमत्तेचा मागोवा घेणे / मालमत्तेचे वर्गीकरण करणे, असुरक्षिततेसाठी स्कॅनिंग सिस्टम, इन्व्हेंटरीच्या विरुद्ध भेद्यतेची कम्प्यारीजन करणे, जोखमीचे वर्गीकरण करणे, पॅचची पूर्व टेस्टिंग , पॅचेस लागू करणे आणि पुन्हा स्कॅनिंग आणि कन्फर्म करणे समाविष्ट आहे. सुरक्षा ऍप्लिकेशन आणि वेबबेस्ड सर्विस सह तुम्ही आता त्यापैकी बहुतांश ऑटोमॅटिक करू शकता.

### सुरक्षा भेद्यता कमी करण्यासाठी सर्वोत्तम मेथड्स आणि पाथ दर्शक तत्त्वे

- 1 वापरण्यापूर्वी सर्व व्हेरिफाईड सुरू करा
- 2 वापरण्यापूर्वी सर्व युजर इनपुट सत्यापित करा
- 3 सर्व्हर आणि डेटाबेसवरील अडमिनिस्ट्रेटिव्ह परवानग्या प्रतिबंधित करा
- 4 त्रुटी हाताळा आणि शेवटच्या वापरकर्त्यासाठी सिस्टम त्रुटी संदेश प्रदर्शित करू नका
- 5 आवश्यक असलेल्या कमीत कमी परवानग्या आणि विशेषाधिकारांसह खाती प्रदान करा
- 6 तुमच्या कोडमध्ये रहस्ये (उदा. पासवर्ड, की) संचयित करू नका
- 7 सामान्य कार्यासाठी टेस्टिंग केलेली, विश्वासाहर्ष लायब्ररी किंवा मॉड्यूल वापरा (उदा. प्रमाणीकरण, एन्क्रिप्शन, सत्र टॅकिंग)
- 8 HTTPS सह प्रमाणीकरणाद्वारे संरक्षित लॉगिन पेजेज आणि पेजेज सुरक्षित करा

- 9 सर्व्हर एलिमेंट (OS, सॉफ्टवेअर/अॅप्स) अद्ययावत असल्याची खात्री करा
- 10 उत्पादन सर्व्हरवर अनावश्यक एप्लिकेशन इंस्टाल करणे टाळा
- 11 वेब सर्व्हरवरून न वापरलेली आणि बॅकअप पेजेज काढा
- 12 शक्य असल्यास, कोड लायब्ररी आणि कॉन्फिगरेशन फाइल्स वेबवरून एक्सेस करण्यायोग्य बनवा
- 13 डिरेक्टरी ब्राउझिंग अक्षम करा
- 14 यूजर च्या इनपुटवर आधारित ऑपरेटिंग सिस्टम कॉल करणे टाळा
- 15 तुमच्या डेव्हलपमेंट फ्रेमवर्कमध्ये तयार केलेल्या सेशन ट्रॅकिंग मेकॅनिझमचा वापर करा.

रिस्क म्हणजे एखाद्या घटनेची संभाव्यता आणि त्याचे रिझल्ट यांचे संयोजन. हे दिलेल्या हल्ल्याद्वारे लक्षित होण्याची शक्यता दर्शवते.

### धोका, असुरक्षितता आणि रिस्क यांच्यातील संबंध

धोका, भेद्यता आणि रिस्क यांच्यातील संबंध परिभाषित करण्यापूर्वी आपण खालील अटींचे रिव्यु करूया:

**प्रॉपर्टी** : इनफार्मेशन सुरक्षेमध्ये, प्रॉपर्टी म्हणजे आम्ही संरक्षित करण्याचा प्रयत्न करतो. ते लोक, प्रॉपर्टी किंवा इनफार्मेशन असू शकते.

**धमकी**: हेतुपुरस्सर किंवा चुकून असुरक्षिततेचे शोषण करणारी कोणतीही गोष्ट आणि प्रॉपर्टी मिळवणे, नुकसान करणे किंवा नष्ट करणे.

**भेद्यता**: हे आमच्या प्रोटेक्शन प्रयत्नांमधील कमकुवतपणा किंवा तफावत दर्शवते.

**धोका**: जेव्हा एखादी धमकी एखाद्या असुरक्षिततेचा फायदा घेते, तेव्हा त्यामुळे मालमतेचे नुकसान, नुकसान किंवा नाश होऊ शकतो. याला रिस्क म्हणतात.

त्यामुळे रिस्क म्हणजे प्रॉपर्टी, धमक्या आणि भेद्यता यांचा छेदनपॉइंट.

म्हणजे, प्रॉपर्टी x धोके x भेद्यता = धोका

### मालमतेचे व्हॅल्यू

इनफार्मेशन वर ठेवलेले व्हॅल्यू हे गुंतलेल्या पक्षांशी संबंधित आहे. ते विकसित करण्यासाठी कोणते काम आवश्यक आहे, त्याची देखभाल करण्यासाठी किती खर्च येईल, ती हरवल्यास किंवा नष्ट झाल्यास काय नुकसान होईल, शत्रूंना कोणते पैसे द्यावे लागतील आणि कोणते दायित्व दंड होऊ शकतात. सहन करणे. जर एखाद्या कंपनीला इनफार्मेशन चे व्हॅल्यू आणि ती संरक्षित करण्याचा प्रयत्न करत असलेल्या इतर मालमतेची इनफार्मेशन नसेल, तर कंपनीने त्यांचे प्रोटेक्शन करण्यासाठी किती पैसा आणि वेळ खर्च करावा हे माहित नसते. मालमतेची व्हॅल्यू नियुक्त करताना, खाली नमूद केलेल्या काही मुद्द्यांचा विचार करणे आवश्यक आहे.

- मालमतेचे अधिग्रहण किंवा विकास करण्यासाठी लागणारा खर्च
- मालमतेची देखभाल आणि प्रोटेक्शन करण्यासाठी खर्च
- मालक आणि वापरकर्त्यांसाठी मालमतेचे व्हॅल्यू

- विरोधकांना मालमतेचे व्हॅल्यू
- बौद्धिक मालमतेचे व्हॅल्यू जे इनफार्मेशन विकसित करण्यासाठी गेले
- इतर लोक मालमतेसाठी देऊ इच्छित असलेली किंमत
- प्रॉपर्टी हरवल्यास किंवा नुकसान झाल्यास पुनर्स्थित करण्याची किंमत
- प्रॉपर्टी अनुपलब्ध असल्यास प्रभावित होणारे ऑपरेशनल आणि उत्पादन क्रियाकलाप
- मालमतेशी तडजोड केल्यास दायित्व समस्या
- संस्थेतील मालमतेची सबयुक्तता आणि भूमिका

मालमतेचे व्हॅल्यू समजून घेणे ही कोणती सुरक्षा यंत्रणा वापरावी आणि तिचे प्रोटेक्शन करण्यासाठी कोणते निधी वापरावे हे समजून घेण्याची पहिली स्टेप आहे.

### धोका सोर्स /एजंट म्हणजे काय?

धोका सोर्स किंवा धमकी एजंटइरादा आणि इफेक्ट पाडण्याची क्षमता असलेली एक संस्था आहे.

धमकी देणारे एजंट मालमतेविरुद्ध खालीलपैकी एक किंवा अधिक क्रिया करू शकतात:

- एक्सेस - साधा अनधिकृत एक्सेस
- गैरवापर - मालमतेचा अनधिकृत वापर (उदा. ओळख चोरी, तडजोड केलेल्या सर्व्हरवर अश्लील वितरण सर्विस सेट करणे इ.)
- खुलासा करा - धमकी देणारा एजंट बेकायदेशीरपणे सेन्सिटिव्ह इनफार्मेशन उघड करतो
- सुधारित करा - मालमतेत अनधिकृत बदल
- एक्सेस नाकारणे - यात नाश, नॉनडेटा मालमतेची चोरी इत्यादींचा समावेश आहे.

धमकी देणारे एजंट खालीलपैकी कोणतेही असू शकतात:

या व्यक्ती आणि गटांचे खालीलप्रमाणे वर्गीकरण करता येईल.

- नॉन-टार्गेट स्पेसिफिक: नॉन-टार्गेट स्पेसिफिक थ्रेट एजंट म्हणजे कॉम्प्युटर व्हायरस, वर्म्स, ट्रोजन आणि लॉजिक बॉम्ब.
- कर्मचारी: कर्मचारी, कंत्राटदार, परिचालन/देखभाल करणारे कर्मचारी किंवा सुरक्षा रक्षक जे कंपनीवर नाराज आहेत.
- संघटित गुन्हेगारी आणि गुन्हेगार: गुन्हेगार त्यांच्यासाठी महत्त्वाची असलेली इनफार्मेशन टार्गेट करतात, जसे की बँक खाती, क्रेडिट कार्ड किंवा बौद्धिक संपदा ज्याचे पैशात रूपांतर केले जाऊ शकते. गुन्हेगार अनेकदा त्यांना मदत करण्यासाठी आतल्यांचा वापर करतात.
- कॉर्पोरेशन: कॉर्पोरेशन आक्षेपार्ह इनफार्मेशन युद्ध किंवा स्पर्धात्मक बुद्धिमत्तेत गुंतलेली असतात. भागीदार आणि प्रतिस्पर्धी या रेन्ज त येतात.
- मानवी, नकळत: अपघात, निष्काळजीपणा.

- मानवी, हेतुपुरस्सर: आतला, बाहेरचा.
- नैसर्गिक: पूर, आग, वीज, उल्का, भूकंप.

## रिस्क कंट्रोल

रिस्क कमी करणे हे इनफार्मेशन सुरक्षेचे केंद्रपॉइंट असल्यास, हे टारगेट साध्य करण्यासाठी कंट्रोल ही प्राथमरी साधने आहेत. कंट्रोल हे कोणतेही साधन किंवा प्रक्रिया आहे जी रिस्क कमी करण्यासाठी वापरली जाते.

बेसिक पणे तीन टाइप चे कंट्रोल आहेत:

- 1 अडमिनिस्ट्रेटिव्ह :** अडमिनिस्ट्रेटिव्ह कंट्रोल म्हणजे लोक ज्या कृती करतात. अडमिनिस्ट्रेटिव्ह कंट्रोल ही धोरणे आणि कार्यमिथड्स विकसित करण्याची आणि त्यांचे पालन सुनिश्चित करण्याची प्रक्रिया आहे
- 2 तांत्रिक किंवा लॉजिकल :** हे आभासी, एप्लिकेशन आणि तांत्रिक कंट्रोल (सिस्टम आणि सॉफ्टवेअर), जसे की फायरवॉल, अँटी व्हायरस सॉफ्टवेअर, एन्क्रिप्शन आणि मेकर/चेकर ऑप्लिकेशन रूटीन आहेत. तांत्रिक कंट्रोल संगणक सिस्टिम द्वारे चालविली जातात किंवा मॅनेज केली जातात.
- 3 क्रियाकलाप चरण कंट्रोल** एकतर तांत्रिक किंवा अडमिनिस्ट्रेटिव्ह असू शकतात आणि रिस्क कमी करण्याच्या लेयर वर आधारित खालीलप्रमाणे वर्गीकृत केले आहेत:
  - दुर्बलतेच्या संपर्कात येण्यापासून धोका टाळण्यासाठी प्रतिबंधात्मक कंट्रोल अस्तित्वात आहेत. ही अशी कंट्रोल आहेत जी नुकसान किंवा हानी होण्यापासून प्रतिबंधित करतात. उदाहरणार्थ, जबाबदाऱ्यांचे पृथक्करण लागू करणारे कंट्रोल (एक व्यक्ती पेमेंट विनंती सबमिट करू शकते, परंतु दुसऱ्या व्यक्तीने ती अधिकृत केली पाहिजे), कर्मचारी फसव्या पेमेंट जारी करण्याची शक्यता कमी करते.
  - आमच्या सिस्टममध्ये धोका आला आहे हे ओळखण्यासाठी गुप्तचर कंट्रोल अस्तित्वात आहेत. ही कंट्रोल कार्यमिथड्स किंवा कार्यमिथड्स पाळल्या गेल्या नसल्याची उदाहरणे ओळखण्यासाठी क्रियाकलापांचे निरीक्षण करतात. उदाहरणार्थ, फसव्या पेमेंट्स ओळखण्यासाठी एखादा व्यवसाय सामान्य लेजर किंवा रिव्यु पेमेंट विनंती ऑडिट लॉगचा समेट करू शकतो.
  - धोक्याचे रिझल्ट कमी करण्यासाठी किंवा कमी करण्यासाठी सुधारात्मक कंट्रोल अस्तित्वात आहेत. सुधारात्मक कंट्रोल सिस्टिम रिस्टोअर करतात किंवा हानीकारक घटनेच्या आधीच्या स्थितीत प्रक्रिया करतात. उदाहरणार्थ, एखाद्या व्यक्तीने पेमेंट डेटामध्ये अयोग्यरित्या बदल केल्याचे पुरावे आढळल्यानंतर एखादा व्यवसाय बँकअप टेपमधून सिस्टमची संपूर्ण रिस्टोअर करू शकतो.
  - भरपाई देणारी कंट्रोल ही मूळ नियंत्रणांचा हेतू शक्य तितक्या जवळून पूर्ण करण्यासाठी डिझाइन केलेली वैकल्पिक कंट्रोल आहेत, जेव्हा मूलतः डिझाइन केलेली कंट्रोल पर्यावरणाच्या मर्यादांमुळे वापरली जाऊ शकत नाहीत.

## धोक्याची शक्यता

रिस्क संभाव्यता ही स्पेसिफिक असुरक्षा आक्रमणकर्त्याद्वारे उघडकीस येण्याची आणि शोषण करण्याची किती शक्यता आहे याचे एक ढोबळ माप आहे. यामध्ये अति तंतोतंत असणे आवश्यक नाही अंदाज साधारणपणे, शक्यता कमी, मध्यम किंवा जास्त आहे की नाही हे ओळखणे पुरेसे आहे.

संभाव्यता निर्धारित करण्यात मदत करणारे अनेक एलिमेंट आहेत. एलिमेंटचा पहिला संच धोका एजंटशी संबंधित आहे. संभाव्य हल्लेखोरांच्या गटाकडून यशस्वी हल्ल्याच्या संभाव्यतेचा अंदाज घेणे हे टारगेट आहे. लक्षात घ्या की एकापेक्षा जास्त धोका एजंट असू शकतात जे एखाद्या स्पेसिफिक असुरक्षिततेचे शोषण करू शकतात, त्यामुळे सर्वात वाईट परिस्थिती वापरणे सामान्यतः सर्वोत्तम आहे. उदाहरणार्थ, निनावी बाहेरील व्यक्तीपेक्षा एक आतला व्यक्ती अधिक संभाव्य हल्लेखोर असू शकतो, परंतु ते अनेक एलिमेंटवर अवलंबून असते.

एलिमेंटचा पहिला संच धोका एजंटशी संबंधित आहे. धोका एजंट्सच्या या गटाद्वारे यशस्वी हल्ल्याच्या संभाव्यतेचा अंदाज घेणे हे येथे टारगेट आहे. या हल्ल्यांमागील हेतू लक्षात घेण्याजोगा दुसरा एलिमेंट आहे. आवश्यक एक्सेस किंवा रिसोर्सेस आणि धोक्याच्या एजंट्सच्या गटाचा आकार हे इतर एलिमेंट विचारात घेतले पाहिजेत.

एलिमेंटचा पुढील संच गुंतलेल्या भेद्यतेशी संबंधित आहे. येथे उद्दिष्ट सर्च ले जाणे आणि शोषण करण्यात गुंतलेल्या स्पेसिफिक असुरक्षिततेच्या संभाव्यतेचा अंदाज घेणे आहे. हे सर्च प्याची सुलभता, शोषणाची सुलभता, धोक्याच्या एजंटच्या या गटाबद्दल जागरूकता, या शोषणाचा सर्च घेण्याची शक्यता लक्षात घेते.

## प्रभावाचा अंदाज लावणारे एलिमेंट

यशस्वी हल्ल्याच्या प्रभावाचा विचार करताना, दोन टाइप चे इफेक्ट आहेत हे लक्षात घेणे महत्त्वाचे आहे. पहिला म्हणजे अनुप्रयोगावरील “तांत्रिक इफेक्ट”, तो वापरत असलेला डेटा आणि ती प्रदान करणारी फंक्शन. दुसरा व्यवसाय आणि एप्लिकेशन चालविणाऱ्या कंपनीवर “व्यवसाय इफेक्ट” आहे.

शेवटी, व्यवसाय इफेक्ट अधिक महत्त्वाचा आहे. तथापि, यशस्वी शोषणाचे व्यावसायिक रिझल्ट सर्च प्यासाठी आवश्यक असलेल्या सर्व इनफार्मेशन मध्ये तुम्हाला एक्सेस नसेल. या प्रकरणात, तांत्रिक जोखमीबद्दल अधिक तपशील प्रदान केल्याने व्यावसायिक जोखमीबद्दल निर्णय घेण्यास योग्य व्यावसायिक प्रतिनिधी सक्षम होईल.

तांत्रिक इफेक्ट हे चिंतेच्या पारंपारिक सुरक्षा क्षेत्रांशी एलाइनमेंट एलिमेंटमध्ये विपार्ट ले जाऊ शकतात: प्रायव्हंसी, अखंडता, उपलब्धता आणि जबाबदारी. जर असुरक्षिततेचा फायदा घ्यायचा असेल तर सिस्टिम वर किती इफेक्ट पडेल याचा अंदाज लावणे हे उद्दिष्ट आहे. प्रायव्हंसी चे नुकसान, एकात्मता कमी होणे, उपलब्धता कमी होणे आणि जबाबदारी कमी होणे हे मुद्दे विचारात घेतले जातील.

आर्थिक नुकसान, प्रतिष्ठेचे नुकसान, धोरणांचे पालन न करणे आणि प्रायव्हंसी चे उल्लंघन हे व्यावसायिक प्रभावाचे असेसमेंट करण्यासाठी विचारात घेतले जाणारे एलिमेंट आहेत.

रिस्क कंट्रोल रिझल्ट कारकता: रिस्क कंट्रोल रिझल्ट कारकता ओळखल्या गेलेल्या सिस्टिम गत जोखमीच्या संख्येवर अवलंबून असते, रिस्क व्हॅल्यू कनामध्ये सामील असलेल्या प्रक्रियेच्या क्षेत्रांची टक्केवारी, कमी केलेल्या प्रमुख जोखमींची टक्केवारी आणि अनेक एलिमेंटमधील मुख्य जोखमींचे परीक्षण केले जाते.

### रिस्क मॅनेजमेन्ट

रिस्क मॅनेजमेन्ट आणि रिस्क असेसमेंट हे इनफार्मेशन सुरक्षा मॅनेजमेन्ट (ISM) चे प्रमुख एलिमेंट आहेत.

रिस्क मॅनेजमेन्ट म्हणजे संस्थेद्वारे वापरल्या जाणाऱ्या इनफार्मेशन संसाधनांना भेद्यता आणि धोके ओळखण्याची आणि मालमत्तेच्या मूल्यावर आधारित ते कमी करण्यासाठी कोणते प्रतिकारक उपाय योजावेत हे ठरवण्याची प्रक्रिया आहे.

रिस्क मॅनेजमेंट ची प्रक्रिया ही एक सतत चालणारी, पुनरावृत्ती प्रक्रिया आहे. ते अनिश्चित काळासाठी पुनरावृत्ती करणे आवश्यक आहे. व्यवसायाचे वातावरण सतत बदलत आहे आणि दररोज नवीन धोके आणि असुरक्षा उदयास येत आहेत. दुसरे, रिस्क मॅनेज करण्यासाठी वापरल्या जाणाऱ्या काउंटरमेजर्स (कंट्रोल ) च्या निवडीने उत्पादकता, किंमत, प्रतिकाराची रिझल्ट कारकता आणि संरक्षित केलेल्या इनफार्मेशन च्या मालमत्तेचे व्हॅल्यू यांच्यात संतुलन राखणे आवश्यक आहे.

याउलट, रिस्क असेसमेंट वेगळ्या वेळेच्या बिंदूवर (उदा. वर्षातून एकदा, मागणीनुसार, इ.) आणि पुढील व्हॅल्यू कनाची कामगिरी होईपर्यंत अंमलात आणले जाते - असेसमेंट केलेल्या जोखमींचे तात्पुरते व्ह प्रदान करते.

### रिस्क मॅनेजमेन्ट चा उद्देश

संस्थेमध्ये रिस्क मॅनेज करण्याचे मुख्य कारण म्हणजे संस्थेचे ध्येय आणि मालमत्तेचे प्रोटेक्शन करणे. म्हणून, रिस्क मॅनेजमेन्ट हे तांत्रिक कार्य न करता मॅनेजमेन्ट कार्य असले पाहिजे. रिस्क समजून घेणे, आणि स्पेशल तः, सिस्टममधील स्पेसिफिक धोके समजून घेणे, सिस्टम मालकास संस्थेच्या मूल्याशी सुसंगत इनफार्मेशन सिस्टिम चे प्रोटेक्शन करण्यास अनुमती देते. वस्तुस्थिती अशी आहे की सर्व संस्थांकडे मर्यादित रिसोर्सेस आहेत आणि रिस्क कधीही शून्यावर कमी केली जाऊ शकत नाही. म्हणून, रिस्क समजून घेणे, स्पेशल तः जोखमीचे प्रमाण, संस्थांना दुर्मिळ संसाधनांना प्राधान्य देण्यास अनुमती देते.

### रिस्क असेसमेंट (टप्पे)

रिसोर्सेस कोठे निर्देशित करायची हे निर्धारित करण्यात मॅनेजमेंट स मदत करणे हा जोखमीचे असेसमेंट करण्याचा उद्देश आहे. रिस्क मॅनेज करण्यासाठी चार बेसिक धोरणे आहेत: कमी करणे, हस्तांतरण, स्वीकृती आणि टाळणे.

### मिटिगेशन

शमन करणे ही सर्वात सामान्यतः मानली जाणारी रिस्क मॅनेजमेन्ट धोरण आहे. कमी करण्यामध्ये दोष दूर करणे किंवा दोषाशी संबंधित संभाव्यता किंवा इफेक्ट कमी करण्यासाठी काही टाइप चे भरपाई कंट्रोल प्रदान करणे समाविष्ट आहे. तांत्रिक सुरक्षेच्या दोषासाठी एक सामान्य उपाय

म्हणजे विक्रेत्याने प्रदान केलेला पॅच इंस्टाल करणे. कधीकधी शमन धोरण ठरवण्याच्या प्रक्रियेला कंट्रोल विश्लेषण म्हणतात.

### हस्तांतरण

हस्तांतरण ही दुसऱ्या पक्षाला तुमच्या वतीने रिस्क स्वीकारण्याची परवानगी देण्याची प्रक्रिया आहे. हे आयटी सिस्टीमसाठी मोठ्या प्रमाणावर केले जात नाही, परंतु प्रत्येकजण त्यांच्या वैयक्तिक जीवनात नेहमीच करतो. कार, आरोग्य आणि जीवन विमा हे सर्व रिस्क हस्तांतरित करण्याचे पाथ आहेत. या प्रकरणांमध्ये, रिस्क व्यक्तीकडून विमा कंपनीसह विमाधारकांच्या समूहाकडे हस्तांतरित केली जाते. लक्षात घ्या की यामुळे संभाव्यता कमी होत नाही किंवा कोणत्याही दोषांचे निराकरण होत नाही, परंतु यामुळे संस्थेवरील एकूण इफेक्ट (प्रामुख्याने आर्थिक) कमी होतो.

### स्वीकृती

स्वीकृती म्हणजे सिस्टीमला ज्ञात जोखमीसह ऑपरेट करण्याची परवानगी देण्याची प्रथा. अनेक कमी धोके सहज स्वीकारले जातात. कमी करण्यासाठी अत्यंत उच्च खर्चाची रिस्क देखील अनेकदा स्वीकारली जाते. मॅनेजमेंट द्वारे स्वीकारल्या जाणाऱ्या उच्च जोखमींपासून सावध रहा. ही रणनीती लिखित स्वरूपात असल्याची खात्री करा आणि निर्णय घेणाऱ्या व्यवस्थापकांनी ती स्वीकारली आहे. अनेकदा धोके स्वीकारली जातात जी स्वीकारली गेली नसावीत आणि नंतर जेव्हा घुसखोरी होते तेव्हा आयटी सुरक्षा कर्मचाऱ्यांना जबाबदार धरले जाते. सामान्यतः, व्यवसाय व्यवस्थापक, आयटी सुरक्षा कर्मचारी नव्हे, एखाद्या संस्थेच्या वतीने रिस्क स्वीकारण्यासाठी अधिकृत असतात.

### टाळणे

टाळणे म्हणजे व्यवस्थेतील असुरक्षित पैलू काढून टाकण्याची प्रथा किंवा अगदी व्यवस्थेचीच. उदाहरणार्थ, रिस्क व्हॅल्यू कनादरम्यान, एक वेबसाइट उघडकीस आली जी विक्रेत्यांना त्यांच्या पावत्या पाहू देते, ओळख म्हणून HTML फाइल नावामध्ये एम्बेड केलेला विक्रेता आयडी वापरून आणि प्रति विक्रेता कोणतेही प्रमाणीकरण किंवा अधिकृतता नाही. वेब पेजेज आणि संस्थेला जोखमीबद्दल सूचित केल्यावर, मॅनेजमेंट ने वेब पेजेज काढून टाकण्याचा आणि दुसऱ्या यंत्रणेद्वारे विक्रेता पावत्या प्रदान करण्याचा निर्णय घेतला. या प्रकरणात, असुरक्षित वेब पेजेज काढून टाकून धोका टळला.

### रिस्क असेसमेंट टाइप

#### परिमाणवाचक रिस्क असेसमेंट

परिमाणवाचक रिस्क असेसमेंट वित्तीय संस्था आणि विमा कंपन्यांनी वापरलेल्या पद्धतींवर आधारित आहे. इनफार्मेशन , सिस्टीम, व्यवसाय प्रक्रिया, पुनर्प्राप्ती खर्च इत्यादी व्हॅल्यू नियुक्त करून, इफेक्ट आणि त्यामुळे रिस्क , प्रत्यक्ष आणि अप्रत्यक्ष खर्चाच्या संदर्भात मोजली जाऊ शकते.

परंतु इनफार्मेशन प्रणालींमध्ये रिस्क मोजण्यासाठी याचा वापर सामान्यतः केला जात नाही कारण

- 1 मालमत्तेचे व्हॅल्यू ओळखण्यात आणि नियुक्त करण्यात अडचणी, आणि
- 2 नंबर शास्त्रीय इनफार्मेशन चा अभाव ज्यामुळे वारंवारता निश्चित करणे शक्य होईल.



अशा प्रकारे, इनफार्मेशन सिस्टिम साठी आज वापरल्या जाणाऱ्या रिस्क व्हॅल्यू मापन साधने बहुतेक गुणात्मक जोखमीचे मोजमाप आहेत.

### गुणात्मक रिस्क असेसमेंट

गुणात्मक रिस्क व्हॅल्यू मापन असे गृहीत धरते की संभाव्यता आणि इफेक्ट व्हॅल्यू मध्ये आधीच मोठ्या प्रमाणात अनिश्चितता आहे आणि त्यांना परिभाषित करते आणि अशा प्रकारे रिस्क , काही प्रमाणात व्यक्तिपरक किंवा गुणात्मक अटींमध्ये. परिमाणवाचक रिस्क व्हॅल्यू मापनातील मुद्दांप्रमाणेच, गुणात्मक रिस्क व्हॅल्यू कनातील मोठी अडचण संभाव्यता आणि रिझल्ट व्हॅल्यू ची व्याख्या करत आहे.

शिवाय, ही व्हॅल्यू अशा रीतीने परिभाषित करणे आवश्यक आहे ज्यामुळे समान स्केल एकापेक्षा जास्त रिस्क व्हॅल्यू कनांमध्ये सातत्याने वापरता येतील. गुणात्मक रिस्क असेसमेंट सामान्यतः "उच्च", "मध्यम" आणि "कमी" चे रिस्क रिझल्ट देतात. तथापि, इफेक्ट आणि संभाव्यता व्याख्या टेबल्स आणि प्रभावाचे डिस्क्रिप्शन प्रदान करून, संस्थेच्या मॅनेजमेंट स असेसमेंट पुरेसे संप्रेषित करणे शक्य आहे.

रिस्क व्हॅल्यू कनामध्ये खालील क्रिया आणि क्रियाकलाप समाविष्ट आहेत:

- मालमत्तेची ओळख;
- ओळखल्या गेलेल्या मालमत्तेसाठी संबंधित असलेल्या कायदेशीर आणि व्यावसायिक आवश्यकतांची ओळख
- ओळखलेल्या मालमत्तेचे असेसमेंट , ओळखल्या गेलेल्या कायदेशीर आणि
- व्यवसाय आवश्यकता आणि प्रायव्हंसी , अखंडता आणि उपलब्धता गमावल्याचे रिझल्ट
- ओळखल्या गेलेल्या मालमत्तेसाठी महत्त्वपूर्ण धोके आणि असुरक्षा ओळखणे;
- धमक्या आणि भेद्यता येण्याच्या संभाव्यतेचे असेसमेंट ;

## डिरेक्टरी सर्विस (Directory services)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- डिरेक्टरी आणि डिरेक्टरी सर्विस चे डिस्क्रिप्शन करा
- डिरेक्टरी सर्विस च्या फायद्यांचे डिस्क्रिप्शन करा
- डिरेक्टरी सर्विस च्या विविध अंमलबजावणीचा उल्लेख करा
- ऍक्टिव्ह डिरेक्टरी च्या लॉजिकल आणि फिजिकल संरचनेचे डिस्क्रिप्शन करा
- ग्लोबल कॅटलॉग आणि ग्रुप धोरणाचे डिस्क्रिप्शन करा.

### परिचय

#### डिरेक्टरी आणि डिरेक्टरी सर्विस

डिरेक्टरी डेटाचा संग्रह किंवा लिस्ट आहे. दूरध्वनी पुस्तके, जमिनीच्या नोंदी आणि कामांची लिस्ट ही वास्तविक जगाची उदाहरणे आहेत. या सर्व उदाहरणांचा उद्देश इनफार्मेशन सेव्ह करणे आणि संबंधित व्यक्तींना मागणीनुसार सबलब्ध करून देणे हा आहे.

- जोखमीची गणना;
- पूर्वनिर्धारित रिस्क स्केलच्या विरुद्ध जोखमीचे व्हॅल्यू मापन.

### ICT पुरवठा चैन मॅनेजमेंट मधील रिस्क

आयसीटी पुरवठा साखळीतील धोके प्रायव्हंसी , अखंडता किंवा इनफार्मेशन किंवा इनफार्मेशन सिस्टिम ची उपलब्धता गमावल्यामुळे उद्भवतात आणि संस्थात्मक ऑपरेशन्स (मिशन, फंक्शन, प्रतिमा किंवा प्रतिष्ठा यासह), संस्थात्मक प्रॉपर्टी , व्यक्ती आणि इतरांवर संभाव्य प्रतिकूल रिझल्ट दर्शवतात. संस्था अशा जोखमीचे फॉर्मेट आळशी सॉफ्टवेअर डेव्हलपमेंट, उत्पादने पाठवण्याआधी त्यांची अपुरी टेस्टिंग , आणि कमी खर्चिक पार्ट वापरणे असे असू शकतात जरी याचा अर्थ ते अस्सल नसले तरीही.

सप्लाय चैन रिस्क मॅनेजमेंट (SCRM) ही रिस्क समजून घेण्याची प्रक्रिया आहे, त्यांचे व्यावसायिक रिझल्ट आणि पुरवठा साखळीतील कमकुवतपणा आणि संपूर्ण सिस्टिम जीवनचक्रात शोषण कमी करून त्यांचे मॅनेजमेंट कसे करावे.

प्रभावी ICT SCRM ला प्रक्रिया, कार्यमेथड्स आणि साधने आवश्यक आहेत जी संस्थांना सर्व ICT प्रणालींवर सातत्याने SCRM तत्त्वे लागू करू देतात. असे एक तत्त्व म्हणजे बनावट भागांचा धोका कमी करणे कारण ते अप्रत्याशित वर्तन, लवकर अपयश किंवा वाईट होऊ शकतात. त्यामुळे बनावट पार्ट आणि अस्सल पार्ट वेगळे करणे आवश्यक आहे.

ही वैशिष्ट्ये व्यक्त करण्यासाठी संरचित भाषेची आवश्यकता आहे, जसे की पुरवठा साखळीतील सर्व सदस्य त्यांच्याबद्दल संवाद साधू शकतात आणि ज्याचा वापर इतरांना बनावट वस्तूंबद्दल सतर्क करण्यासाठी किंवा कायदेशीर वस्तूंचे निकष व्यक्त करण्यासाठी केला जाऊ शकतो. कायदेशीर आणि बेकायदेशीर दोन्ही एलिमेंटच्या या निरीक्षणीय गुणधर्मांचे डिस्क्रिप्शन करण्यासाठी एक संरचित भाषा पुरवठा शृंखला रिस्क कमी करण्यासाठी एक साधन आहे.

इनफार्मेशन तंत्रज्ञानामध्ये डिरेक्टरी हा शब्द एका स्पेसिफिक टाइप च्या डेटा स्टोरेजसाठी वापरला जातो. हे संरचित स्टोरेज आणि ऑब्जेक्ट्सचे कार्यक्षम पुनर्प्राप्ती करण्यास अनुमती देते जे सहसा वास्तविक जगातून घेतले जातात (उदा. व्यक्ती, IT उपकरणे). या स्टोरेजचे मुख्य वैशिष्ट्य म्हणजे सर्व डेटा तथाकथित नोंदींमध्ये स्टोर केला जातो. डिरेक्टरी तील नोंदींचा संच एक वृक्ष बनवतो (पदानुक्रमित डेटाबेस).

## डिरेक्टरी सर्विस

डिरेक्टरी सर्विस ही सामान्य वस्तू आणि नेटवर्क रिसोर्सेस सर्चणे, मॅनेज करणे, मॅनेज करणे आणि मॅनेज करणे यासाठी सामायिक केलेली इनफार्मेशन पायाभूत सुविधा आहे. यामध्ये व्हॉल्यूम, फोल्डर्स, फाइल्स, प्रिंटर, वापरकर्ते, ग्रुप, उपकरणे, टेलिफोन नंबर समाविष्ट असू शकतात

आणि इतर वस्तू. डिरेक्टरी सर्विस हे एक समाधान आहे जे वापरकर्त्यांना सु-परिभाषित इंटरफेसद्वारे डिरेक्टरी मध्ये स्टोर इनफार्मेशन मध्ये एक्सेस देते. नेटवर्क वापरले असल्यास, यासाठी योग्य प्रोटोकॉल परिभाषित करणे आवश्यक आहे. लाइट वेट डिरेक्टरी एक्सेस प्रोटोकॉल (LDAP) हा असा प्रोटोकॉल आहे.

### डिरेक्टरी सर्विस वापरण्याचे फायदे:

**1 संसाधन मॅनेजमेंट :** नेटवर्क एडमिनिस्ट्रेटर नेटवर्क रिसोर्सेस मॅनेज करण्यासाठी एकच साधन वापरू शकतो-- युजर खाती, सर्व्हर, ड्राइव्हस, फाइल्स, प्रिंटर इ. हे एका साध्या झाडाच्या संरचनेत प्रदर्शित केले जातात, त्यामुळे ते सर्चणे आणि मॅनेज करणे सोपे आहे. या संरचनेत, प्रत्येक संसाधन ऑब्जेक्ट नावाच्या चिन्हाद्वारे प्रदर्शित केले जाते. एखादी वस्तू निवडून, तिची सेटिंग्ज सबलब्ध असतात आणि नेटवर्क एडमिनिस्ट्रेटर त्याला योग्य वाटेल त्याप्रमाणे त्यात बदल करू शकतो.

**2 वापरकर्ते :** ज्या संसाधनांवर तुम्हाला अधिकार देण्यात आले आहेत त्यामध्ये एक्सेस करण्यासाठी तुम्ही सिंगल लॉग ऑन वापरू शकता. प्रत्येक सर्व्हरवर लॉग इन करण्याऐवजी किंवा प्रत्येक प्रिंटर किंवा इतर इक्विपमेंट्स वर प्रमाणीकरण करण्याऐवजी, डिरेक्टरी ही इनफार्मेशन असते. तुम्ही एकदा लॉग ऑन करा आणि तुम्हाला दिलेल्या अधिकारांद्वारे नेटवर्क प्रशासकाने परवानगी दिली असेल ते तुम्ही करू शकता.

**3 सुरक्षा :** फक्त एक डोमेन असणे म्हणजे सिंगल सुरक्षा धोरण आणि प्रशासकांच्या एका संचाद्वारे चांगली सुरक्षा. तुमच्याकडे मल्टिपल डोमेन आणि फॉरेस्ट असल्यास, प्रत्येकाचा स्वतःचा एडमिनिस्ट्रेटर असतो. एक कमकुवत परंतु विश्वसनीय डोमेन इतर सर्व जंगले आणि डोमेन उघड करते. केवळ एका डोमेनसह, संस्था-व्यापी सुरक्षा धोरण लागू करणे देखील खूप सोपे आहे

**4 सिंगल प्लॅटफॉर्म :** सिंगल डिरेक्टरी सर्विस किंवा ग्लोबल कॅटलॉग (GC) म्हणजे मॉनिटरिंग आणि मेसेजिंगसह इतर सर्व डिरेक्टरी -वेअर सर्विस साठी एकच प्लॅटफॉर्म.

**5 जलद डिप्लॉयमेंट :** फक्त एकाच डोमेनसह संस्थेमध्ये सुरू होते आणि शेअर केलेले खाते डेटाबेस सोल्यूशन्स फक्त एकदाच डिप्लॉयमेंट केले जाणे आवश्यक आहे, याचा अर्थ संस्थेकडे मल्टिपल आणि स्वतंत्र डोमेन असल्यास त्यापेक्षा कंपनी व्यापक सबयोजन खूप जलद आहे.

**6 सिंगल मॅनेजमेंट :** पायाभूत सुविधा- सिंगल मॅनेजमेंट पायाभूत सुविधा असणे म्हणजे इतर सर्व डिरेक्टरी सर्विस कार्यासाठी फक्त एकच पायाभूत सुविधा आहे, जसे की सॉफ्टवेअर सबयोजन, इन्व्हेंटर्री, आणि ऑब्जेक्ट मॅनेजमेंट शेअरिंग आणि डेलिगेशन (जसे की युजर खात्यांसाठी).

**7 सिंगल ग्रुप पॉलिसी कंटेनर (GPC):** सिंगल GPC सह, मॅनेजमेंट धोरणे फक्त एकदाच परिभाषित करणे आवश्यक आहे आणि समूह

धोरण ऑब्जेक्ट्स (GPOs) मॅन्युअली निर्यात आणि आयात न करता संपूर्ण एंटरप्राइझमध्ये वापरले जाऊ शकते.

**8 बॅकअप आणि पुनर्प्राप्ती:** फक्त एकच डोमेन असणे म्हणजे उत्तम लवचिकता आहे कारण प्रत्येक स्थानाचा पूर्ण डोमेन बॅकअप असतो.

**9 कमी हार्डवेअर:** मल्टिपल डोमेन असलेल्या संस्थेमध्ये, प्रत्येक स्थानासाठी दोन डोमेन नियंत्रकांची (DCs) आवश्यकता असते. एकाच डोमेनसह, प्रत्येक स्थानासाठी फक्त एकच DC आवश्यक आहे कारण लोकल DC अयशस्वी झाल्यास, स्थाने हब डीसी वापरू शकतात. कमी केलेले हार्डवेअर म्हणजे कमी परवाने, कमी मॅनेजमेंट सॉफ्टवेअर आणि सर्व्हर मॅनेजमेंट साठी कमी ओव्हरहेड. रिमोट डीसीचा बॅकअप घेण्याचीही गरज नाही कारण रिमोट डीसी फक्त सेंट्रल डीसी सारखीच इनफार्मेशन धारण करतात - असे गृहीत धरून की डीसी फक्त डिरेक्टरी सर्विस करतात.

### डिरेक्टरी सर्विस ची अंमलबजावणी

डिरेक्टरी सर्विस या ओपन सिस्टिम इंटरकनेक्शन (OSI) सबक्रमाचा पार्ट होत्या ज्यायोगे उद्योगातील प्रत्येकाने मल्टी व्हॅंडर इंटरऑपरेबिलिटी प्रदान करण्यासाठी समान नेटवर्क मानकांशी सहमत व्हावे. 1980 च्या दशकात, इंटरनेशनल टेलिकम्युनिकेशन युनियन (ITU) आणि इंटरनेशनल ऑर्गनायझेशन फॉर स्टॅंडर्डायझेशन (ISO) ने निर्देशांक सर्विस साठी मानकांचा एक संच - X.500 आणला. लाइट वेट डायरेक्टरी एक्सेस प्रोटोकॉल, LDAP, जे X.500 च्या डिरेक्टरी इनफार्मेशन सर्विस वर आधारित आहे, परंतु TCP/IP स्टॅक आणि X.500 प्रोटोकॉल DAP ची स्ट्रिंग एन्कोडिंग योजना वापरते.

### LDAP/X.500 आधारित अंमलबजावणीमध्ये हे आहेत:

- चालू डिरेक्टरी:** Windows साठी मायक्रोसॉफ्टची आधुनिक डिरेक्टरी सर्विस , X.500 डिरेक्टरीमधून उद्भवलेली, एक्सचेंज सर्व्हरमध्ये वापरण्यासाठी तयार केलेली, प्रथम Windows 2000 सर्व्हरसह पाठविली गेली आणि Windows च्या लागोपाठ आवृत्त्यांद्वारे समर्थित आहे.
- अपाचे डिरेक्टरी सर्व्हर:** LDAP, Kerberos 5 आणि चेंज पासवर्ड प्रोटोकॉलला सपोर्ट देणारी Java मध्ये लिहीलेली डिरेक्टरी सर्विस . LDAPv3 प्रमाणित. Apache Directory Server देखील Apache Software Foundation चा एक उच्चस्तरीय प्रकल्प आहे.
- eDirectory:** ही NetIQ ची डिरेक्टरी सर्विस ची अंमलबजावणी आहे. हे विंडोज, नेटवेअर, लिनक्स आणि युनिक्सच्या अनेक प्लेवर्ससह अनेक आर्किटेक्चर्सना सपोर्ट देते आणि बर्बाद काळापासून युजर एडमिनिस्ट्रेशन, कॉन्फिगरेशन मॅनेजमेंट आणि सॉफ्टवेअर मॅनेजमेंट साठी वापरले जात आहे. eDirectory ओळख मॅनेजमेंट उत्पादनांच्या विस्तृत रेन्ज त मध्यवर्ती एलिमेंट म्हणून विकसित झाली आहे. हे पूर्वी नोव्हेल डिरेक्टरी सर्व्हीसेस म्हणून ओळखले जात असे.
- Red Hat डिरेक्टरी सर्व्हर:** Red Hat ने एक डिरेक्टरी सर्विस जारी केली, जी ती AOL च्या नेटस्केप सिक्युरिटी सोल्युशन्स युनिटकडून विकत घेतली, Red Hat Enterprise Linux वर Red Hat Directory

Server नावाचे व्यावसायिक उत्पादन म्हणून आणि समुदायाने 389 डिरेक्टरी सर्व्हर प्रकल्पाला सपोर्ट दिले.

- **ओरॅकल इंटरनेट डिरेक्टरी** : (OID) ही ओरॅकल कॉर्पोरेशनची डिरेक्टरी सर्विस आहे, जी LDAP आवृत्ती 3 शी सुसंगत आहे.
- **सन जावा सिस्टम डिरेक्टरी सर्व्हर**: सन मायक्रोसिस्टम्सची करंट डिरेक्टरी सर्विस ऑफर.
- **OpenDS**: सन मायक्रोसिस्टम्स द्वारे समर्थित, Java मध्ये सुरवातीपासून एक ओपन सोर्स डिरेक्टरी सर्विस अंमलबजावणी.
- **IBM Tivoli Directory Server**: हे OpenLDAP च्या जुन्या रिलीजचे कस्टमाइज बिल्ड आहे.
- **विंडोज एनटी डिरेक्ट्री सर्व्हिसेस (एनटीडीएस)**, ज्याचे नंतर अॅक्टिव्ह डिरेक्ट्री असे नामकरण करण्यात आले, ती पूर्वीची एनटी डोमेन सिस्टिम बदलते.
- **OpenLDAP** : हे युनिक्स आणि युनिक्स डेरिव्हेटिव्ह्ज, लिनक्स, विंडोज, z/OS आणि विविध एम्बेडेड/रिअलटाइम सिस्टमसह सर्व करंट संगणक आर्किटेक्चरला सपोर्ट देते.

OpenLDAP आणि Kerberos प्रोटोकॉल, आणि Samba सॉफ्टवेअरसह डिरेक्टरी सर्विस तयार करण्यासाठी भरपूर ओपन-सोर्स साधने देखील आहेत.

## एॅक्टिव्ह डिरेक्टरी

- एॅक्टिव्ह डिरेक्टरी (AD) ही एक डिरेक्टरी सर्विस आहे जी Microsoft Windows डोमेन नेटवर्कसाठी विकसित केली आहे आणि बहुतेक Windows Server ऑपरेटिंग सिस्टममध्ये प्रक्रिया आणि सर्विस चा संच म्हणून समाविष्ट आहे.
- एक AD डोमेन कंट्रोलर सर्व वापरकर्ते आणि संगणकांना Windows डोमेन प्रकारातील नेटवर्क-सर्व संगणकांसाठी सुरक्षा धोरणे नियुक्त करणे आणि लागू करणे आणि सॉफ्टवेअर इंस्टाल करणे किंवा अपडेट करणे यासाठी प्रमाणित आणि अधिकृत करतो. उदाहरणार्थ, जेव्हा युजर Windows डोमेनचा पार्ट असलेल्या संगणकावर लॉग इन करतो, तेव्हा Active Directory सबमिट केलेला पासवर्ड तपासते आणि युजर सिस्टम एडमिनिस्ट्रेटर आहे की सामान्य युजर आहे हे ठरवते.[3]
- एॅक्टिव्ह डिरेक्टरी लाइटवेट डायरेक्ट्री एॅक्सेस प्रोटोकॉल (LDAP) आवृत्ती 2 आणि 3, Microsoft च्या Kerberos ची आवृत्ती आणि DNS चा वापर करते.
- एॅक्टिव्ह डिरेक्टरी देखील युजर मॅनेजमेंट सुलभ करते कारण ती या सर्व युजर साठी आणि संगणकाशी संबंधित इनफार्मेशन साठी सिंगल भांडार म्हणून कार्य करते.
- AD त्याचा एॅक्सेस प्रोटोकॉल म्हणून LDAP चा वापर करते.
- AD त्याची लोकेटर सर्विस म्हणून DNS वर अवलंबून असते, क्लायंट सक्षम करते DNS केरीद्वारे डोमेन कंट्रोलर सर्च प्यासाठी.

## एॅक्टिव्ह डिरेक्टरी ची लॉजिकल रचना

Active Directory हा एक वितरित डेटाबेस आहे जो नेटवर्क संसाधनाविषयी इनफार्मेशन स्टोर करतो आणि मॅनेज करतो, तसेच डिरेक्टरी -सक्षम एॅप्लिकेशन्समधील एॅप्लिकेशन -स्पेसिफिक डेटा.

एॅक्टिव्ह डिरेक्ट्री प्रशासकांना नेटवर्कचे एलिमेंट (जसे की वापरकर्ते, संगणक, उपकरणे इत्यादी) रेन्ज बद्ध (हायरार्कीकल) कंटेनमेंट स्ट्रक्चरमध्ये मॅनेज करण्यास अनुमती देते.

एॅक्टिव्ह डिरेक्ट्रीमध्ये, रिसोर्सेस लॉजिकल संरचनेत आयोजित केली जातात आणि संसाधनांचे हे गटीकरण लॉजिकल दृष्ट्या संसाधनाच्या फिजिकल स्थानाऐवजी त्याच्या नावाने सर्च प्यास सक्षम करते.

## एडी लॉजिकल स्ट्रक्चरचे फायदे

- लॉजिकल स्ट्रक्चर केवळ स्पेसिफाइड गटांना (OU) संसाधनांमध्ये एॅक्सेस प्रदान करून अधिक नेटवर्क सुरक्षा प्रदान करते.
- लॉजिकल संरचनेने नेटवर्कचे एडमिनिस्ट्रेशन, कॉन्फिगरेशन आणि नियंत्रणाद्वारे नेटवर्क मॅनेजमेंट सुलभ केले.
- डोमेन आणि जंगलांच्या लॉजिकल संरचनेतील संबंध संस्थेमध्ये संसाधनांची वाटणी सुलभ करते.
- लॉजिकल रचना सरलीकृत नेटवर्क मॅनेजमेंट प्रदान करते म्हणून, ते नेटवर्क संसाधनावरील भार कमी करते आणि मालकीची एकूण किंमत कमी करते.

## एडी लॉजिकल स्ट्रक्चरचे एलिमेंट

लॉजिकल रचना एलिमेंटचा एकमेकांशी संबंध असतो म्हणून ते स्टोर डेटावर एॅक्सेस कंट्रोल करण्यास मॅनेज करते आणि जंगलातील भिन्न डोमेन दरम्यान डेटा कसा मॅनेज केला जाईल ते सर्च ते.

- ऑब्जेक्ट्स: जसे की युजर, संगणक, ग्रुप, प्रिंटर इ...
- संस्थात्मक एकके - कोणत्याही फोल्डरप्रमाणे परंतु एॅक्टिव्ह डिरेक्ट्रीच्या नियंत्रणात
- डोमेन - ऑब्जेक्ट्ससाठी लॉजिकल बॉर्डर
- वृक्ष - मल्टिपल डोमेनसाठी लॉजिकल बॉर्डर
- जंगले - अनेक झाडांसाठी लॉजिकल बॉर्डर

एकंदरीत, मायक्रोसॉफ्ट डोमेन कंट्रोलर म्हणून चालणारे एक फिजिकल मशीन स्पेसिफिक कार्य करण्यासाठी समर्पित 'ए ऑपरेशन मास्टर' च्या मदतीने या सर्व लॉजिकल विभागांना कंट्रोल करू शकते.

वरच्या लेयर वरील कंटेनर जंगल आहे. जंगल हा वृक्षांचा संग्रह आहे ज्यात एक सामान्य ग्लोबल कॅटलॉग, डिरेक्टरी स्कीमा, लॉजिकल रचना आणि डिरेक्टरी कॉन्फिगरेशन सामायिक केले जाते. जंगल सुरक्षा बॉर्डर दर्शवते ज्यामध्ये वापरकर्ते, संगणक, ग्रुप आणि इतर वस्तू एॅक्सेस योग्य आहेत.

जंगलांमध्ये डोमेन आहेत. डोमेन हे नेटवर्क ऑब्जेक्ट्स (संगणक, वापरकर्ते, डिव्हाइसेस) च्या लॉजिकल ग्रुप म्हणून परिभाषित केले जाते जे समान एॅक्टिव्ह डिरेक्टरी डेटाबेस सामायिक करतात.

डोमेनमध्ये संस्थात्मक एकेके आहेत. OU डोमेनला पदानुक्रम प्रदान करू शकतात, त्याचे एडमिनिस्ट्रेशन सुलभ करू शकतात आणि व्यवस्थापकीय किंवा भौगोलिक अटींमध्ये संस्थेच्या संरचनेसारखे असू शकतात. OUs मध्ये इतर OU समाविष्ट असू शकतात-डोमेन हे कंटेनर आहेत

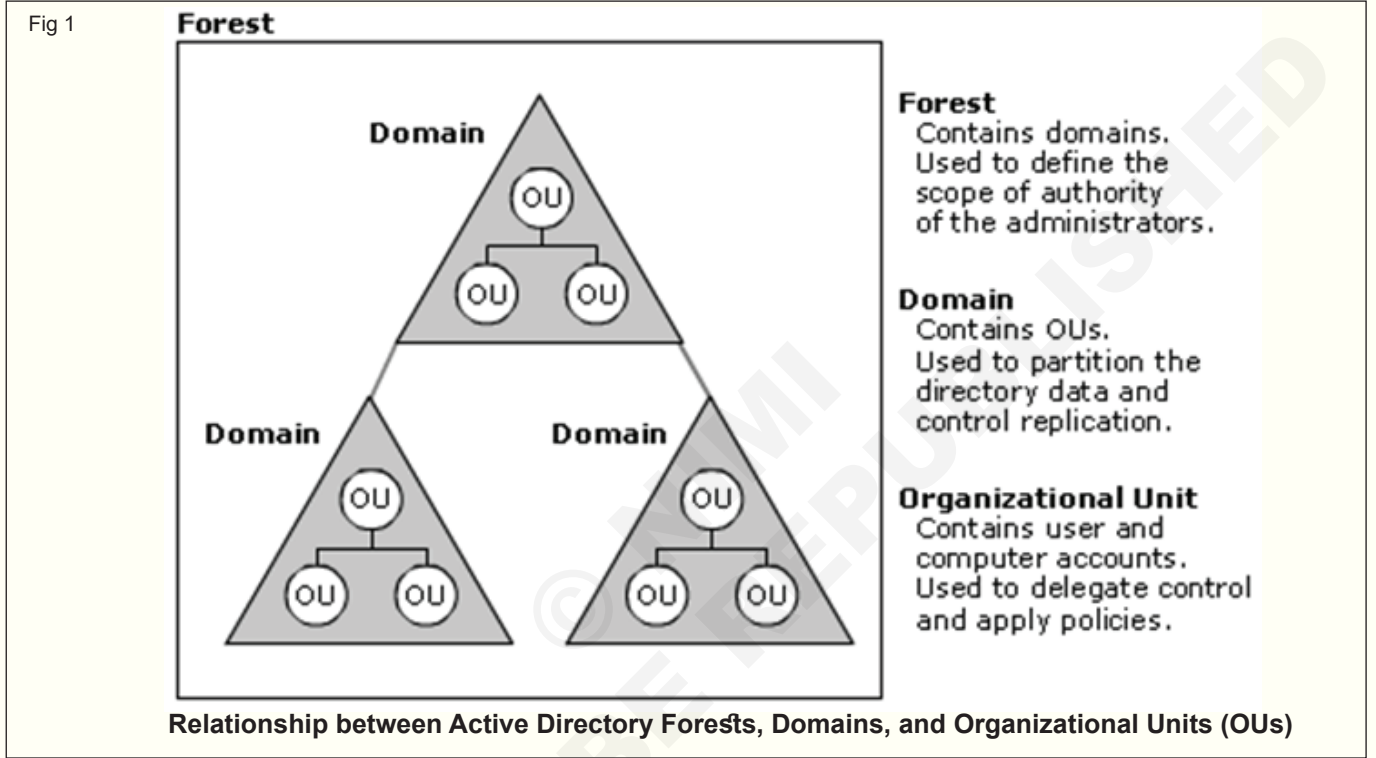
अर्थ OU ही अशी पातळी आहे ज्यावर अडमिनिस्ट्रेटिव्ह अधिकार सामान्यतः सोपवले जातात, परंतु वैयक्तिक वस्तू किंवा गुणधर्मांवर देखील सोपवले जाऊ शकते.

याला लॉजिकल मॉडेल असे म्हणतात कारण ते सबयोजनाच्या फिजिकल पैलूंपासून स्वतंत्र आहे, जसे की प्रत्येक डोमेन आणि नेटवर्क टोपोलॉजीमध्ये आवश्यक डोमेन नियंत्रकांची नंबर .

आकृती 1 जंगले, डोमेन आणि संस्थात्मक एकेके यांच्यातील संबंध दर्शविते. आकृती 1 ऍक्टिव्ह डिरेक्टरी फॉरेस्ट, डोमेन आणि ऑर्गनायझेशनल युनिट्स (OUs) यांच्यातील संबंध

### ऍक्टिव्ह डिरेक्टरी ची फिजिकल रचना

ऍक्टिव्ह डिरेक्टरी फिजिकल रचना केव्हा आणि कुठे लॉगऑन आणि प्रतिकृती रहदारी येते हे तपासते. ऍक्टिव्ह डिरेक्टरीच्या फिजिकल संरचनेमध्ये नेटवर्कमध्ये सबस्थित असलेले सर्व फिजिकल सबनेट जसे की डोमेन कंट्रोलर आणि डोमेन नियंत्रकांमधील प्रतिकृती समाविष्ट आहे.



### ऍक्टिव्ह डिरेक्टरी ची फिजिकल रचना:

- **डोमेन कंट्रोलर** : हे संगणक Microsoft Windows Server 2003/2000, आणि Active Directory चालवतात. प्रत्येक डोमेन कंट्रोलर प्रतिकृती, संचयन आणि प्रमाणीकरण यासारखी स्पेसिफिक फंक्शन करतो. हे जास्तीत जास्त एका डोमेनला सपोर्ट करू शकते. प्रत्येक डोमेनमध्ये एकापेक्षा जास्त डोमेन कंट्रोलर असणे नेहमीच सुचवले जाते.
- **ऍक्टिव्ह डिरेक्टरी साइट्स**: या साइट्स चांगल्या-कनेक्ट केलेल्या संगणकांचा संग्रह आहेत. आम्ही साइट का तयार करतो याचे कारण म्हणजे डोमेन कंट्रोलर साइटवर वारंवार संवाद साधू शकतात. अशा प्रकारे ते एका डोमेन कंट्रोलरवर केलेले बदल इतर डोमेन कंट्रोलरवर प्रतिकृति करण्यासाठी साइटमधील विलंब कमी करते. साइट तयार करण्यामागील दुसरे कारण म्हणजे डोमेन कंट्रोलर्स दरम्यान बँडविड्थ ऑप्टिमाइझ करणे वेगवेगळ्या ठिकाणी स्थित.

कॉमन लोकल एरिया नेटवर्क (LAN) कनेक्टिव्हिटी सामायिक करणार्या सर्व IP सबनेट्सना संगणकाचे वास्तविक फिजिकल पोजिशन माहित

नसताना साइट म्हणतात.

ग्लोबल कॅटलॉग हा डेटा स्टोरेज सोर्स आहे ज्यामध्ये मल्टीडोमेनमध्ये आढळलेल्या वस्तूंचे आंशिक प्रतिनिधित्व असते ऍक्टिव्ह डिरेक्टरी डोमेन सर्विस (ADDS) जंगल. ग्लोबल कॅटलॉग स्पेशल तः ग्लोबल म्हणून नियुक्त केलेल्या डोमेन नियंत्रकांवर स्टोर केला जातो कॅटलॉग सर्व्हर. हे वास्तविक डोमेन नाव जाणून घेतल्याशिवाय कोणत्याही डोमेनमधील ऑब्जेक्ट्स शोधू शकते.

ग्लोबल कॅटलॉगकडे निर्देशित केलेले सर्च वेगवान आहेत कारण त्यामध्ये भिन्न डोमेन नियंत्रकांचे रेफरन्स समाविष्ट नसतात.

### OU मध्ये रिसोर्सेस आयोजित करणे

युजर, संगणक आणि इतर ऑब्जेक्ट इनफार्मेशन अधिक सहजपणे समजण्यायोग्य लेआउटमध्ये आयोजित करण्यासाठी OU ही प्रायमरी मेथड्स आहे. संस्थेचे मूळ संडव्हेट्स त्मक एकक आहे जेथे तीन नेस्टेड संडव्हेट्स त्मक युनिट्स ठेवल्या आहेत. हे नेस्टिंग संस्थेला नेटवर्क रिसोर्सेस सुलभपणे पाहण्यासाठी आणि प्रशासनासाठी मल्टिपल कंटेनरमध्ये वापरकर्त्यांना वितरित करण्यास सक्षम करते.

प्रशासनाच्या सोप्या संघटनेसाठी आणि सोप्या मेथड्स ने OUs ची संसाधन OU मध्ये विपार्ट णी केली जाऊ शकते. दूरवरच्या कार्यालयांमध्ये लोकल प्रशासनासाठी स्वतःचे ओयू असू शकतात. तथापि, हे समजून घेणे महत्वाचे आहे की, जर संस्थेला प्रशासकांच्या दुसऱ्या संचाला एडमिनिस्ट्रेशन सोपवण्याची स्पेसिफिक गरज असेल तरच एक OU तयार केले जावे. जर एकच व्यक्ती किंवा लोकांचा समूह संपूर्ण डोमेन प्रशासित करत असेल तर, OU जोडून पर्यावरणाची जटिलता वाढवण्याची गरज नाही. खरं तर, खूप जास्त OU ग्रुप धोरणे, लॉगऑन आणि इतर एलिमेंटवर रिझल्ट करू शकतात.

OU ची रचना स्वतंत्र विभागांना त्यांच्या स्वतः च्या वापरकर्त्यांवर विविध स्तरांचे अडमिनिस्ट्रेटिव्ह कंट्रोल ठेवण्याची परवानगी देण्यासाठी केली जाऊ शकते. उदाहरणार्थ, अभियांत्रिकी विभागातील सचिवाला त्याच्या स्वतःच्या OU मधील वापरकर्त्यांसाठी पासवर्ड रीसेट करण्याचे कंट्रोल सोपवले जाऊ शकते. या परिस्थितीत OU वापरण्याचा आणखी एक फायदा असा आहे की वापरकर्त्यांना सहजपणे ड्रॅग केले जाऊ शकते आणि एका OU मधून दुसऱ्यावर सोडले जाऊ शकते. उदाहरणार्थ, वापरकर्त्यांना एका विभागातून दुसऱ्या विभागात हलवले असल्यास, त्यांना त्यांच्या नवीन विभागाच्या OU मध्ये हलवणे अत्यंत सोपे आहे.

हे लक्षात ठेवणे महत्वाचे आहे की जेव्हा प्रशासकाला स्ट्रक्चर त्मक बदल करण्यास योग्य वाटेल तेव्हा OU स्ट्रक्चर बदलता येऊ शकते. हे ऍक्टिव्ह डिरेक्ट्रीला कधीही बदल करण्याचा अतिरिक्त फायदा देते.

ग्रुप पॉलिसी ऑपरेटिंग सिस्टीमच्या Microsoft Windows NT कुटुंबाचे वैशिष्ट्य आहे जे युजर खाती आणि संगणक खात्यांचे कार्य वातावरण कंट्रोल करते. ग्रुप धोरण ऍक्टिव्ह डिरेक्टरी वातावरणात ऑपरेटिंग सिस्टम, ऍप्लिकेशन्स आणि वापरकर्त्यांच्या सेटिंग्जचे सेंट्रलाइज्ड मॅनेजमेंट आणि कॉन्फिगरेशन प्रदान करते. लोकल ग्रुप पॉलिसी ("LGPO" किंवा "Lo-

calGPO") नावाची ग्रुप पॉलिसीची आवृत्ती स्टँडअलोन आणि नॉन-डोमेन कॉम्प्युटरवर ग्रुप पॉलिसी ऑब्जेक्ट मॅनेजमेंटला देखील परवानगी देते. ग्रुप धोरण हे ऍक्टिव्ह डिरेक्टरी डिप्लॉयमेंट करण्याचे प्रमुख कारण आहे कारण ते तुम्हाला युजर आणि संगणक वस्तू मॅनेज करण्यास अनुमती देते.

ग्रुप धोरण, अंशतः, वापरकर्ते संगणक सिस्टिम वर काय करू शकतात आणि काय करू शकत नाहीत हे कंट्रोल करते, उदाहरणार्थ: पासवर्ड जटिलता धोरण लागू करणे जे वापरकर्त्यांना अती साधा पासवर्ड निवडण्यापासून प्रतिबंधित करते, अज्ञात वापरकर्त्यांना परवानगी देणे किंवा प्रतिबंधित करणे.

नेटवर्क शेअरशी कनेक्ट करण्यासाठी, विंडोज टास्क मॅनेजरमध्ये एक्सेस अवरोधित करण्यासाठी किंवा स्पेसिफिक फोल्डर्समध्ये एक्सेस प्रतिबंधित करण्यासाठी दूरस्थ संगणक. अशा कॉन्फिगरेशनच्या संचाला ग्रुप पॉलिसी ऑब्जेक्ट (GPO) म्हणतात.

ऍक्टिव्ह डिरेक्टरी बॅकअप आणि रिस्टोर करा

ऍक्टिव्ह डिरेक्टरी हा तुमच्या इन्फ्रास्ट्रक्चरमधील सर्वात महत्वाचा एलिमेंट आहे. ते खाली गेल्यास, तुमचे नेटवर्क निरुपयोगी रेंडर केले जाते. त्यामुळे, व्यवसायातील सातत्य आणि अनुपालन सुनिश्चित करण्यासाठी, तुमच्याकडे ऍक्टिव्ह डिरेक्टरीसाठी एक ठोस बॅकअप आणि पुनर्प्राप्ती योजना असणे आवश्यक आहे.

जेथे समूह धोरण प्राधान्य सेटिंग्ज कॉन्फिगर केली आहेत आणि समतुल्य ग्रुप धोरण सेटिंग्ज देखील कॉन्फिगर केली आहे, तेव्हा ग्रुप धोरण सेटिंग्जच्या मूल्याला प्राधान्य दिले जाईल. समूह धोरण म्हणजे डोमेनची सुरक्षा. बॅकअप युटिलिटी ऑटोमॅटिक पणे शोधून काढेल आणि जेव्हा तुम्ही सिस्टम स्थितीचा बॅकअप घेता तेव्हा त्यांचा समावेश होईल.

## एक्सेस कंट्रोल , ऑडिट आणि टेस्टिंग (Access Control, Audit and testing)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एक्सेस नियंत्रणातील उद्देश आणि स्टेप्स चे डिस्क्रिप्शन करा
- एक्सेस नियंत्रणातील विविध स्तरांचे डिस्क्रिप्शन करा
- एक्सेस कंट्रोल यंत्रणेचे डिस्क्रिप्शन करा
- सुरक्षिततेसाठी इनफार्मेशन चे वर्गीकरण करा
- एक्सेस कंट्रोल प्रोटोकॉलची यादी करा
- सुरक्षा ऑडिटचे डिस्क्रिप्शन करा
- सुरक्षा लेखापरीक्षांच्या महत्वाचे डिस्क्रिप्शन करा
- ऑडिट सुरक्षा पार पाडण्याच्या प्रक्रियेचे डिस्क्रिप्शन करा
- असुरक्षितता असेसमेंट आणि एक्सेस टेस्टिंग चे डिस्क्रिप्शन करा
- विविध टाइप च्या सुरक्षा ऑडिट साधनांचा उल्लेख करा.

**परिचय:** संरक्षित इनफार्मेशन चा एक्सेस हा इनफार्मेशन वर एक्सेस करण्यासाठी अधिकृत असलेल्या लोकांसाठी प्रतिबंधित केला पाहिजे. एक्सेस नियंत्रणाचा उद्देश संगणक सिस्टिम चा कायदेशीर युजर करू शकणार्या क्रिया किंवा ऑपरेशन्स तसेच वापरकर्त्यांच्या वतीने कार्यान्वित करणार्या प्रोग्राम्सना काय करण्याची परवानगी आहे यावर मर्यादा घालणे

हा आहे. अशा प्रकारे एक्सेस कंट्रोल सुरक्षेचा भंग होऊ शकणार्या क्रियाकलापांना प्रतिबंधित करण्याचा प्रयत्न करते.

ऍक्सेस कंट्रोल मेकॅनिझमची परिष्कृतता संरक्षित केलेल्या इनफार्मेशन च्या मूल्याच्या प्रमाणात असावी - इनफार्मेशन जितकी अधिक सेन्सिटिव्ह

किंवा मौल्यवान असेल तितकी कंट्रोल यंत्रणा मजबूत असणे आवश्यक आहे. ज्या पायावर एक्सेस कंट्रोल यंत्रणा तयार केली जाते ते ओळख आणि प्रमाणीकरणाने सुरू होते.

एक्सेस कंट्रोल सामान्यतः तीन स्टेप्स मध्ये प्राप्त केले जाते: ओळख, प्रमाणीकरण आणि अधिकृतता.

- ओळख म्हणजे कोणी कोण आहे किंवा काहीतरी काय आहे याचे प्रतिपादन आहे. युजर ते युजर नाव एंटर करून एंटर करून स्वतःला ओळखतो.
- प्रमाणीकरण ही यूजर ची ओळख सत्यापित करण्याची प्रक्रिया आहे. प्रमाणीकरण हे इनफार्मेशन आश्वासनाच्या (IA) पाच स्तंभांपैकी एक आहे. इतर चार म्हणजे अखंडता, उपलब्धता, प्रायव्हसी आणि अप्रत्यक्षता.

संकेतशब्द प्रविष्ट करणे ही प्रमाणीकरणाची एक सामान्य मेथड्स आहे. परंतु पासवर्ड-आधारित प्रमाणीकरण संगणक नेटवर्कवर वापरण्यासाठी योग्य नाही. नेटवर्कवर पाठवलेले पासवर्ड रोखले जाऊ शकतात आणि नंतर हॅकर्सद्वारे त्याचा गैरवापर केला जाऊ शकतो. सुरक्षिततेच्या चिंतेव्यतिरिक्त, पासवर्ड आधारित प्रमाणीकरण युजर म्हणून गैरसोयीचे आहे प्रत्येक वेळी नेटवर्क सेवेत एक्सेस करायचा असेल तेव्हा पासवर्ड टाकावा लागेल. महत्त्वपूर्ण असलेल्या व्यवहारांसाठी (जसे की पैशांची देवाणघेवाण) या सिस्टिम तील कमकुवतपणा म्हणजे पासवर्ड अनेकदा चोरीला जाऊ शकतो, चुकून उघड होऊ शकतो किंवा विसरला जाऊ शकतो. अधिकृतता: एखादी व्यक्ती, प्रोग्राम किंवा संगणक यशस्वीरित्या ओळखल्यानंतर

आणि प्रमाणीकृत केल्यानंतर त्यांना कोणत्या इनफार्मेशन च्या संसाधनांमध्ये एक्सेस करण्याची परवानगी आहे आणि त्यांना कोणत्या क्रिया करण्याची परवानगी दिली जाईल (चालवा, पहा, तयार करा, डिलेट करा किंवा बदला) हे निर्धारित केले पाहिजे. याला अधिकृतता म्हणतात. इनफार्मेशन आणि इतर संगणकीय सर्विस मध्ये एक्सेस करण्यासाठी अधिकृतता अडमिनिस्ट्रेटिव्ह धोरणे आणि प्रक्रियांपासून सुरू होते. धोरणे विहित करतात की कोणती इनफार्मेशन आणि संगणकीय सर्विस, कोणाद्वारे आणि कोणत्या परिस्थितीत एक्सेस करता येईल.

### प्रमाणीकरण मेथड्स

- इव्हस्ट्रॉपिंगचा धोका प्रमाणीकरणासाठी डायजेस्ट वापरून मॅनेज केला जाऊ शकतो. कनेक्टिंग पार्टी एक व्हॅल्यु पाठवते, स्पेशल तः क्लायंटचा IP एड्रेस, टाइम स्टॅम्प आणि अतिरिक्त गुप्त इनफार्मेशन चा हॅश. प्रत्येक एक्सेस केलेल्या URI साठी हा हॅश अद्वितीय असल्यामुळे, इतर कोणत्याही डॉक्युमेंट्स मध्ये एक्सेस केला जाऊ शकत नाही किंवा तो सर्च ल्याशिवाय इतर IP पत्त्यावरून वापरला जाऊ शकत नाही. हॅशिंगमुळे पासवर्ड देखील इव्हस्ट्रॉपिंगसाठी असुरक्षित नाही. तथापि, -मॅन-इन-द मिडल अटॅक सारख्या ऍक्टिव्ह हल्ल्यांसाठी सिस्टिम असुरक्षित आहे.
- वन-टाइम पासवर्ड: पासवर्ड पुनर्वापराशी संबंधित समस्या टाळण्यासाठी, वन-टाइम पासवर्ड विकसित केले गेले. एक-वेळ पासवर्डचे दोन टाइप आहेत, एक आव्हान-प्रतिसाद संकेतशब्द आणि पासवर्ड लिस्ट.

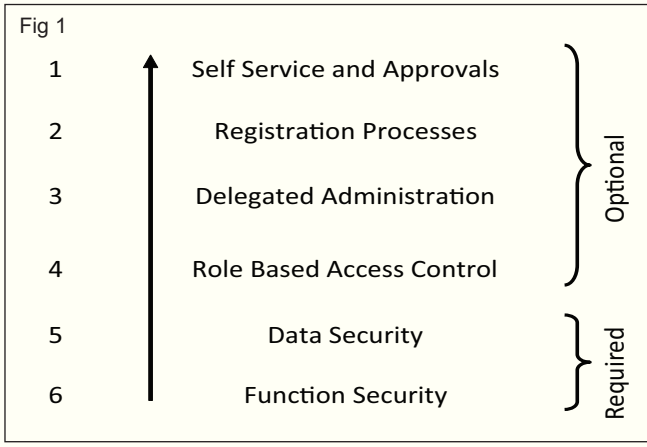
- सार्वजनिक-की क्रिप्टोग्राफी: सार्वजनिक की क्रिप्टोग्राफी ही अत्यंत क्लिष्ट गणिती समस्यांवर आधारित आहे ज्यांना अतिशय स्पेशल ज्ञान आवश्यक आहे. सार्वजनिक की क्रिप्टोग्राफी दोन की वापरते, एक प्राइवेट आणि दुसरी सार्वजनिक. अत्यंत गुंतागुंतीच्या गणितीय समीकरणाद्वारे दोन कळा एकमेकांशी जोडलेल्या आहेत. प्रायव्हेंट की चा वापर कम्युनिकेशन मशीनमधील संदेश डिक्रिप्ट करण्यासाठी आणि कूटबद्ध करण्यासाठी केला जातो. एन्क्रिप्शन आणि स्वाक्षरीचे सत्यापन दोन्ही सार्वजनिक की सह पूर्ण केले जाते.
- शून्य-ज्ञान पुरावे: शून्य-ज्ञान पुरावे यजमानाला कोणतीही "गुप्त इनफार्मेशन" उघड न करता एक्सेस करण्यास परवानगी देण्यासाठी दुसऱ्या होस्टला पटवणे शक्य करतात. प्रमाणीकरणाच्या या स्वरूपामध्ये सामील असलेले होस्ट सहसा प्रमाणीकरण अंतिम करण्यासाठी अनेक वेळा कम्युनिकेशन करतात.
- सार्वजनिक की इन्फ्रास्ट्रक्चरचा पार्ट म्हणून प्रमाणपत्र प्राधिकरण (CA) द्वारे जारी केलेल्या आणि सत्यापित केलेल्या डिजिटल प्रमाणपत्रांचा वापर हा इंटरनेटवर प्रमाणीकरण करण्याचा आणखी एक स्टॅण्डर्ड पाथ मानला जातो.
- प्रमाणीकरणाची दुसरी मेथड्स, बायोमेट्रिक्स, यूजर ची सबस्थिती आणि जैविक मेकअप (म्हणजे डोळ्यातील पडदा किंवा बोटांचे ठसे) यावर अवलंबून असते. या तंत्रज्ञानामुळे हॅकर्सना संगणक सिस्टिम मध्ये एक्सेस करणे अधिक कठीण होते. साठी स्पेशल प्रोटोकॉल वापरणे प्रमाणीकरण जसे की सुरक्षित सॉकेट लेयर (SSL), IP SEC, Secure Shell (SSH), Kerberos प्रमाणीकरण आणि एक्स्टेंसिबल ऑथेंटिकेशन प्रोटोकॉल (EAP) इ.

सुरक्षा एडमिनिस्ट्रेटर संस्थेच्या सुरक्षा धोरणावर आधारित अधिकृततेचा डेटाबेस ठेवतो. ऑपरेशन करण्याचा प्रयत्न करणारा युजर प्रत्यक्षात ते ऑपरेशन करण्यासाठी अधिकृत आहे की नाही हे निर्धारित करण्यासाठी रेफरेन्स मॉनिटर अधिकृतता डेटाबेसचा सल्ला घेतो. ऑडिटिंग सिस्टिममधील संबंधित क्रियाकलापांचे निरीक्षण करते आणि रेकॉर्ड ठेवते.

प्रमाणीकरण आणि एक्सेस कंट्रोल यामध्ये स्पष्ट फरक करणे महत्त्वाचे आहे. यूजर ची ओळख अचूकपणे इंस्टाल करणे ही प्रमाणीकरण सेवेची जबाबदारी आहे. एक्सेस कंट्रोल हे गृहीत धरते की रेफरेन्स मॉनिटरद्वारे एक्सेस नियंत्रणाची अंमलबजावणी करण्यापूर्वी यूजर चे प्रमाणीकरण यशस्वीरित्या सत्यापित केले गेले आहे.

एक्सेस नियंत्रणाचे सक्सेसिव्ह लेयर

एक्सेस कंट्रोल सक्सेसिव्ह लेयर मध्ये लागू केले जाते आणि प्रत्येक स्तर त्याच्या आधीच्या लेयर वर तयार होतो. संस्था, वैकल्पिकरित्या, विद्यमान फंक्शन आणि डेटा सिव्युरिटी मॉडेल्सवर तयार करू इच्छित असलेल्या ऑटोमेशन आणि स्केलेबिलिटीच्या प्रमाणात अवलंबून विविध स्तर घेऊ शकतात. याची अंमलबजावणी करण्यासाठी विविध मॉडेल्स असू शकतात. उदाहरण म्हणून, ऑरॅकल युजर मॅनेजमेंटमध्ये एक्सेस नियंत्रणाचे सहा स्तर आहेत. (पिव्चर 1 पहा) कोर सुरक्षा स्तरांमध्ये हे समाविष्ट आहे:



- कार्य सुरक्षा
- डेटा सुरक्षा

पुढील चार स्तर ओरॅकल युजर मॅनेज्मेंट चा पार्ट आहेत:

- रोल बेस्ड एक्सेस कंट्रोल
- नियुक्त एडमिनिस्ट्रेशन
- नोंदणी प्रक्रिया
- सेल्फ सर्विस आणि मंजूरी

सर्वसाधारणपणे, ऍक्सेस कंट्रोल हे बेसिक सिस्टम एडमिनिस्ट्रेशन कार्यापासून सुरू होते, अधिक वितरित, प्रशासनाच्या लोकल पद्धतींमध्ये प्रगती करते आणि शेवटी वापरकर्त्यांना काही बेसिक, पूर्वनिर्धारित नोंदणी फंक्शन स्वतःच करण्यास सक्षम करते. चार्ट 1. थर एकमेकांवर कसे तयार होतात हे स्पष्ट करते.

सुरक्षा आणि डेटा सुरक्षा यंत्रणा सुरक्षा सिस्टम चा आधारभूत स्तर बनवतात आणि त्यामध्ये पारंपारिक सिस्टम अडमिनिस्ट्रिटिव्ह क्षमता असतात. ते स्पेसिफिक मेनूमध्ये एक्सेस देऊन युजर मॅनेज्मेंट ची व्याप्ती बेसिक सिस्टम प्रशासनापर्यंत मर्यादित करतात.

प्रवेशाचा लेयर	एडमिनिस्ट्रेशन ची पातळी
सेल्फ सर्विस आणि मंजूरी	एन्ड यूजर
नोंदणी प्रक्रिया	
एडमिनिस्ट्रेशन नियुक्त केले	लोकल एडमिनिस्ट्रेशन
भूमिका आधारित एक्सेस कंट्रोल	
डेटा सुरक्षा	सिस्टम प्रशासकाशी
कार्य सुरक्षा	

**लोकल एडमिनिस्ट्रटर :** जेव्हा डेटा सिक्युरिटी आणि फंक्शन सिक्युरिटी लेयर्समध्ये रोल-बेस्ड ऍक्सेस कंट्रोल आणि डेलिगेटेड अॅडमिनिस्ट्रेशन जोडले जातात, तेव्हा सिस्टम अॅडमिनिस्ट्रेशनची फंक्शन लोकल प्रशासकांना वितरित केली जाऊ शकतात जे संस्थेच्या वापरकर्त्यांचा सबसंच मॅनेज करतात.

**एन्ड यूजर :** नोंदणी प्रक्रिया आणि सेल्फ सर्विस आणि मंजूरी काही नोंदणी फंक्शन ऑटोमॅटिक करून सिस्टम प्रशासनाचे वितरण करतात जेणेकरून एन्ड यूजर ती करू शकतील. एन्ड यूजर नवीन युजर खाती मिळवणे, सिस्टममध्ये अतिरिक्त प्रवेशाची विनंती करणे आणि संकेतशब्द रीसेट करणे ही कामे करू शकतात.

**सेल्फ सर्विस आणि मंजूरी:** नोंदणी प्रक्रिया आवश्यकतेनुसार कॉन्फिगर केल्यानंतर, व्यक्ती नंतर स्वयं-सर्विस नोंदणी फंक्शन करू शकतात, जसे की नवीन युजर खाती मिळवणे किंवा सिस्टममध्ये अतिरिक्त प्रवेशाची विनंती करणे. याशिवाय, संस्था या विनंत्यांसाठी कस्टमाइज मंजूरी पाथ तयार करण्यासाठी ओरॅकल अप्रूव्हल्स मॅनेजमेंट इंजिन वापरू शकतात.

तीन टाइप चे प्रतिबंधात्मक कंट्रोल आहेत:

### अडमिनिस्ट्रिटिव्ह कंट्रोल

- **धोरणे/प्रक्रिया:** प्रक्रिया कोणत्या मार्गांनी पार पाडणे आवश्यक आहे हे ओळखण्यासाठी. हे प्रशिक्षण, गुप्तहेर कंट्रोल आणि ऑडिटसह हाताने जाणे आवश्यक आहे.

### फिजिकल कंट्रोल

- बायोमेट्रिक सेन्सर्स, स्मार्ट कार्ड इ. वापरणे.

### तांत्रिक (लॉजिकल) कंट्रोल

- एनक्रिप्शन
- पासवर्ड आणि टोकन
- बायोमेट्रिक्स
- O.S. आणि एप्लिकेशन कंट्रोल
- ओळख आणि अधिकृतता तंत्रज्ञान

### एक्सेस कंट्रोल यंत्रणा

एक्सेस नियंत्रणासाठी खालील मॉडेल/यंत्रणा आहेत. वरील प्रत्येक ऍक्सेस मॉडेलचे स्वतःचे फायदे आणि तोटे आहेत. संस्थेद्वारे योग्य ऍक्सेस मॉडेलची निवड व्यवसायाचा टाइप, वापरकर्त्यांची नंबर, संस्थेचे सुरक्षा धोरण इत्यादी विविध एलिमेंटचा विचार करून केली पाहिजे.

चार्ट 2

कंट्रोल सर्विस	डिस्ट्रिप्शन
प्रतिबंधात्मक	अनिष्ट गोष्टी घडण्यापासून दूर ठेवा
गुप्तहेर	घडलेल्या अनिष्ट गोष्टी ओळखा
सुधारक	घडलेल्या अनिष्ट गोष्टी दुरुस्त करा
ते रोखतात	सुरक्षा उल्लंघन होण्यापासून परावृत्त करा
पुनर्प्राप्ती	उल्लंघन किंवा अपघातानंतर रिसोर्स किंवा क्षमता रिस्टोअर करा
भरपाई	इतर नियंत्रणांना पर्याय प्रदान करा

## • रोल बेस्ड ऍक्सेस कंट्रोल (RBAC)

ऍक्सेस निर्णय एखाद्या व्यक्तीच्या संस्था किंवा यूजर बेसमधील भूमिका आणि जबाबदाऱ्यांवर आधारित असतात. RBAC ला नॉन-डिस्क्रिशनरी ऍक्सेस कंट्रोल म्हणूनही ओळखले जाते कारण यूजर ला त्याच्या भूमिकेशी जोडलेले विशेषाधिकार वारशाने मिळतात. यूजर चे त्याला नियुक्त केलेल्या भूमिकेवर कंट्रोल नसते.

## • विवेकी ऍक्सेस कंट्रोल (DAC)

नावाप्रमाणेच, हे ऍक्सेस कंट्रोल मॉडेल यूजर च्या विवेकबुद्धीवर आधारित आहे. म्हणजेच, संसाधनाचा मालक त्याच्या विवेकबुद्धीच्या आधारावर इतर वापरकर्त्यांना त्या संसाधनावरील ऍक्सेस अधिकार देऊ शकतो. ऍक्सेस कंट्रोल याद्या (ACLs) हे DAC चे स्पेसिफिक उदाहरण आहे. तुमच्या मालकीच्या युनिक्स फाईलवर "rwx" परवानग्या स्पेसिफाइड करणे हे DAC चे आणखी एक उदाहरण आहे Windows, Unix च्या फ्लेवर्ससह बहुतेक ऑपरेटिंग सिस्टम DAC मॉडेलवर आधारित आहेत.

## • अनिवार्य ऍक्सेस कंट्रोल (MAC)

या मॉडेलमध्ये, वापरकर्ते/मालक त्यांच्या फायलींमध्ये कोण ऍक्सेस करू शकतात हे ठरवण्याचा विशेषाधिकार घेत नाहीत. येथे ऑपरेटिंग सिस्टीम ही यूजर च्या इच्छेवर अवलंबून निर्णय घेणारी आहे. या मॉडेलमध्ये प्रत्येक विषय (वापरकर्ते) आणि ऑब्जेक्ट (रिसोर्स) वर्गीकृत आणि सुरक्षा लेबलसह नियुक्त केले आहेत. विषयाची सुरक्षा लेबले आणि सुरक्षा धोरणासह ऑब्जेक्ट ऑब्जेक्टमध्ये ऍक्सेस करू शकतो की नाही हे निर्धारित करतात. सुरक्षा अधिकारी, प्रशासकाद्वारे कॉन्फिगर केलेले, ऑपरेटिंग सिस्टीमद्वारे लागू केलेले आणि सुरक्षा तंत्रज्ञानाद्वारे समर्थित असलेले विषय वस्तूंमध्ये ऍक्सेस कसा करतात याचे नियम.

DAC च्या तुलनेत हे एक कठोर आणि ऐवजी स्थिर ऍक्सेस कंट्रोल मॉडेल आहे आणि बहुतेक लष्करी संस्थांसाठी सबयुक्त आहे जेथे डेटा वर्गीकरण आणि प्रायव्हसी ला महत्त्व आहे. स्पेशल टाइप ची युनिक्स ऑपरेटिंग सिस्टम MAC मॉडेलवर आधारित आहेत.

## • एट्रिब्यूट आधारित ऍक्सेस कंट्रोल (ABAC)

हे यूजर च्या कंट्रोल गुणधर्मांवर, ऑब्जेक्टच्या कंट्रोल गुणधर्मांच्या आधारावर यूजर च्या विनंत्या मंजूर करणे किंवा नाकारणे आहे आणि सध्याच्या धोरणांशी अधिक संबंधित असू शकतील अशा पर्यावरणीय परिस्थिती.

## पासवर्ड क्रॅक करण्याच्या मेथड्स आणि त्यांचे काउंटर उपाय

खाती हॅक करण्यासाठी किंवा वैयक्तिक इनफार्मेशन चोरण्यासाठी हॅकर्सद्वारे अनेक मेथड्स वापरल्या जातात. पासवर्ड क्रॅक करण्यासाठी सर्वात सामान्यपणे वापरल्या जाणाऱ्या काही मेथड्स आणि त्यांचे प्रतिकारक उपाय खालीलप्रमाणे आहेत:

### 1 BruteForce अटॅक

ब्रूट-फोर्स अटॅक वापरून कोणताही पासवर्ड क्रॅक केला जाऊ शकतो. जोपर्यंत योग्य पासवर्ड जुळत नाही तोपर्यंत ब्रूट-फोर्स हल्ले नंबर , अक्षरे

आणि स्पेशल कॅरक्टर च्या प्रत्येक संभाव्य संयोजनाचा प्रयत्न करतात. पासवर्डच्या जटिलतेनुसार ब्रूट-फोर्स हल्ल्यांना बराच वेळ लागू शकतो. संगणकाचा वेग आणि पासवर्डची जटिलता यावर अवलंबून.

**प्रतिकारक उपाय :** लांब आणि जटिल पासवर्ड वापरा. नंबर सह अप्पर आणि लोअरकेस अक्षरांचे संयोजन वापरण्याचा प्रयत्न करा. ब्रूट-फोर्स अटॅक असे जटिल आणि लांब पासवर्ड क्रॅक करण्यासाठी खूप वेळ लागेल. तुम्ही पासवर्डही वारंवार बदलत राहू शकता.

## 2 सोशल इंजिनरींग

सोशल इंजिनरींग म्हणजे एखाद्याला तुमच्यावर विश्वास ठेवण्यासाठी आणि त्यांच्याकडून इनफार्मेशन मिळविण्यासाठी हाताळण्याची प्रक्रिया. उदाहरणार्थ, जर हॅकर सहकर्मचारी किंवा मित्रांच्या संगणकाचा पासवर्ड मिळवण्याचा प्रयत्न करत असेल, तर तो आयटी विपार्ट किंवा बँकेचा असल्याचे भासवून त्याला कॉल करू शकतो आणि त्याचे लॉगिन किंवा क्रेडिट कार्ड तपशील विचारू शकतो.

**काउंटरमेजर:** क्रेडिट कार्ड तपशीलासारखी तुमची सेन्सिटिव्ह इनफार्मेशन फोनवर कधीही देऊ नका.

## 3 राट्स आणि कीलॉगर्स

कीलॉगिंग किंवा रेटिंगमध्ये हॅकर पीडित व्यक्तीला कीलॉगर किंवा राट्स पाठवतो. हे हॅकरला त्याच्या संगणकावर पीडित प्रत्येक गोष्टीचे निरीक्षण करण्यास अनुमती देते. प्रत्येक कीस्ट्रोक पासवर्डसह लॉग केलेला आहे. शिवाय हॅकर पीडिताच्या संगणकावरही कंट्रोल ठेवू शकतो.

**प्रतिकारक उपाय:** सायबर कॅफे किंवा इतर कोणाच्या संगणकावरून कधीही तुमच्या बँक खात्यात लॉग इन करू नका. हे खूप महत्वाचे असल्यास, लॉगिन बांधताना ऑन-स्क्रीन किंवा व्हर्चुअल कीबोर्ड वापरा. नवीनतम अँटी-व्हायरस सॉफ्टवेअर वापरा आणि व्याख्या अपडेट ठेवा.

## 4 फिशिंग

फिशिंग ही सर्वात सोपी आणि लोकप्रिय हॅकिंग मेथड्स आहे जी हॅकर्सद्वारे एखाद्याच्या खात्याचे तपशील मिळवण्यासाठी वापरली जाते. फिशिंग हल्ल्यात हॅकर पीडित व्यक्तीला फेसबुक, जीमेल सारख्या वास्तविक वेबसाइटचे बनावट पृष्ठ पाठवतो. जेव्हा कोणी त्या बनावट पृष्ठाद्वारे लॉग इन करते तेव्हा त्याचे तपशील हॅकरला पाठवले जातात. ही बनावट पेजेज सहजपणे तयार केली जाऊ शकतात आणि त्यावर होस्ट केली जाऊ शकतात विनाव्हॅल्यू वेबहोस्टिंग साइट्स.

**काउंटरमेजर:** फिशिंग हल्ले टाळणे खूप सोपे आहे. या फिशिंग पेजची url वास्तविक पृष्ठापेक्षा वेगळी आहे. उदाहरणार्थ फेसबुकच्या फिशिंग पेजची URL facbook.com सारखी दिसू शकते (जसे तुम्ही पाहू शकता की दोन "b" आहेत). वेबसाइटची url बरोबर असल्याची नेहमी खात्री करा.

## 5 रेनबो टेबल

रेनबो टेबल ही कॅरक्टर च्या प्रत्येक संभाव्य संयोजनासाठी हॅशची एक मोठी पूर्व-संगणित लिस्ट आहे. पासवर्ड हॅश हा एक पासवर्ड आहे जो md5 सारख्या गणिती अल्गोरिदममधून गेला आहे आणि ओळखण्यायोग्य



नसलेल्या गोष्टीमध्ये बदलला आहे. हॅश हे एकेरी एन्क्रिप्शन आहे म्हणून एकदा पासवर्ड हॅश केल्यावर हॅश केलेल्या स्ट्रिंगमधून मूळ स्ट्रिंग मिळवण्याचा कोणताही पाथ नाही.

**प्रतिकारक उपाय:** तुम्ही लांब आणि गुंतागुंतीचा पासवर्ड निवडल्याची खात्री करा. लांब आणि गुंतागुंतीच्या पासवर्डसाठी टेबल तयार करण्यासाठी खूप वेळ लागतो आणि भरपूर रिसोर्सस लागतात.

## 6 अंदाज लावणे

ही एक सोपी मेथड्स आहे जी तुम्हाला काही सेकंदात एखाद्याचा पासवर्ड मिळवण्यात मदत करते. जर हॅकर तुम्हाला ओळखत असेल, तर तो तुमच्या पासवर्डचा अंदाज घेण्यासाठी त्याला तुमच्याबद्दल माहीत असलेली इनफार्मेशन वापरू शकतो. तुमचा पासवर्ड मिळवण्यासाठी हॅकर सोशल इंजिनीअरिंग आणि गेसिंगचाही वापर करू शकतो.

**काउंटर मेजर:** तुमचा पासवर्ड म्हणून तुमचे नाव, आडनाव, फोन नंबर किंवा जन्म date वापरू नका. तुमच्याशी संबंधित पासवर्ड तयार करणे टाळण्याचा प्रयत्न करा. अक्षरे आणि नंबर च्या संयोजनाने जटिल आणि लांब पासवर्ड तयार करा.

## इनफार्मेशन साठी सुरक्षा वर्गीकरण

सर्व इनफार्मेशन समान नसते आणि म्हणून सर्व इनफार्मेशन साठी समान प्रमाणात प्रोटेक्शन आवश्यक नसते. यासाठी सुरक्षा वर्गीकरण नियुक्त करणे आवश्यक आहे.

इनफार्मेशन वर्गीकरणाची पहिली स्टेप म्हणजे वरिष्ठ मॅनेज्मेंट च्या सदस्याला स्पेसिफिक इनफार्मेशन चा मालक म्हणून ओळखणे. पुढे, वर्गीकरण धोरण विकसित करा. धोरण सक्षम असावे:

- इनफार्मेशन साठवणे
- इनफार्मेशन प्रसारित करणे
- विविध वर्गीकरण लेबल्सचे डिस्क्रिप्शन करा,
- स्पेसिफिक लेबल नियुक्त करण्यासाठी इनफार्मेशन चे निकष परिभाषित करा आणि
- प्रत्येक वर्गीकरणासाठी आवश्यक सुरक्षा कंट्रोल लिस्ट बद्ध करा.
- अनावश्यक इनफार्मेशन ची विल्हेवाट लावणे
- इनफार्मेशन च्या अखंडतेचे प्रोटेक्शन करणे
- योग्य एक्सेस आणि प्रकटीकरण करण्याची परवानगी देणे
- जबाबदारी प्रिंस्टाल करणे.

कोणती वर्गीकरण इनफार्मेशन नियुक्त केली जावी यावर काही एलिमेंट इफेक्ट टाकतात:

- त्या इनफार्मेशन चे संस्थेसाठी किती महत्त्व आहे
- इनफार्मेशन किती जुनी आहे आणि
- इनफार्मेशन अप्रचलित झाली आहे की नाही.
- कायदे आणि इतर नियामक आवश्यकता

इनफार्मेशन सुरक्षेसाठी बिझनेस मॉडेल सुरक्षा व्यावसायिकांना सिस्टीमच्या दृष्टीकोनातून सुरक्षेचे परीक्षण करण्यास सक्षम करते, असे वातावरण तयार करते जेथे सुरक्षिततेचे सर्वांगीण मॅनेजमेंट केले जाऊ शकते, वास्तविक जोखमीचे निराकरण केले जाऊ शकते.

इनफार्मेशन सुरक्षा वर्गीकरण लेबल्सचा टाइप निवडलेला आणि वापरला जातो हे संस्थेच्या स्वरूपावर अवलंबून असेल, उदाहरणार्थ:

- व्यवसाय क्षेत्रात, लेबले जसे की: सार्वजनिक, सेन्सिटिव्ह , प्राइवेट आणि गोपनीय.
- सरकारी क्षेत्रात, लेबले जसे की: अवर्गीकृत, सेन्सिटिव्ह परंतु अवर्गीकृत, प्रतिबंधित, गोपनीय, गुप्त, शीर्ष गुप्त आणि त्यांचे गैर-इंग्रजी समकक्ष.
- क्रॉस-सेक्टरल फॉर्मेशन्समध्ये, ट्रॅफिक लाइट प्रोटोकॉल, ज्यामध्ये पांढरा, हिरवा, अंबर आणि लाल यांचा समावेश असतो.

संस्थेतील सर्व कर्मचारी तसेच व्यावसायिक भागीदारांना वर्गीकरण योजनेचे प्रशिक्षण दिले पाहिजे आणि प्रत्येक वर्गीकरणासाठी आवश्यक सुरक्षा कंट्रोल आणि हाताळणी प्रक्रिया समजून घेणे आवश्यक आहे. स्पेसिफिक इनफार्मेशन मालमतेचे वर्गीकरण वर्गीकरण अद्याप इनफार्मेशन साठी योग्य आहे याची खात्री करण्यासाठी आणि वर्गीकरणासाठी आवश्यक सुरक्षा कंट्रोल कार्यरत आहेत याची खात्री करण्यासाठी नियुक्त केलेल्यांचे वेळोवेळी रिव्यु केले जावे.

## वर्गीकरण आणि डाउनग्रेडिंग

इनफार्मेशन चे वर्गीकरण किंवा नियुक्त केले जाणे आवश्यक आहे ज्या वेळेस तिला संरक्षणाची आवश्यकता आहे, त्यानंतर ती अवर्गीकृत किंवा अवनत केली जाईल. याचे कारण असे की वर्गीकृत किंवा नियुक्त इनफार्मेशन वेळोवेळी किंवा स्पेसिफिक घटनांच्या घटनेसह तिची सेन्सिटिव्ह ता गमावेल. ही प्रक्रिया सुरक्षा व्यवस्थेच्या संपूर्ण अखंडतेमध्ये योगदान देते आणि लोकांच्या स्वारस्य असलेल्या सदस्यांना इनफार्मेशन जलद आणि अनौपचारिकपणे सबलब्ध करून दिली जाते याची खात्री करते.

## एक्सेस कंट्रोल एडमिनिस्ट्रेशन

एक्सेस कंट्रोल एडमिनिस्ट्रेशन दोन प्रकारे केले जाऊ शकते.

- **सेंट्रलाइज्ड :** येथे एक संस्था (dept. किंवा एखादी व्यक्ती) सर्व कॉर्पोरेट संसाधनांच्या प्रवेशावर देखरेख करण्यासाठी जबाबदार आहे. या टाइप चे एडमिनिस्ट्रेशन वापरकर्त्यांच्या एक्सेस अधिकारांवर कंट्रोल ठेवण्यासाठी एक सुसंगत आणि एकसमान मेथड्स प्रदान करते. उदाहरण: RADIUS, TACACS आणि व्यास

- डीसेंट्रलाइज्ड

एक्सेस कंट्रोल / डेटा संकलन प्रोटोकॉल: AAA (RADIUS, व्यास, आणि TACACS+)

RADIUS, Diameter आणि TACACS+ हे नेटवर्क ऍक्सेस सर्व्हर (NAS) मधील लिंक्स किंवा अंतिम वापरकर्त्यांना प्रमाणीकरण करू इच्छिणाऱ्या प्रमाणीकरण, अधिकृतता आणि लेखा (AAA) इनफार्मेशन घेऊन जाण्यासाठी तीन प्रोटोकॉल आहेत.

## रेडियस

रिमोट ऑथेंटिकेशन डायल-इन युजर सर्विस (RADIUS) हा ल्युसेंट इंटरनेटवर्किंग सिस्टम्सद्वारे तयार केलेला क्लायंट/सर्वर सुरक्षा प्रोटोकॉल आहे. RADIUS हा इंटरनेट मसुदा स्टॅण्डर्ड प्रोटोकॉल आहे. युजर प्रोफाइल मध्यवर्ती ठिकाणी स्टोर केले जातात, ज्याला RADIUS सर्वर म्हणतात. RADIUS क्लायंट (जसे की PortMaster कम्युनिकेशन सर्वर) वापरकर्त्यांना प्रमाणीकृत करण्यासाठी RADIUS सर्वरशी संवाद साधतात. सर्वर क्लायंटला परत स्पेसिफाइड करतो की प्रमाणीकृत युजर काय करण्यास अधिकृत आहे. जरी RADIUS हा शब्द क्लायंट आणि सर्वर कम्युनिकेशन करण्यासाठी वापरत असलेल्या नेटवर्क प्रोटोकॉलचा रेफरेन्स देत असला तरी, तो बहुतेकदा संपूर्ण क्लायंट/सर्वर सिस्टमचा रेफरेन्स देण्यासाठी वापरला जातो.

**डायमीटर:** व्यास हा संगणक नेटवर्कसाठी प्रमाणीकरण, अधिकृतता आणि लेखा प्रोटोकॉल आहे. ते त्याच्या आधीच्या कमी सक्षम RADIUS प्रोटोकॉलपासून विकसित झाले आणि पुनर्स्थित करते. व्यास ऍप्लिकेशन्स नवीन कमांड आणि/किंवा स्पेशल ता जोडून बेस प्रोटोकॉल वाढवतात. हे RADIUS च्या तुलनेत उत्तम, उत्तम वाहतूक, उत्तम सुरक्षा, उत्तम प्रॉक्सी, उत्तम सत्र कंट्रोल आणि उत्तम इंटरऑपरेबिलिटी प्रदान करते.

**TACACS:** टर्मिनल ऍक्सेस कंट्रोल ऍक्सेस-कंट्रोल सिस्टम (TACACS, सहसा टॅक-अॅक्स सारखे उच्चारले जाते) हे सेंट्रलाइज्ड सर्वरद्वारे नेटवर्क ऍक्सेस कंट्रोलसाठी रिमोट ऑथेंटिकेशन आणि संबंधित सर्विस हाताळणाऱ्या संबंधित प्रोटोकॉलच्या कुटुंबाचा रेफरेन्स देते. मूळ TACACS प्रोटोकॉल, जो 1984 चा आहे, जुन्या UNIX नेटवर्कमध्ये सामान्य असलेल्या प्रमाणीकरण सर्वरशी संवाद साधण्यासाठी वापरला गेला. एक्सटेंडेड TACACS (XTACACS) हा TACACS चा एक मालकीचा एक्सपान्शन आहे जो 1990 मध्ये सिस्को सिस्टीम्सने मूळ प्रोटोकॉलशी सुसंगततेसाठी सादर केला होता. टर्मिनल ऍक्सेस कंट्रोल ऍक्सेसकंट्रोल सिस्टीम प्लस (TACACS+) हा सिस्कोने विकसित केलेला प्रोटोकॉल आहे आणि 1993 मध्ये खुला स्टॅण्डर्ड म्हणून जारी केला आहे.

## डीसेंट्रलाइज्ड एक्सेस कंट्रोल

- डीसेंट्रलाइज्ड एक्सेस कंट्रोल एडमिनिस्ट्रेशन मेथड्स संसाधनांच्या जवळच्या लोकांपर्यंत एक्सेस कंट्रोल करते
- या दृष्टिकोनामध्ये, बहुतेकदा कार्यात्मक व्यवस्थापक असतो जो कर्मचाऱ्यांना एक्सेस कंट्रोल अधिकार नियुक्त करतो.
- या टाइप च्या प्रशासनाद्वारे बदल जलद होऊ शकतात कारण केवळ एक एलिमेंट संपूर्ण संस्थेसाठी बदल करत नाही.
- संघर्ष निर्माण होण्याची शक्यता आहे ज्यामुळे संस्थेला फायदा होणार नाही कारण भिन्न व्यवस्थापक आणि विपार्ट वेगवेगळ्या प्रकारे सुरक्षा आणि एक्सेस नियंत्रणाचा सराव करू शकतात.
- काही कंट्रोल ओव्हरलॅप होण्याची शक्यता असते, अशा परिस्थितीत कृती योग्यरित्या प्रतिबंधित किंवा प्रतिबंधित नसतात.
- या टाइप चे एडमिनिस्ट्रेशन सेंट्रलाइज्ड मेथड्स प्रमाणे सातत्यपूर्ण नियंत्रणासाठी मेथड्स प्रदान करत नाही.

**एक सुरक्षा ऑडिटमूलत:** संस्थेचे सुरक्षा धोरण किती प्रभावीपणे अंमलात आणले जात आहे याचे असेसमेंट आहे. हे IT सिस्टिम चे धोरण, रेकॉर्ड आणि क्रियाकलापांचे स्वतंत्र रिव्यु आणि परीक्षण आहे.

इनफार्मेशन सिस्टिम ऑडिट महत्वाचे आहे कारण ते खात्री देते की IT सिस्टिम पुरेसे संरक्षित आहेत, वापरकर्त्यांना विश्वसनीय इनफार्मेशन प्रदान करतात आणि त्यांचे इच्छित फायदे साध्य करण्यासाठी योग्यरित्या मॅनेज केले जातात. हे डेटा छेडछाड, डेटा गमावणे किंवा गळती, सर्विस व्यत्यय आणि आयटी सिस्टिमचे खराब मॅनेजमेंट देखील कमी करते.

सिक्युरिटी ऑडिटिंग अँड टेस्टिंग (SAT) एखाद्या संस्थेला इंटरनल कारणांसाठी सुरक्षिततेची स्थिती समजून घेण्यास मदत करते आणि एक्सटर्नल पक्षांना खात्री देते, याकडे सर्वोच्च पातळीवर लक्ष देणे आवश्यक आहे. हे विद्यमान संरक्षणातील अंतर ओळखण्यास मदत करते.

## ऑडिट उद्दिष्टे इंस्टाल करणे

ऑडिटचे नियोजन केल्यानंतर आणि ऑडिट करण्यासाठी पुढे जाण्यापूर्वी, एखाद्याने ऑडिटची उद्दिष्टे इंस्टाल केली पाहिजेत. लेखापरीक्षकाने ज्या उद्दिष्टांचे रिव्यु केले पाहिजे त्यांची यादी खालीलप्रमाणे आहे:

- सिस्टिम आणि क्रॉस-फंक्शनल प्रशिक्षणासह कार्मिक प्रक्रिया आणि जबाबदाऱ्या
- बदल मॅनेजमेंट प्रक्रिया सुरू आहेत आणि IT आणि मॅनेजमेंट कर्मचारी त्यांचे पालन करतात. सेवेवरील कोणत्याही संबंधित घटनांची नंबर आणि रिझल्ट कमी करण्यासाठी, आयटी पायाभूत सुविधांवर कंट्रोल ठेवण्यासाठी सर्व बदलांच्या कार्यक्षम आणि त्वरित हाताळणीचा रेफरेन्स बदल मॅनेजमेंट चा आहे.
- डाउनटाइम कमी करण्यासाठी आणि महत्त्वाच्या डेटाचे नुकसान टाळण्यासाठी योग्य बॅकअप प्रक्रिया आहेत
- डेटा सेंटरमध्ये अनधिकृत एक्सेस रोखण्यासाठी डेटा सेंटरमध्ये पुरेशी फिजिकल सुरक्षा कंट्रोल आहेत
- उपकरणे आग आणि पुरापासून संरक्षित आहेत याची खात्री करण्यासाठी पुरेशी पर्यावरण कंट्रोल आहेत.

## ऑडिट नियोजन आणि तयारी

डेटा सेंटर रिव्यु आयोजित करण्यापूर्वी ऑडिटरला कंपनी आणि तिच्या महत्त्वपूर्ण व्यावसायिक क्रियाकलापांबद्दल पुरेसे शिक्षण दिले पाहिजे. रिव्यु करण्यापूर्वी ऑडिटरने खालील गोष्टी केल्या पाहिजेत:

- संभाव्य चिंतेची एरिया निश्चित करण्यासाठी IT मॅनेजमेंट स भेटा
- करंट IT संस्था चार्टचे रिव्यु करा
- डेटा सेंटर कर्मचाऱ्यांच्या जॉब कॅरक्टर नाचे रिव्यु करा
- डेटा सेंटरमध्ये कार्यरत असलेल्या सर्व ऑपरेटिंग सिस्टम्स, सॉफ्टवेअर ऍप्लिकेशन्स आणि डेटा सेंटर उपकरणांचे रिसर्च न करा
- कंपनीच्या IT धोरणांचे आणि कार्यपद्धतींचे रिव्यु करा
- कंपनीचे IT बजेट आणि सिस्टिम नियोजन दस्तऐवजीकरणाचे असेसमेंट

- डेटा सेंटरच्या आपत्ती पुनर्प्राप्ती योजनेचे रिव्यु करा.

### ऑडिट करत आहे

कोणतीही स्टॅण्डर्ड सुरक्षा-ऑडिट प्रक्रिया नाही, परंतु ऑडिटर्स सामान्यतः वैयक्तिक मुलाखती, असुरक्षा स्कॅन, OS ची तपासणी आणि सुरक्षा एप्लिकेशन सेटिंग्ज आणि नेटवर्क विश्लेषणे तसेच इव्हेंट लॉग सारख्या ऐतिहासिक डेटाचा अभ्यास करून त्यांचे कार्य पूर्ण करतात. ते काय कव्हर करतात, ते कसे वापरले जातात आणि चालू असलेल्या आणि भविष्यातील धोक्यांना तोंड देण्यासाठी ते प्रभावी आहेत की नाही हे निर्धारित करण्यासाठी ऑडिटर्स व्यवसायाच्या सुरक्षा धोरणांवर देखील लक्ष केंद्रित करतात.

साधारणपणे, संगणक सुरक्षा ऑडिट याद्वारे केले जातात:

- 1 फेडरल किंवा राज्य नियामक.
- 2 कॉर्पोरेट इंटर्नल ऑडिटर.
- 3 एक्सटर्नल लेखापरीक्षक - तंत्रज्ञान ऑडिटिंगशी संबंधित क्षेत्रांमध्ये स्पेशल .
- 4 सल्लागार - तंत्रज्ञान ऑडिटिंगचे आउटसोर्सिंग जेथे संस्थेकडे स्पेशल कौशल्य संच नसतो.

प्रथम, ऑडिटची व्याप्ती ठरवली जावी आणि त्यात संगणक उपकरणे, फोन, नेटवर्क, ईमेल, डेटा आणि कार्ड, टोकन आणि पासवर्ड यांसारख्या प्रवेशाशी संबंधित कोणत्याही वस्तूंसह इनफार्मेशन सुरक्षिततेची संबंधित सर्व कंपनी मालमत्तांचा समावेश करावा. त्यानंतर, भूतकाळातील आणि संभाव्य भविष्यातील मालमत्तेच्या धोक्यांचे रिव्यु करणे आवश्यक आहे. इनफार्मेशन सुरक्षा क्षेत्रातील कोणीही नवीन ट्रेंड, तसेच इतर कंपन्यांनी घेतलेल्या सुरक्षा उपायांबद्दल अवगत असले पाहिजे. पुढे, ऑडिटिंग टीमने धोक्याच्या परिस्थितीत किती विनाश होऊ शकतो याचा अंदाज लावला पाहिजे. धोका आल्यानंतर व्यवसाय चालवण्यासाठी एक इन्स्टाल योजना आणि कंट्रोल असावीत, ज्याला घुसखोरी प्रतिबंधक सिस्टिम म्हणतात.

### पुनरावलोकन करत आहे

पुढील स्टेप म्हणजे डेटा सेंटर ऑडिट उद्दिष्टे पूर्ण करण्यासाठी पुरावे गोळा करणे. यामध्ये डेटा सेंटर स्थानावर प्रवास करणे आणि डेटा सेंटरमध्ये केलेल्या प्रक्रिया आणि प्रक्रियांचे निरीक्षण करणे समाविष्ट आहे. पूर्वनिश्चित ऑडिट उद्दिष्टे पूर्ण करण्यासाठी खालील रिव्यु प्रक्रिया आयोजित केल्या पाहिजेत:

- ऑडिटरने डेटा सेंटर कर्मचार्यांचे त्यांचे उद्दिष्टे पूर्ण करण्यासाठी त्यांचे निरीक्षण केले पाहिजे आणि त्यांची मुलाखत घेतली पाहिजे.
- ऑडिटरने तपासले पाहिजे की सर्व डेटा सेंटर उपकरणे योग्य आणि प्रभावीपणे काम करत आहेत.
- सर्व डेटा सेंटर धोरणे आणि प्रक्रिया दस्तऐवजीकरण केल्या पाहिजेत आणि डेटा सेंटरमध्ये स्थित असाव्यात. महत्त्वाच्या दस्तऐवजीकरण प्रक्रियेमध्ये हे समाविष्ट आहे: डेटा सेंटर कर्मचारी नोकरीच्या जबाबदाऱ्या, बँकअप धोरणे, सुरक्षा धोरणे, कर्मचारी समाप्ती धोरणे, सिस्टिम ऑपरेटिंग प्रक्रिया आणि ऑपरेटिंग सिस्टिमचे ओवरविव्ह .

- ऑडिटरने क्लायंटच्या डेटा सेंटरच्या सुरक्षिततेचे फिजिकल सुरक्षा नियंत्रणांच्या संदर्भात असेसमेंट केले पाहिजे आणि डेटा सेंटर उपकरणांची सुरक्षा सुनिश्चित करण्यासाठी पर्यावरणीय कंट्रोल असावीत. यामध्ये हे समाविष्ट आहे: वातानुकूलन युनिट्स, उंच मजले, ह्यूमिडिफायर आणि अखंड वीजपुरवठा.

- बँकअप प्रक्रिया - ऑडिटरने हे सत्यापित केले पाहिजे की क्लायंटकडे सिस्टिम अयशस्वी झाल्यास बँकअप प्रक्रिया आहेत. क्लायंट वेगळ्या ठिकाणी बँकअप डेटा सेंटर ठेवू शकतात जे त्यांना सिस्टिम अयशस्वी झाल्यास त्वरित ऑपरेशन्स सुरू ठेवू शकतात.

### एक्सेस चाचण्या, असुरक्षितता असेसमेंट आणि सुरक्षा ऑडिट

सिस्टिम आणि नेटवर्कमध्ये घुसखोरी करण्यासाठी एक्सटर्नल हल्लेखोर अनेकदा आधीच ज्ञात असुरक्षा आणि शोषणाचा वापर करतात. योग्य बचावात्मक उपाय आणि पुरेशी सुरक्षा रचना केल्याने ही समस्या कमी होऊ शकते. आधीच अस्तित्वात असलेल्या उघड सिस्टिम आणि त्यांचे धोके नियमित अंतराने ओळखूनही हे बऱ्याच प्रमाणात साध्य केले जाऊ शकते. आधीच्या टप्प्यात रिस्क सर्च ली जाऊ शकतात आणि आधीच्या टप्प्यावर योग्य उपाययोजना केल्या जाऊ शकतात.

असुरक्षितता व्हॅल्यु मापन ही एक सराव आहे ज्याचा वापर सर्व संभाव्य असुरक्षा ओळखण्यासाठी केला जातो ज्यांचा वातावरणात शोषण केला जाऊ शकतो. फिजिकल सुरक्षा, कर्मचारी किंवा सिस्टिम आणि नेटवर्क सुरक्षिततेचे असेसमेंट करण्यासाठी असेसमेंट वापरले जाऊ शकते. त्यांना ओळखण्यासाठी असुरक्षा ओळख साधने वापरली जाऊ शकतात. IP पत्त्याद्वारे प्रत्येक संगणक सिस्टिम ची लिस्ट आणि त्यांच्याशी संबंधित असुरक्षा आणि असुरक्षा कशा "निराकरण" करायच्या यावरील पायऱ्या त्यानंतर तयार केल्या पाहिजेत.

ज्ञात भेद्यता वापरून "नेटवर्कमध्ये एक्सेस करणे" या हेतूने एक्सेस टेस्टिंग केली जाते. येथून, नेटवर्कमधील सर्वात गंभीर सिस्टिम वर एडमिनिस्ट्रेटर किंवा root एक्सेस मिळवणे हे उद्दिष्ट आहे. हे सिस्टिम आणि सिस्टिमवरील डेटामध्ये छेडछाड किंवा सुधारण्यासाठी नेटवर्कमध्ये पूर्ण एक्सेस देते. वास्तविक हॅकर काय करेल याचे अनुकरण करण्यासाठी पेनिट्रेशन टेस्टिंग केली जाते आणि ती कंपनीला सिद्ध करते की संस्थेमध्ये खरोखर एक्सेस केला जाऊ शकतो.

पेनिट्रेशन टेस्टिंगला एथिकल हॅकिंग असेही संबोधले जाते. बऱ्याच प्रकरणांमध्ये, सुरक्षा व्यावसायिक असुरक्षा स्कॅनरचे रिपोर्ट पाहू शकतो आणि कंपनीला कोणत्या धोक्याचा सामना करावा लागत आहे हे समजू शकतो.

### ऑडिट कंट्रोल

एप्लिकेशन कंट्रोलचे रिव्यु - हे डिप्लॉयमेंट तंत्रज्ञानाच्या जोखमीची ओळख आणि कंपनीच्या अशा जोखमीच्या संपर्कात कमी करणे, आवश्यक कंट्रोल आणि सुरक्षितता आहेत याची खात्री करून.

सामान्य संगणक नियंत्रणांचे रिव्यु - हे कंपनीमधील विविध प्लॅटफॉर्मवर चालणार्या एप्लिकेशन सिस्टिमसाठी सुरक्षित आणि स्थिर वातावरण प्रदान करण्यासाठी केले जाते.

ऑपरेशनल सिस्टममध्ये त्रुटी आढळल्यास सिस्टम गमावले जाऊ शकते नियंत्रणाची उद्दिष्टे

- संगणकात प्रविष्ट केलेला डेटा योग्य असल्याची खात्री करण्यासाठी
- संगणकावर डेटा इनपुट करण्यापूर्वी कारकुनी हाताळणी तपासा
- खराब डेटा किंवा खराब प्रोग्राममुळे उद्भवलेल्या त्रुटी सर्च पे आणि ट्रेस करण्याचे साधन प्रदान करा
- कायदेशीर आवश्यकता पूर्ण झाल्याची खात्री करा
- फसवणुकीपासून बचाव करण्यासाठी

## एक्सेस कंट्रोल आणि ऑडिटिंग

### फिजिकल आणि लॉजिकल सुरक्षा

सुरक्षिततेचे अनेक टाईप आहेत, फिजिकल आणि लॉजिकल . फिजिकल सुरक्षिततेमध्ये लॉक किंवा बायोमेट्रिक्स सारख्या गोष्टींचा समावेश असतो. लॉजिकल सुरक्षितता उदाहरणांमध्ये एक्सेस कंट्रोल आणि ऑडिटिंग, युजर खाते मॅनेजमेंट , उल्लंघन आणि सुरक्षा क्रियाकलाप रिपोर्ट आणि फायरवॉलसह सॉफ्टवेअर सुरक्षा उपायांचा समावेश आहे.

एक्सेस कंट्रोल ही एक सिस्टिम आहे जी प्राधिकरणांना दिलेल्या फिजिकल सुविधा किंवा संगणक-आधारित इनफार्मेशन सिस्टिम मधील एरिया आणि संसाधनांवर एक्सेस कंट्रोल करण्यास सक्षम करते. संगणकाचा पासवर्ड किंवा एटीएम सिस्टिम चा पिन हे एक्सेस नियंत्रणाचे टाईप आहेत. जेव्हा व्यक्ती गोपनीय, महत्त्वाची किंवा सेन्सिटिव्ह इनफार्मेशन आणि उपकरणे सुरक्षित करण्याचा प्रयत्न करतात तेव्हा एक्सेस कंट्रोल यंत्रणा वापरणे महत्त्वाचे असते.

एक्सेस कंट्रोल सिस्टिमचे ऑडिट करणे हा सिस्टिममध्ये एक्सेस द्वाराच्या घटनेचा मागोवा घेण्याचा एक पाथ आहे किंवा एक्सेस करण्याचा प्रयत्न केला आहे. हे महत्त्वाचे आहे कारण एक्सेस कंट्रोल सिस्टिम किती यशस्वी आहे, तसेच कोणाला एक्सेस नाकारण्यात आला होता आणि त्यांनी एकापेक्षा जास्त वेळा एक्सेस करण्याचा प्रयत्न केल्यास, त्यांचा हेतू काय आहे हे ते दर्शवते?

### खालील गोष्टी तपासण्यासाठी लॉजिकल ऑडिट केले जाते

- एक्सेस उल्लंघनाची ताकद आणि कमकुवतता
- सुरक्षा एक्टिविटी रिपोर्ट
- लॉगिंग क्रियाकलाप रिपोर्ट
- फायरवॉलची कार्यक्षमता
- सुरक्षेच्या उल्लंघनाचा रिपोर्ट
- अनधिकृत व्यक्तींनी हॅक करण्याचा प्रयत्न केल्याचे रिपोर्ट सिस्टिम इ.

**लेखापरीक्षकांसाठी व्यावसायिक नैतिकता:** वस्तुनिष्ठ ऑडिटमध्ये विश्वास एडिटिंग करण्यासाठी, व्यावसायिक नैतिकतेचा संच राखणे आवश्यक आहे. व्यावसायिक नैतिकता वैयक्तिक व्यक्तींनी तसेच इनफार्मेशन सुरक्षा ऑडिटिंगच्या क्षेत्रात सर्विस प्रदान करणाऱ्या कंपन्यांनी पाळली पाहिजे. व्यावसायिक नैतिकतेमध्ये खालील तत्त्वे असतात:

**प्रामाणिकपणा आणि प्रायव्हसी :** प्रामाणिकपणा हा विश्वासाचा पाया आहे आणि व्हॅल्यु कनाच्या विश्वासाहतेचा आधार बनतो. सेन्सिटिव्ह व्यावसायिक प्रक्रिया आणि इनफार्मेशन अनेकदा इनफार्मेशन च्या सुरक्षेवर अवलंबून असल्याचे आढळून आल्याने, ऑडिट दरम्यान मिळालेल्या इनफार्मेशन ची प्रायव्हसी आणि विवेकपूर्ण हाताळणी. IS ऑडिटचे निकाल आणि निष्कर्ष हे अशा कामासाठी महत्त्वाचे आधार आहेत. IS लेखापरीक्षकांना त्यांना मिळालेल्या इनफार्मेशन चे व्हॅल्यु आणि ती कोणाची मालकी आहे याची त्यांना जाणीव असते आणि ते कायदेशीर किंवा व्यावसायिकरित्या तसे करणे आवश्यक असल्याशिवाय संबंधित परवानगीशिवाय ही इनफार्मेशन उघड करणार नाहीत.

**तज्ञांचे ज्ञान:** IS लेखापरीक्षक फक्त त्या नोकऱ्या स्वीकारतात ज्यासाठी त्यांच्याकडे आवश्यक ज्ञान आणि कौशल्ये तसेच संबंधित अनुभव आहेत आणि त्यांचे कार्य पार पाडताना त्यांचा वापर करतात. ते सतत त्यांचे ज्ञान तसेच त्यांच्या कामाची रिझल्ट कारकता आणि गुणवत्ता सुधारतात.

**वस्तुनिष्ठता आणि परिपूर्णता:** IS ऑडिटरने लेखापरीक्षित केलेल्या क्रियाकलाप किंवा व्यवसाय प्रक्रियांबद्दल इनफार्मेशन गोळा करताना, व्हॅल्यु मापन करताना आणि पास करताना तज्ञांची वस्तुनिष्ठता आणि परिपूर्णतेची सर्वोच्च संभाव्य पातळी प्रदर्शित करणे आवश्यक आहे. सर्व संबंधित परिस्थितींचे व्हॅल्यु मापन निःपक्षपातीपणे केले जाणे आवश्यक आहे आणि ते लेखापरीक्षकांच्या स्वतःच्या हितसंबंधांनी किंवा इतरांच्या हितसंबंधांवर इफ्रेक्ट टाकू शकत नाही.

**वस्तुनिष्ठ प्रेसेंटेशन :** IS ऑडिटरचे कर्तव्य आहे की ते परीक्षेचे निकाल अचूकपणे आणि सत्यतेने त्याच्या क्लायंटला कळवावे. यामध्ये IS लेखापरीक्षण अहवालांमधील तथ्यांचे निष्पक्ष आणि समजण्यायोग्य प्रेसेंटेशन , निश्चित केलेल्या तथ्यांचे रचनात्मक असेसेमेंट आणि सुरक्षितता आणि प्रक्रिया सुधारण्यासाठी स्पेसिफिक शिफारसी समाविष्ट आहेत.

**पडताळणी आणि पुनरुत्पादनक्षमता:** विश्वसनीय आणि समजण्याजोगे निष्कर्ष आणि परिणामांचा तर्कसंगत आधार म्हणजे वास्तविक तथ्यांचे स्पष्ट आणि सुसंगत दस्तऐवजीकरण.

यामध्ये हे देखील समाविष्ट आहे की IS ऑडिट टीम त्याच्या निष्कर्षापर्यंत पोहोचण्यासाठी दस्तऐवजीकरण केलेल्या आणि पुनरुत्पादित मेथड्स चे अनुसरण करते.

**अनुपालन ऑडिट:** अनुपालन लेखापरीक्षण हे नियामक पाथ दर्शक तत्त्वांचे संस्थेच्या पालनाचे सर्वसमावेशक रिव्यु आहे. स्वतंत्र लेखा, सुरक्षा किंवा आयटी सल्लागार अनुपालन तयारीची ताकद आणि कसून असेसेमेंट करतात. अनुपालन ऑडिट दरम्यान ऑडिटर्स सुरक्षा धोरणे, युजर एक्सेस कंट्रोल आणि रिस्क मॅनेजमेंट प्रक्रियांचे रिव्यु करतात.

अनुपालन लेखापरीक्षणामध्ये काय तपासले जाते ते संस्था सार्वजनिक किंवा प्राइवेट कंपनी आहे की नाही, ती कोणत्या टाईप चा डेटा हाताळते आणि ती सेन्सिटिव्ह आर्थिक डेटा प्रसारित करते किंवा स्टोर करते यावर अवलंबून असते.

**इनफार्मेशन सुरक्षा धोरणे:** संस्था इनफार्मेशन सुरक्षा धोरणांच्या विकासासाठी अधिक प्राधान्य देत आहेत, कारण त्यांच्या मालमत्तेचे प्रोटेक्शन

करणे ही एक प्रमुख गोष्ट आहे ज्याचा विचार करणे आवश्यक आहे. इनफार्मेशन सुरक्षा धोरणांमध्ये स्पष्टतेच्या अभावामुळे गंभीर नुकसान होऊ शकते जे भरून काढता येत नाही. त्यामुळे सुरक्षा धोरण यशस्वीपणे राबविण्यासाठी संवैदिक विविध धोरणे आखते. इनफार्मेशन सुरक्षा धोरण संपूर्ण संस्थेतील इनफार्मेशन सुरक्षिततेसाठी मॅनेजमेंट दिशा आणि सपोर्ट प्रदान करते.

प्रायव्हसी, अखंडता आणि सबलब्धतेच्या उल्लंघनाच्या परिणामांपासून संस्थेतील इनफार्मेशन योग्यरित्या सुरक्षित करणे आवश्यक आहे. अनधिकृत बदल, हटवणे आणि प्रकटीकरणांपासून इनफार्मेशन कंट्रोल करण्यासाठी आणि सुरक्षित करण्यासाठी योग्य सुरक्षा उपायांची अंमलबजावणी करणे आवश्यक आहे. लागू करणे आवश्यक असलेल्या सुरक्षा उपायांची पातळी सर्च प्यासाठी, रिस्क असेसमेंट अनिवार्य आहे.

इनफार्मेशन सिस्टिम च्या संदर्भात संस्थेतील कर्मचाऱ्यांकडून काय अपेक्षित आहे हे परिभाषित करण्यासाठी सुरक्षा धोरणांचा हेतू आहे. इनफार्मेशन च्या मालमत्तेला धोका कमी करण्यासाठी सिस्टिमच्या वापरावर पाथ दर्शन करणे किंवा त्यावर कंट्रोल ठेवणे हे उद्दिष्ट आहे. हे इनफार्मेशन प्रणालींशी व्यवहार करणाऱ्या कर्मचाऱ्यांना स्वीकार्य वापराचे धोरण देखील देते, काय परवानगी आहे आणि काय नाही हे स्पष्ट करते. सर्व कंपन्यांची सुरक्षा धोरणे सारखी नसतात, परंतु त्यामागील मुख्य हेतू मालमत्तेचे प्रोटेक्शन करणे हा आहे. सुरक्षा धोरणे स्पेसिफिक उद्दिष्टांसह तयार केली जातात.

**इनफार्मेशन सुरक्षा ऑडिट साधने:** इनफार्मेशन सुरक्षा ऑडिट टूल्समध्ये युटिलिटीज आणि पॉवर टूल्सचा समावेश होतो, ओपन सोर्स आणि कमर्शियल दोन्ही.

**यूटिलिटी टूल्स:** ही सिंगल -उद्देश साधने आहेत जी एकतर ऑपरेटिंग सिस्टिमसाठी मूळ असू शकतात किंवा ओपन पणे सबलब्ध असू शकतात. युटिलिटी टूल्सना मॅन्युअल दृष्टीकोन आवश्यक आहे, जरी ते सहसा कस्टमाइज्ड स्क्रिप्टमध्ये समाविष्ट केले जातात--किंवा व्यावसायिक उत्पादनांमध्ये देखील. नेटवर्क टार्गेट ICMP पॅकेटला प्रतिसाद देते की नाही हे निर्धारित करण्यासाठी वापरल्या जाणाऱ्या बहुतेक प्लॅटफॉर्मवर सबलब्ध असलेल्या पिंगसारख्या लोकल सबयुक्तता देखील तुम्ही समाविष्ट करू शकता.

**Pros:** सबयुक्तता साधने ओपन पणे सबलब्ध आहेत आणि स्पेसिफिक कार्यासाठी घट्टपणे केंद्रित आहेत, ज्यामुळे ते अधिक कार्यक्षम बनतात. ते व्यक्तिचलितपणे आढळलेल्या भेद्यतेपेक्षा असुरक्षा सर्च प्यात मदत करतात.

**Cons:** त्यांचा वापर करण्यासाठी कौशल्य आवश्यक आहे. मोठ्या ऑडिटसाठी, मॅन्युअल टेस्टिंग वेळ घेणारी असते आणि ऑडिटरच्या कौशल्यावर अवलंबून, विसंगत रिझल्ट देऊ शकते.

**ट्रेसराउट:** नेटवर्क ट्रेसिंग युटिलिटी होस्टला नेटवर्क पाथ निर्धारित करण्यासाठी वापरली जाते.

**nslookup:** डोमेन मालकी निश्चित करण्यासाठी वापरले जाते.

आणि ओपन -सोर्स स्क्रिप्ट, यासह:

**Nmap:** विनाव्हॅल्यु पोर्ट-स्कॅनिंग सबयुक्तता.

**क्रॅक:** लोकप्रिय पासवर्ड-क्रॅकिंग साधन पासवर्ड कमकुवत आहेत की नाही हे निर्धारित करण्यासाठी ते तोडण्याचा प्रयत्न केला जातो.

**जॉन द रिपर:** युनिक्स पासवर्ड सर्च प्यासाठी प्रामुख्याने वापरलेले पासवर्ड-क्रॅकिंग साधन.

**binfo.c:** एक BIND आवृत्ती तपासक, binfo ही रिमोट नेम सर्व्हरवर चालू असलेल्या नावाची आवृत्ती मागे घेण्यासाठी एक द्रुत छोटी स्क्रिप्ट आहे.

**ghba.c:** दिलेल्या क्लास B किंवा C सबनेटचे सर्व मशीन नावे आणि IP एड्रेस काढण्यासाठी एक सुलभ साधन.

## पॉवर टूल्स

पॉवर टूल्स ही ऑडिट प्रक्रियेचे पार्ट सुव्यवस्थित आणि ऑटोमॅटिक करण्यासाठी मल्टी-फंक्शन बंडल युटिलिटीज आहेत. काही ओपन-सोर्स पॅकेजेस आहेत, तर अनेक कस्टम असुरक्षा डेटाबेससह व्यावसायिक उत्पादने आहेत.

**Pros:** ऑटोमॅटिक साधने डेटाबेसच्या विरुद्ध भेद्यतेसाठी स्कॅन करतात. सूचना हेल्प डेस्क मॉनिटरिंग टूल्समध्ये जोडल्या जाऊ शकतात. काही प्रकरणांमध्ये, स्कॅनिंग साधन फायरवॉल किंवा घुसखोरी सर्च मॅनेजमेंट स्टेशनसह एकत्रित केले जाऊ शकते. काही व्यावसायिक स्कॅनर एक्सपोजर आणि संबंधित रिस्क तपशीलवार उत्कृष्ट रिपोर्ट तयार करतात.

**Cons:** स्कॅनर फक्त त्यांच्या डेटाबेसमधील भेद्यता तपासतात, ज्या करंट असणे आवश्यक आहे. अनेक स्कॅनर असुरक्षितता तपासण्यांच्या संख्येवर विकले जातात. हे साधनाच्या रिझल्ट कारकतेचे नेहमीच चांगले संकेत नसते. अनेकदा, असुरक्षिततेचे चुकीचे निदान केले जाते. स्कॅनर जोखमीचे अचूक असेसमेंट करू शकत नाही.

काही ओपन-सोर्स पॉवर टूल्स म्हणजे Nessus, SARA (द सिक्स्युरिटी ऑडिटर्स रिसर्च असिस्टंट), व्हिस्कर, इत्यादी.

आज सबलब्ध असलेले काही व्यावसायिक स्कॅनर म्हणजे इंटरनेट सिक्स्युरिटी सिस्टमचे इंटरनेट स्कॅनर, eEye डिजिटल सिक्स्युरिटीचे रेटिना, BindView चे BV-Control, CORE सिक्स्युरिटी टेक्नॉलॉजीचे ऑडिटिंग टूल्स सूट आणि फाउंडस्टोनचे फाउंडस्कॅन.

## Google

खरा हॅकर बॉक्सच्या बाहेर विचार करतो आणि साधने अशा प्रकारे वापरण्यास शिकतो ज्याचा हेतू नसावा. Google सर्च इंजिन हे काटेकोरपणे सांगायचे तर ऑडिटिंग साधन नसले तरी साइटबद्दल इनफार्मेशन गोळा करण्यासाठी ते उत्तम आहे. उदाहरणार्थ, "@DGET.com" प्रविष्ट करण्याचा प्रयत्न करा (जेथे "DGET" तुमचे डोमेन आहे). काहीवेळा, यामुळे काही चांगला डेटा मिळू शकतो, जसे की सिस्टम एडमिनिस्ट्रेटर त्याच्या साइटबद्दल तांत्रिक तपशील पोस्ट करतो, ज्यामध्ये त्याचे खाते नाव सोपीस्कर असते. Google हे स्टिरॉइड्सवरील युनिक्स "grep" कमांडसारखे आहे.

## रिझल्ट कम्युनिकेशन

लेखापरीक्षकांशी संवाद कसा साधायचा हा विचार केला जाणारा अंतिम मुद्दा आहे, म्हणजे. ज्या व्यक्तींच्या मालमत्तेचे ऑडिट केले जात आहे. लेखापरीक्षकांना सतत ऑडिट क्रियाकलाप परिणामांची इनफार्मेशन देताना, एक्सचेंज स्वतंत्र आणि सातत्यपूर्ण असणे महत्वाचे आहे.

**परिभाषित पैरामीटर्सवर वरिष्ठ मॅनेज्मेंट स रिपोर्ट देणे**

मॅनेज्मेंट ला दिलेला सामान्य लेखापरीक्षण रिपोर्ट आणि मॅनेज्मेंट चा प्रतिसाद खालील तक्त्या 1 मध्ये दर्शविल्याप्रमाणे दिसू शकतो, परंतु अहवालांचे इतर अनेक फॉर्मेट आणि प्रतिसाद वापरात आहेत.

**Table 1**

Sl. No,	Findings	Impacts	Recommendations	Management Action / Response Plan	TimeLimit
1	The organization uses templates to configure new windows firewalls and servers, but does not have a documented technology configuration standard for other technologies such as DBMS, UNIX and LINUX operating systems etc.	The situation increases the risk of the unauthorized access to the organization's systems	The organization should continue documenting configuration standard based on the technologies in place.	Person Responsible : The chief information officer (CIO):The auditor's recommendations will be implemented. The security team will continue documenting configuration standards especially for UNIX and LINUX O.S.	No further action required
2	Patches are not up to date on LINUX based servers.	The situation increases the risk of the unauthorized access to the organization's systems. It also increases the risk of system failure.	Install the latest patches on the servers running on LINUX operating systems.	Person Responsible : The chief information Office. This was a result of complications in red hat LINUX maintenance contracts between the suppliers and the security team. The issue was settled in october 2014 and patches were installed.	No further action required.
3	We noted that users are their own workstation administrators, so that they can deactivate their workstation antivirus and idle system configurations.	This situation increases the risk of the unauthorized access to the organization's systems. It also increases the risk of system failure	Remove workstation administration privileges.	Person Responsible : The chief information officer. The situation was especially prevalanet with windows 2000 with the latest servers users only have the privileges to do their job. There are a few exceptions. The situation will be remediated as we are upgrading all our systems.	November 1st 2014

**प्रायव्हसी (प्रायव्हसी)प्रोटेक्शन आणि आयटी कायदा (Privacy Protection and IT Act)**

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- इनफार्मेशन च्या प्रायव्हसी चे डिस्क्रिप्शन करा
- इनफार्मेशन प्रणालींमध्ये प्रायव्हसी चे प्रोटेक्शन करण्याच्या मेथड्स चे डिस्क्रिप्शन करा
- सर्वोत्तम ऑनलाइन प्रायव्हसी पद्धतीचे डिस्क्रिप्शन करा
- आयटी कायदा काय आहे ते स्पष्ट करा
- सायबर गुन्हांचे डिस्क्रिप्शन करा.

**परिचय**

**प्रायव्हसी** एखाद्या व्यक्तीची किंवा गटाची त्यांची सिस्टिम , डेटा किंवा इनफार्मेशन वेगळे करण्याची आणि ती इलेक्ट्रिक पणे सामायिक करण्याची क्षमता आहे. जेव्हा एखादी गोष्ट एखाद्या व्यक्तीसाठी प्राइवेट असते, तेव्हा त्याचा अर्थ असा होतो की ती काहीतरी स्पेशल किंवा सेन्सिटिव्ह आहे.

इनफार्मेशन किंवा डेटा प्रायव्हसी चा रेफरेन्स आहे तंत्रज्ञान आणि कायदेशीर अधिकार यांच्यातील विकसित संबंध, किंवा सार्वजनिक अपेक्षा, एखाद्याच्या स्वतःबद्दलचा डेटा गोळा करणे आणि सामायिक करणे यामधील प्रायव्हसी . एखाद्या व्यक्तीशी किंवा व्यक्तीशी संबंधित अनन्यपणे ओळखण्यायोग्य डेटा डिजिटल स्वरूपात किंवा अन्यथा संकलित आणि स्टोर केला जातो तेव्हा प्रायव्हसी ची चिंता असते. काही प्रकरणांमध्ये या चिंता डेटा कसा संकलित केला जातो, स्टोर केला जातो आणि संबंधित असतो. इतर प्रकरणांमध्ये इनफार्मेशन चा एक्सेस कोणाला दिला जातो हा मुद्दा आहे. इतर समस्यांमध्ये एखाद्या व्यक्तीकडे त्यांच्या डेटावर मालकी हक्क आहेत की नाही आणि/किंवा ती इनफार्मेशन पाहण्याचा, पडताळण्याचा आणि आव्हान करण्याचा अधिकार आहे.

**इनफार्मेशन सेकुरेटी** अनधिकृत व्यक्तींना किंवा प्रणालींना प्रतिबंधित इनफार्मेशन वर एक्सेस मिळवण्यापासून रोखते. प्रायव्हसी हे नियम आणि दायित्वांचे संकलन आहे जे कोणत्याही माध्यमात एक्सेस कसा आणि केव्हा अधिकृत केला जावा हे निर्धारित करतात. हे खालीलप्रमाणे आहे की चांगली सुरक्षा आणि प्रायव्हसी मेथड्स एकमेकांवर अवलंबून आहेत. प्रायव्हसी चे डोमेन अंशतः सुरक्षिततेला ओव्हरलॅप करते, उदाहरणार्थ योग्य वापराच्या संकल्पना तसेच इनफार्मेशन चे प्रोटेक्शन .

संगणक सुरक्षा आणि प्रायव्हसी मधील संबंध या वस्तुस्थितीत आहे की पुरेशी संगणक सुरक्षा किंवा त्याचा अभाव, संगणक यूजर कडून अपेक्षित असलेल्या प्रायव्हसी च्या पातळीचे निर्धारक आहे. व्यवसाय, बँकिंग, सामाजिकीकरण आणि प्राइवेट इनफार्मेशन साठवणे यासह अनेक कामे करण्यासाठी लोक संगणक वापरतात. संगणकाच्या सुरक्षेचा भंग झाल्यास, या टाइप ची फंक्शन ज्या मेथड्स ने पार पाडली जातात त्यावर त्याचा नकारात्मक रिझल्ट होतो.

ई-कॉमर्सच्या क्षेत्रात, संगणक सुरक्षा आणि प्रायव्हसी चा मुद्दा व्यवसाय पक्षांमधील विश्वासाची पातळी निश्चित करेल. दोन्ही बाजूंनी सुरक्षेचा भंग

झाल्याची शंका असल्यास, यामुळे विश्वासाचा नाश होईल आणि व्यावसायिक संबंध संपुष्टात येतील. यामध्ये तृतीय पक्षांकडून जोखमीचा आणि धमक्यांचा समावेश आहे जो व्यवसाय भागीदारांशी देखील संबंधित नाही.

अयोग्य किंवा अस्तित्वात नसलेले प्रकटीकरण कंट्रोल हे प्रायव्हसी च्या समस्यांचे मूळ कारण असू शकते. डेटा प्रायव्हसी च्या समस्या अनेक स्त्रोतांकडून मिळालेल्या इनफार्मेशन च्या प्रतिसादात उद्भवू शकतात, जसे की:

- आरोग्य सर्विस नोंदी
- तपास आणि कार्यवाही
- वित्तीय संस्था आणि व्यवहार
- निवासपोजिशन आणि भौगोलिक नोंदी
- प्रोटेक्शन डेटा
- प्रायव्हसी चा भंग
- पोजिशन -आधारित सर्विस आणि भौगोलिक पोजिशन
- वैज्ञानिक रिसर्च इ.

वैयक्तिकरित्या ओळखण्यायोग्य इनफार्मेशन चे प्रोटेक्शन करताना डेटा सामायिक करणे हे डेटा प्रायव्हसी मधील आव्हान आहे. डेटा सुरक्षा आणि इनफार्मेशन सुरक्षा डिझाइनची फील्ड या समस्येचे निराकरण करण्यासाठी सॉफ्टवेअर, हार्डवेअर आणि मानवी रिसोर्सस वापरतात. डेटा संरक्षणाशी संबंधित कायदे आणि नियम सतत बदलत असल्याने, कायद्यातील कोणत्याही बदलांची इनफार्मेशन ठेवणे आणि डेटा प्रायव्हसी आणि सुरक्षा नियमांचे पालन केल्याचे सतत पुनर्मूल्यांकन करणे महत्त्वाचे आहे.

इनफार्मेशन प्रणालींमध्ये प्रायव्हसी चे प्रोटेक्शन करणे:भिन्न प्रायव्हसी नियमांसह विविध इनफार्मेशन सिस्टिम एकमेकांशी जोडलेली असल्याने आणि इनफार्मेशन सामायिक केली जात असल्याने, धोरण उपकरणांना प्रायव्हसी धोरण नियम (आणि कायदे) च्या वाढत्या प्रमाणात सामंजस्य, अंमलबजावणी आणि निरीक्षण करणे आवश्यक असेल. व्यावसायिक IT प्रणालींमध्ये प्रायव्हसी संरक्षणासाठी तंत्रज्ञानाच्या दोन रेन्ज आहेत: कम्प्युनिकेशन आणि अंमलबजावणी.

पॉलिसी कम्प्युनिकेशन P3P : हे प्रायव्हसी प्राधान्यांसाठी प्लॅटफॉर्म आहे. P3P हे प्रायव्हसी च्या मेथड्स कम्प्युनिकेशन करण्यासाठी आणि व्यक्तींच्या प्राधान्यांशी त्यांची कम्प्यारीजन करण्यासाठी एक स्टॅण्डर्ड आहे.

### धोरण अंमलबजावणी

- XACML - त्याच्या प्रायव्हसी प्रोफाइलसह एक्स्टेंसिबल ऍक्सेस कंट्रोल मार्कअप लॅंग्वेज हे मशीन-वाचण्यायोग्य भाषेत प्रायव्हसी धोरणे व्यक्त करण्यासाठी एक स्टॅण्डर्ड आहे ज्याचा वापर सॉफ्टवेअर सिस्टम एंटरप्राइझ आयटी सिस्टममध्ये धोरण लागू करण्यासाठी करू शकते.
- EPAL - एंटरप्राइझ प्रायव्हसी ऑथोरायझेशन भाषा ही XACML सारखीच आहे, परंतु ती अद्याप स्टॅण्डर्ड नाही.
- WS-प्रायव्हसी - "वेब सर्विस प्रायव्हसी" हे वेब सर्विस मधील प्रायव्हसी धोरण कम्प्युनिकेशन करण्यासाठी एक तपशील असेल. उदाहरणार्थ, वेब सर्विस संदेशाच्या SOAP लिफाफ्यात प्रायव्हसी धोरणाची इनफार्मेशन कशी एम्बेड केली जाऊ शकते हे ते स्पेसिफाइड करू शकते.

### इंटरनेटवर प्रायव्हसी चे प्रोटेक्शन करणे

इंटरनेटवर तुम्ही जवळजवळ नेहमीच तुमच्याबद्दल बरीच इनफार्मेशन देता. एनक्रिप्ट न केलेले ई-मेल ई-मेल सर्व्हरच्या प्रशासकांद्वारे वाचले जाऊ शकतात जेथे कनेक्शन एनक्रिप्ट केलेले नाही (https नाही). तसेच इंटरनेट सर्विस प्रदाता आणि त्या कनेक्शनची रहदारी सर्व गारे इतर पक्ष कंटेंट जाणून घेण्यास सक्षम आहेत. शिवाय, हेच इंटरनेटवर व्युत्पन्न होणाऱ्या कोणत्याही टाइप च्या ट्रॅफिकला लागू होते (वेब ब्राउझिंग, इन्स्टंट मेसेजिंग, ...) जास्त वैयक्तिक इनफार्मेशन देऊ नये म्हणून, ई-मेल्स कूटबद्ध केले जाऊ शकतात आणि वेबपेजेज तसेच इतर ब्राउझिंग करू शकतात. ऑनलाइन ॲक्टिव्हिटी निनावी द्वारे, किंवा, ज्यांवर विश्वास नसेल अशा प्रकरणांमध्ये, ओपन सोर्स डिस्ट्रिब्युटेड एनोनिमायझरद्वारे, तथाकथित मिक्स नेटद्वारे केले जाऊ शकतात. प्रसिद्ध ओपन-सोर्स मिक्स नेट म्हणजे I2P - निनावी नेटवर्क किंवा टोर.

### तुमच्या प्रायव्हसी चे रक्षण करा

इंटरनेट वापरत असताना तुमची प्रायव्हसी, वैयक्तिक इनफार्मेशन ऑनलाइन सुरक्षित ठेवण्यासाठी आणि फसवणूक आणि गैरवापर टाळण्यासाठी टिपा आणि पाथ दर्शक तत्वांची यादी खालीलप्रमाणे आहे.

- **नवीन पासवर्ड मिळवा:** तुमच्या प्रत्येक ऑनलाइन खात्यासाठी वेगवेगळे, मजबूत पासवर्ड वापरा जेणेकरून एखाद्याशी तडजोड झाली असेल तर बाकीचे सुरक्षित राहतील. मजबूत पासवर्डमध्ये अक्षरे, नंबर, भिन्न केस आणि सिम्बॉल्स असतात. तुमच्या पासवर्डची ताकद येथे तपासा.
- **जुनी ऑनलाइन खाती बंद करा:** न वापरलेली ऑनलाइन खाती ही जबाबदारी आहे. हॅकर्स त्यांचा वापर तुमच्या महत्त्वाच्या खात्यांमध्ये घुसखोरी करण्यासाठी करू शकतात. त्यांच्यापासून ओपन व्हा.
- **तुमच्या मित्रांची यादी कमी करा.**

- **पेपरलेस व्हा:** सेन्सिटिव्ह डेटा ऑनलाइन किंवा तुमच्या मेल खात्यांमध्ये ठेवू नका.
- **सेन्सिटिव्ह डॉक्युमेंट्स चे तुकडे करा:** सेन्सिटिव्ह डेटा असलेल्या नको त्या डॉक्युमेंट्स पासून ओपन व्हा. श्रेडर वापरून त्यांची सुरक्षितपणे विल्हेवाट लावा.

### ब्राउझर प्रायव्हसी

आधुनिक ब्राउझरमध्ये प्रायव्हसी वर्धित करण्याच्या क्षमता आणि पर्यायांची प्रभावी रेंज आहे. उदाहरणार्थ, तुम्ही संशयास्पद किंवा फसव्या वेबसाइट्सना भेट देण्यापूर्वी ते तुम्हाला चेतावणी देऊ शकतात आणि कुकीजसारख्या ट्रॅकिंग फाइल्स तुमच्या कॉम्प्युटरवर डाउनलोड न करता तुम्हाला वेब सर्फ करू शकतात. तसेच, जेव्हा एखादी वेबसाइट SSL वापरते तेव्हा बहुतेक ब्राउझर तुम्हाला इनफार्मेशन देऊ शकतात, एक सुरक्षितता उपाय जो तुमचा डेटा एन्क्रिप्ट करतो. जेव्हा एखादी वेबसाइट SSL वापरते तेव्हा ब्राउझर तुम्हाला पॅडलॉक चिन्ह (सामान्यतः वर स्थित) प्रदर्शित करून हे सूचित करू शकते तुमच्या ब्राउझरच्या तळाशी किंवा ॲड्रेस बारमध्ये वेबसाइटचे नाव हिरव्या रंगात हायलाइट करून.

फक्त विश्वसनीय वेबसाइट्सना भेट द्या. तुम्ही उघडत असलेल्या साइटबद्दल जाणून घेण्यासाठी साइट ॲडव्हायझर इत्यादी सारख्या ॲप्लिकेशन्सचा वापर करा. आज वेबसाइट अधिक इनफार्मेशन सामायिक करत असताना, ते त्यांच्या वापरकर्त्यांना या शेअरिंग क्रियाकलापांवर उत्कृष्ट स्पेसिफिक ता आणि कंट्रोल देखील प्रदान करतात. अनेक वेबसाइट्सवर तुम्हाला आढळले की तुम्ही वैयक्तिक इनफार्मेशन किंवा कंटेंट शेअर करता तेव्हा तुम्ही तुमचे प्रेक्षक परिभाषित करू शकता, मग ते एकाचे किंवा संपूर्ण लोकांचे प्रेक्षक असोत.

गेल्या दशकात ईमेल मोठ्या प्रमाणात अपरिवर्तित राहिले आहे. ईमेलचे शोषण करण्याच्या मेथड्स, तथापि, लक्षणीयरीत्या विकसित झाल्या आहेत आणि ईमेल वातावरणात वैयक्तिक इनफार्मेशन चे प्रोटेक्शन करणे अधिक आव्हानात्मक बनले आहे. गेल्या दशकात हॅकिंग अधिक प्रभावी आणि फिशिंग तंत्र, अधिक विस्तृत झाले आहे. ईमेल वापरताना तुमच्या प्रायव्हसी चे प्रोटेक्शन करण्यासाठी येथे काही धोरणे आहेत:

- 1 दुय्यम, "स्पॅम" ईमेल एड्रेस वापरा
- 2 मजबूत सुरक्षा आणि स्पॅम फिल्टरसह ईमेल सर्विस प्रदाते वापरा
- 3 ईमेल उघडताना सावधगिरी बाळगा
- 4 ईमेल मोकळेपणा आणि इंटरकनेक्टिव्हिटीकडे विकसित होत आहे हे ओळखा
- 5 मजबूत पासवर्ड वापरा आणि साइन-आउट करणे लक्षात ठेवा

### सर्वोत्तम ऑनलाइन प्रायव्हसी मेथड्स

- 1 वैयक्तिक इनफार्मेशन शेअरिंग कमी करा
- 2 वेबसाइट्सवर ट्रस्टमार्क शोधा आणि त्यांची सत्यता पडताळा
- 3 ऑनलाइन खरेदी करताना तात्पुरत्या क्रेडिट कार्ड क्रमांकांचा विचार करा



- 4 मजबूत पासवर्ड वापरा आणि साइन-आउट करणे लक्षात ठेवा
- 5 तुमचे पासवर्ड वारंवार बदला.
- 6 अँटी-व्हायरस आणि अँटी-स्पायवेअर प्रोटेक्शन वापरा
- 7 ब्राउझर प्रायव्हसी वर्धित करण्याच्या क्षमता आणि पर्यायांचा लाभ घ्या
- 8 तुमचा ब्राउझर आणि इतर साधने अपडेट करा.

### मोबाइल प्रायव्हसी

- 1 मोबाइल डिव्हाइसवर तुमची वैयक्तिक इनफार्मेशन डिव्हाइस चोरी किंवा हरवण्याद्वारे धोक्यात येण्याची शक्यता असते – योग्य खबरदारी घ्या
- 2 तुमचे मोबाईल डिव्हाइस कदाचित तुमच्या स्थानाबद्दल जागरूक असेल आणि तो डेटा ॲप्लिकेशन आणि जाहिरातदारांसोबत शेअर करू शकेल

### सायबर क्राईम कायदा

सायबर जगताच्या युगात जसा संगणकाचा वापर अधिक लोकप्रिय होत गेला, तसतसे तंत्रज्ञानाचाही एक्सपान्शन होत गेला आणि ही संज्ञा 'सायबर' लोकांना अधिक परिचित झाले. इनफार्मेशन तंत्रज्ञान (IT) च्या उल्लांतीने सायबर स्पेसला जन्म दिला ज्यामध्ये इंटरनेट सर्व लोकांना उच्च तंत्रज्ञानाच्या वापरासह कोणतीही इनफार्मेशन, डेटा स्टोरेज, विश्लेषण इत्यादींमध्ये एक्सेस करण्यासाठी समान संधी प्रदान करते. नेटिझन्सच्या संख्येत वाढ झाल्यामुळे, सायबर स्पेसमध्ये तंत्रज्ञानाचा गैरवापर वाढला आहे ज्यामुळे देशांतर्गत आणि आंतरराष्ट्रीय लेयर वरही सायबर गुन्हांना जन्म मिळाला.

जरी क्राईम या शब्दाचा सामान्य अर्थ "कायदेशीर चुकीचा आहे ज्याचा रिझल्ट गुन्हेगारी कार्यवाहीद्वारे केला जाऊ शकतो ज्यामुळे शिक्षा होऊ शकते" तर सायबर गुन्हे "बेकायदेशीर कृत्ये असू शकतात ज्यामध्ये संगणक एकतर एक साधन किंवा टारगेट किंवा दोन्ही आहे".

सायबर तंत्रज्ञानाच्या प्रगतीसह 'डेटा प्रोटेक्शन ऍक्ट, 1970' या स्वरूपात जर्मन राज्य हेसेने 1970 मध्ये जगातील पहिला संगणक स्पेसिफिक कायदा लागू केला. तंत्रज्ञानाचा उदय झाल्यामुळे तंत्रज्ञानाचा गैरवापरही त्याच्या इष्टतम पातळीपर्यंत वाढला आहे आणि त्यानंतर सायबर जगतात गुन्हेगारी कृतींचे नियमन करण्यासाठी आणि तांत्रिक प्रगती सिस्टिम चे प्रोटेक्शन करण्यासाठी कठोर वैधानिक कायदांची आवश्यकता आहे. अशा परिस्थितीत भारतीय संसदेने "इनफार्मेशन तंत्रज्ञान कायदा, 2000" 17 ऑक्टोबर रोजी ई-कॉमर्स, ई-गव्हर्नन्स, ई-बँकिंग तसेच सायबर गुन्हांच्या क्षेत्रातील दंड आणि शिक्षेच्या क्षेत्रातील तंत्रज्ञानाशी निगडित कायदा करण्यासाठी त्याचा संपूर्ण कायदा आहे.

### सायबर गुन्हे वास्तविक

हे हॅकर्स तुमच्या साइटची तोडफोड करणे, गोपनीय इनफार्मेशन पाहणे, व्यापार रहस्ये किंवा इंटरनेटच्या वापरासह बौद्धिक संपत्ती चोरणे असू शकते. यामध्ये 'सर्विस नाकारणे' आणि नियमित रहदारीला तुमच्या साइटवर पोहोचण्यापासून रोखणारे व्हायरस हल्ले यांचाही समावेश असू शकतो. सायबर गुन्हे हे व्हायरसच्या बाबतीत आणि सुरक्षेची संबंधित

सायबर गुन्हांव्यतिरिक्त बाहेरील लोकांपुरते मर्यादित नाहीत जे सहसा स्पेसिफिक कंपनीच्या कर्मचार्यांनी केले जातात जे त्यांच्या फायद्यासाठी कंपनीचा पासवर्ड आणि डेटा स्टोरेज सहजपणे ऍक्सेस करू शकतात. सायबर गुन्हांमध्ये संगणकाच्या वापराने केलेल्या गुन्हेगारी कृतींचाही समावेश होतो ज्या गुन्हांना पुढे कायम ठेवतात जसे की आर्थिक गुन्हे, बेकायदेशीर लेखांची विक्री, पोर्नोग्राफी, ऑनलाइन जुगार, बौद्धिक संपदा गुन्हे, ई-मेल, स्पूफिंग, बनावटगिरी, सायबर बदनामी, सायबर स्टर्किंग, अनधिकृत एक्सेस. संगणक सिस्टिम, इलेक्ट्रॉनिक स्वरूपात असलेल्या इनफार्मेशन ची चोरी, ई-मेल बॉम्बस्फोट, संगणक सिस्टिम चे शारीरिक नुकसान इ.

सायबर गुन्हांचे वर्गीकरण: सायबर गुन्हे जे दिवसेंदिवस वाढत आहेत, प्रत्यक्षात सायबर गुन्हा काय आहे आणि पारंपारिक गुन्हा काय आहे हे सर्च णे फार कठीण आहे, त्यामुळे या संभ्रमातून बाहेर पडण्यासाठी, सायबर गुन्हांचे विविध रेन्ज मध्ये वर्गीकरण केले जाऊ शकते जे खालीलप्रमाणे आहेत:

### व्यक्तीविरुद्ध सायबर गुन्हे:

असे काही गुन्हे आहेत जे व्यक्तीच्या व्यक्तिमत्त्वावर रिझल्ट करतात ते खालीलप्रमाणे परिभाषित केले जाऊ शकतात:

ई-मेल्सद्वारे छळवणूक: पत्रे, फाइल्स आणि फोल्डर्सचे संलग्नक म्हणजे ई-मेलद्वारे पाठवून त्रास देणे हा अतिशय सामान्य टाइप आहे. सध्या सोशल साइट्स म्हणजेच फेसबुक, ट्विटर इत्यादींचा वापर दिवसेंदिवस वाढत असल्याने त्रास देणे सामान्य आहे.

### सायबर-स्टॉकिंग

याचा अर्थ इंटरनेट, ई-मेल, फोन, टेक्स्ट संदेश, वेबकॅम, वेबसाइट किंवा व्हिडिओ यासारख्या संगणक तंत्रज्ञानाच्या वापराद्वारे भीती निर्माण करणारा शारीरिक धोका व्यक्त किंवा निहित असा आहे.

### अश्लील साहित्याचा प्रसार

यात अशोभनीय प्रदर्शन/पोर्नोग्राफी (मुळात बाल पोर्नोग्राफी), या प्रतिबंधित कंटेंट असलेल्या वेब साइटचे होस्टिंग समाविष्ट आहे. या अश्लील गोष्टींमुळे किशोरवयीनांच्या मनाला हानी पोहोचू शकते आणि त्यांचे मन भ्रष्ट किंवा भ्रष्ट होऊ शकते.

### बदनामी

कोणत्याही व्यक्तीचे मेल खाते हॅक करून आणि अज्ञात व्यक्तींच्या मेल खात्यावर असभ्य भाषेचा वापर करून काही मेल पाठवून त्या व्यक्तीची प्रतिष्ठा कमी करण्याच्या हेतूने त्यांच्यावर आरोप करणे हे कृत्य आहे.

### हॅकिंग

याचा अर्थ संगणक सिस्टिम वर अनधिकृत कंट्रोल/एक्सेस आणि हॅकिंगची कृती संपूर्ण डेटा तसेच संगणक प्रोग्राम पूर्णपणे नष्ट करते. हॅकर्स सहसा टेलिकम्युनिकेशन आणि मोबाईल नेटवर्क हॅक करतात.

## क्रॅकिंग

आजपर्यंत ज्ञात असलेल्या सर्वात गंभीर सायबर गुन्हांपैकी हा एक आहे. एखाद्या अनोळखी व्यक्तीने तुमच्या इनफार्मेशन शिवाय आणि संमतीशिवाय तुमच्या संगणक सिस्टिम मध्ये घुसखोरी केली आहे आणि मौल्यवान गोपनीय डेटा आणि इनफार्मेशन शी छेडछाड केली आहे हे जाणून घेणे एक भयानक भावना आहे.

## ई-मेल स्पूफिंग

फसवणूक केलेला ई-मेल एक असे म्हटले जाऊ शकते, जे त्याचे मूळ चुकीचे दर्शवते. हे दर्शवते की ते मूळ आहे ज्यापासून ते वेगळे आहे.

## एसएमएस स्पूफिंग:

स्पूफिंग म्हणजे स्पॅमद्वारे अवरोधित करणे म्हणजे नको त्या अननिमंत्रित संदेश. येथे एक गुन्हेगार मोबाईल फोन नंबरच्या रूपात दुसऱ्याची ओळख चोरतो आणि इंटरनेटद्वारे एसएमएस पाठवतो आणि प्राप्तकर्त्याला पीडितेच्या मोबाइल फोन नंबरवरून एसएमएस प्राप्त होतो. कोणत्याही व्यक्तीविरुद्ध हा अतिशय गंभीर सायबर गुन्हा आहे.

## कार्डिंग

याचा अर्थ खोटी एटीएम कार्डे म्हणजेच डेबिट आणि क्रेडिट कार्डे गुन्हेगारांकडून त्यांच्या आर्थिक फायद्यासाठी पीडितेच्या बँक खात्यातून निष्ठेने पैसे काढणे. अशा टाइप च्या सायबर गुन्हांमध्ये नेहमीच एटीएम कार्डचा अनधिकृत वापर होत असतो.

## चॅटींग आणि फ्रॉड

याचा अर्थ सायबर गुन्हाची कृती करणाऱ्या व्यक्तीने म्हणजे पासवर्ड आणि डेटा स्टोरेज चोरणे हे अपराधी मनाने केले आहे ज्यामुळे फसवणूक आणि फसवणूक होते.

## चाइल्ड पोर्नोग्राफी

यामध्ये अल्पवयीन मुलांचे लैंगिक शोषण करणारी कंटेंट तयार करणे, वितरित करणे किंवा त्यात एक्सेस करणे यासाठी संगणक नेटवर्कचा वापर समाविष्ट आहे.

## धमकीने प्राणघातक हल्ला

धमकीद्वारे हल्ला म्हणजे एखाद्या व्यक्तीला त्यांच्या जीवाची किंवा त्यांच्या कुटुंबियांच्या जीवाची भीती असलेल्या संगणक नेटवर्कच्या वापराद्वारे म्हणजे ई-मेल, व्हिडिओ किंवा फोनद्वारे धमकी देणे होय.

## व्यक्तीच्या मालमत्तेविरुद्ध गुन्हे:

आंतरराष्ट्रीय व्यापारात झपाट्याने वाढ होत असल्याने व्यवसाय आणि ग्राहक पारंपरिक कागदी डॉक्युमेंट्स च्या ऐवजी इलेक्ट्रॉनिक स्वरूपात इनफार्मेशन तयार करण्यासाठी, प्रसारित करण्यासाठी आणि स्टोर करण्यासाठी संगणकाचा वापर करत आहेत. असे काही गुन्हे आहेत जे व्यक्तीच्या मालमत्तेवर रिझल्ट करतात जे खालीलप्रमाणे आहेत:

## बौद्धिक संपदा गुन्हे

बौद्धिक मालमत्तेमध्ये अधिकारांचा एक समूह असतो. कोणतीही

बेकायदेशीर कृती ज्याद्वारे मालक त्याच्या हक्कांपासून पूर्णपणे किंवा अंशतः वंचित ठेवला जातो तो गुन्हा आहे. सॉफ्टवेअर पायरसी, कॉपीराइटचे उल्लंघन, ट्रेडमार्क, पेटंट, डिझाईन्स आणि सर्व्हिस मार्कचे उल्लंघन, कॉम्प्युटर सोर्स कोडची चोरी इ.

## सायबर स्कॅटिंग

याचा अर्थ असा की जेथे दोन व्यक्ती एकाच डोमेन नावासाठी दावा करतात की त्यांनी नाव नोंदणीकृत केले आहे असा दावा करून ते दुसऱ्यासमोर वापरण्याच्या अधिकाराने किंवा पूर्वीसारखे काहीतरी वापरून.

## सायबर तोडफोड

तोडफोड म्हणजे जाणूनबुजून दुसऱ्याच्या मालमत्तेचा नाश करणे किंवा नुकसान करणे. अशाप्रकारे सायबर तोडफोड म्हणजे जेव्हा नेटवर्क सर्विस बंद होते किंवा व्यत्यय आणली जाते तेव्हा डेटा नष्ट करणे किंवा नुकसान करणे. कोणत्याही व्यक्तीच्या संगणकाला झालेल्या कोणत्याही टाइप ची शारीरिक हानी त्याच्या कार्यक्षेत्रात समाविष्ट असू शकते. ही कृत्ये चोरीचे फॉरमॅट घेऊ शकतात संगणक, संगणकाचा काही पार्ट किंवा संगणकाला जोडलेले पेरीफेरल.

**संगणक सिस्टिम हॅकिंग:** हॅकटिव्हिझममध्ये प्रसिद्ध ट्विटर, ब्लॉगिंग प्लॅटफॉर्मवर अनधिकृत एक्सेस /कंट्रोल समाविष्ट असलेल्यांवर हल्ला होतो. हॅकिंग अॅक्टिव्हिटीमुळे संगणकासह डेटाचेही नुकसान होणार आहे. तसेच रिसर्च न स्पेशल तः असे सूचित करते की ते हल्ले प्रामुख्याने आर्थिक लाभासाठी आणि स्पेसिफिक व्यक्ती किंवा कंपनीची प्रतिष्ठा कमी करण्यासाठी नव्हते.

**व्हायरस प्रसारित करणे:** व्हायरस हे असे प्रोग्राम आहेत जे स्वतःला संगणक किंवा फाइलशी संलग्न करतात आणि नंतर स्वतःला इतर फाइल्स आणि नेटवर्कवरील इतर संगणकांमध्ये प्रसारित करतात. ते सहसा संगणकावरील डेटा बदलून किंवा हटवून प्रभावित करतात. कृमी हल्ल्यांमुळे व्यक्तींच्या संगणकीकृत सिस्टिम वर रिझल्ट होतो.

**सायबर अतिक्रमण:** याचा अर्थ मालकाच्या योग्य अधिकृतेशिवाय एखाद्याच्या संगणकावर एक्सेस करणे आणि वायरलेस इंटरनेट कनेक्शन वापरून डेटा किंवा सिस्टमला त्रास देणे, बदलणे, गैरवापर किंवा नुकसान होत नाही.

**इंटरनेट वेळ चोरी:** मुळात इंटरनेट वेळ चोरी हॅकिंग इंटरनल येते. हा अनधिकृत व्यक्तीने केलेला वापर आहे, दुसऱ्या व्यक्तीने दिलेल्या इंटरनेट तासांचा. हॅक करून किंवा बेकायदेशीर मार्गाने एक्सेस मिळवून इतर कोणाच्या तरी ISP युजर आयडी आणि पासवर्डमध्ये एक्सेस मिळवणारी व्यक्ती दुसऱ्या व्यक्तीच्या इनफार्मेशन शिवाय इंटरनेटवर एक्सेस करण्यासाठी त्याचा वापर करते. क्वचित वापर करूनही, इंटरनेटचा वेळ वारंवार रिचार्ज करावा लागल्यास, तुम्ही वेळेची चोरी ओळखू शकता.

## सरकार विरुद्ध सायबर गुन्हे:

इंटरनेट सुविधा वापरून आंतरराष्ट्रीय सरकारांना धमकावण्याचा हेतू असलेल्या व्यक्तींच्या गटाद्वारे काही गुन्हे केले जातात. यात हे समाविष्ट आहे:

## सायबर दहशतवाद

सायबर दहशतवाद ही देशांतर्गत तसेच ग्लोबल चिंतेची एक मोठी ज्वलंत समस्या आहे. इंटरनेटवरील या दहशतवादी हल्ल्यांचे सामान्य फॉरमॅट म्हणजे सर्विस हल्ले, द्वेषयुक्त वेबसाइट्स आणि द्वेषयुक्त ई-मेल, सेन्सिटिव्ह संगणक नेटवर्कवरील हल्ले इत्यादींना वितरित नकार देणे. सायबर दहशतवादी कारवाया राष्ट्रचे सार्वभौमत्व आणि अखंडता धोक्यात आणतात.

## सायबर युद्ध

हे तोडफोड आणि हेरगिरी करण्यासाठी राजकीयदृष्ट्या प्रेरित हॅकिंगचा रेफरेन्स देते. हा इनफार्मेशन युद्धाचा एक टाइप आहे जो काहीवेळा पारंपारिक युद्धाशी साधर्म्य म्हणून पाहिला जातो, जरी हे साधर्म्य तिची अचूकता आणि राजकीय प्रेरणा दोन्हीसाठी विवादास्पद आहे.

## पायरेटेड सॉफ्टवेअरचे वितरण

याचा अर्थ एका संगणकावरून दुसऱ्या संगणकावर पायरेटेड सॉफ्टवेअर वितरित करणे आणि सरकारचा डेटा आणि अधिकृत रेकॉर्ड नष्ट करणे.

## अनधिकृत इनफार्मेशन चा ताबा:

इंटरनेटच्या साहाय्याने दहशतवाद्यांकडून कोणतीही इनफार्मेशन मिळवणे आणि राजकीय, धार्मिक, सामाजिक, वैचारिक उद्दिष्टांसाठी ती इनफार्मेशन ताब्यात घेणे खूप सोपे आहे.

## मोठ्या प्रमाणावर समाजाविरुद्ध सायबर गुन्हे:

सायबर स्पेसला हानी पोहोचवण्याच्या उद्देशाने केलेले बेकायदेशीर कृत्य मोठ्या संख्येने व्यक्तींना प्रभावित करेल. या गुन्हांमध्ये हे समाविष्ट आहे:

**चाइल्ड पोर्नोग्राफी:** यामध्ये अल्पवयीन मुलांचे लैंगिक शोषण करणारी कंटेंट तयार करणे, वितरित करणे किंवा त्यात एक्सेस करणे यासाठी संगणक नेटवर्कचा वापर समाविष्ट आहे. यात अशोभनीय प्रदर्शन आणि अश्लीलता यासंबंधीच्या क्रियाकलापांचा देखील समावेश आहे.

**सायबर ट्रॅफिकिंग:** हे अंमली पदार्थ, मानव, शस्त्रास्त्रे इत्यादींची तस्करी असू शकते ज्यामुळे मोठ्या प्रमाणात व्यक्ती प्रभावित होतात. सायबर स्पेसमध्ये होणारी तस्करी हा देखील एक गंभीर गुन्हा आहे.

## ऑनलाइन जुगार

ऑनलाइन फसवणूक आणि फसवणूक हा सर्वात फायदेशीर व्यवसायांपैकी एक आहे जो आज सायबर स्पेसमध्ये वाढत आहे. क्रेडिट कार्डचे गुन्हे, कंत्राटी गुन्ह्ये, नोकरी देणे इत्यादी अनेक प्रकारे उघडकीस आली आहेत.

## आर्थिक गुन्हे

हा टाइप सामान्य आहे कारण नेटवर्किंग साइट्स आणि फोन नेटवर्किंगच्या वापरकर्त्यांमध्ये झपाट्याने वाढ होत आहे जेथे गुन्हेगार इंटरनेटद्वारे बोगस मेल किंवा संदेश पाठवून हल्ला करण्याचा प्रयत्न करेल. उदा: बेकायदेशीरपणे पासवर्ड मिळवून क्रेडिट कार्ड वापरणे.

## खोटारडेपणा

याचा अर्थ धमकीचे मेल पाठवून मोठ्या संख्येने लोकांची फसवणूक करणे, कारण ऑनलाइन व्यवसाय व्यवहार ही आजच्या जीवनस्टाइल ची सवय झाली आहे.

## कोणावर रिझल्ट होतो

सायबर गुन्हे नेहमीच कोणत्याही आकाराच्या कंपन्यांना प्रभावित करतात कारण जवळजवळ सर्व कंपन्या ऑनलाइन सबस्थिती मिळवतात आणि तंत्रज्ञानातील जलद नफ्याचा फायदा घेतात परंतु त्यांच्या सुरक्षिततेच्या जोखमीकडे जास्त लक्ष दिले जाते. आधुनिक सायबर जगात सायबर गुन्हे ही एक मोठी समस्या आहे जी व्यक्ती तसेच समाजावरही रिझल्ट करत आहे.

## सायबर कायद्याची गरज

इनफार्मेशन तंत्रज्ञान जगभर पसरले आहे. संगणकाचा वापर प्रत्येक क्षेत्रात केला जातो ज्यामध्ये सायबरस्पेस आर्थिक वाढ आणि मानवी विकासासाठी सर्वांना समान संधी प्रदान करते. सायबर स्पेसचा युजर जसजसा वैविध्यपूर्ण होत आहे आणि ऑनलाइन परस्परसंवादाची रेन्ज एक्सपान्शन त आहे, तसतसे सायबर गुन्हांमध्ये एक्सपान्शन होत आहे, म्हणजे ऑनलाइन करारांचे उल्लंघन, ऑनलाइन टॉर्ट्स आणि गुन्हे इत्यादी. द सायबर स्पेस अथॉरिटी सायबरशी संबंधित गुन्हेगारी क्रियाकलापांचे नियमन करण्यासाठी आणि सायबर गुन्हातील पीडित व्यक्तीला न्यायाचे चांगले एडमिनिस्ट्रेशन प्रदान करण्यासाठी. आधुनिक सायबर तंत्रज्ञानाच्या जगात सायबर गुन्हांचे नियमन करणे अत्यंत आवश्यक आहे आणि सर्वात महत्त्वाचे म्हणजे सायबर दहशतवाद आणि हॅकर्सच्या बाबतीत सायबर कायदा अधिक कठोर करणे आवश्यक आहे.

## संगणक सिस्टिम च्या नुकसानीसाठी दंड

'इनफार्मेशन तंत्रज्ञान कायदा, 2000' च्या कलम 43 नुसार जो कोणी संगणक सिस्टिम च्या संपूर्ण डेटाचे नुकसान करण्याच्या उद्देशाने संगणक नष्ट करणे, हटवणे, बदलणे आणि व्यत्यय आणणे किंवा व्यत्यय आणण्याचे कोणतेही कृत्य करतो. संगणकाचा मालक, उपायाच्या मार्गाने प्रभावित झालेल्या व्यक्तीला 1 कोटीपर्यंत दंड भरण्यास जबाबदार असेल. कलम: 43A नुसार जे 'इनफार्मेशन तंत्रज्ञान (दुरुस्ती) कायदा, 2008' द्वारे समाविष्ट केले गेले आहे जेथे एखादी संस्था कॉर्पोरेट केंद्र सरकारने प्रदान केलेल्या व्यक्तींच्या डेटाची देखरेख आणि प्रोटेक्शन करत आहे, जर काही निष्काळजीपणाचे कृत्य किंवा प्रोटेक्शन करण्यात अपयश आले तर डेटा/इनफार्मेशन नंतर बॉडी कॉर्पोरेट प्रभावित व्यक्तीला नुकसान भरपाई देण्यास जबाबदार असेल. आणि कलम 66 'संगणक सिस्टिम सह हॅकिंग'शी संबंधित आहे आणि 3 वर्षांपर्यंत कारावास किंवा दंडाची तरतूद आहे, जी 2 वर्षांपर्यंत वाढू शकते किंवा दोन्ही.

## क्लाउड सर्विस सह कार्य करणे (Working with Cloud Services)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- क्लाउड कॅम्प्युटिंगचा परिचय
- क्लाउड संगणनाचे टाइप
- क्लाउडकॉम्प्युटिंगचे तोटे आणि फायदे.

### क्लाउड कॉम्प्युटिंगचा परिचय

क्लाउड कॉम्प्युटिंग: क्लाउड कॉम्प्युटिंग सर्विस चे फायदे, तोटे आणि टाइप  
क्लाउड कॉम्प्युटिंग हे आजच्या व्यवसायातील सर्वात लोकप्रिय कॅचफ्रेसेस आहे. याने संस्थांची इनफार्मेशन स्टोर करणे, एक्सेस करणे आणि सामायिक करणे, कोलायब्रेशन करणे आणि संगणकीय रिसोर्सेस मॅनेज करणे यात बदल केले आहेत. इंटरनेटच्या आगमनाने, क्लाउड कॅम्प्युटिंगने कंपन्यांना परंपरागत ऑन-प्रिमाइसेस IT पायाभूत सुविधांपेक्षा वर येण्याची परवानगी देऊन व्यवसाय चालवण्याचे नवीन पाथ प्रदान केले आहेत.

क्लाउड कॅम्प्युटिंग आधुनिक व्यवसायांना लवचिकता, कार्यक्षमता, स्केलेबिलिटी, सुरक्षा, वाढीव कोलायब्रेशन आणि कमी खर्च ऑफर करते. कोविड-19 साठीच्या रोगाने क्लाउड दत्तक घेण्यास गती दिली असताना, 2022 मध्ये क्लाउड तंत्रज्ञानावरील अवलंबित्व कायम राहणार आहे, स्पेशल त: संकरित कार्य केंद्रस्थानी आहे. त्यामुळे, एखादी संस्था आधीच क्लाउड सर्विस वापरत असली किंवा येल्या वर्षात योजना आखत असली तरी, क्लाउड पॉवर सोल्यूशन्सचा पूर्ण लाभ घेण्यासाठी क्लाउड कॉम्प्युटिंगची बेसिक इनफार्मेशन समजून घेणे अत्यावश्यक आहे.

या ब्लॉगमध्ये, आम्ही क्लाउड कॉम्प्युटिंग म्हणजे नेमके काय आहे, ते कसे कार्य करते, त्याचे फायदे आणि तोटे आणि कंपनी त्यांच्या SaaS डेटाचे अधिक चांगल्या प्रकारे प्रोटेक्शन कसे करू शकतात याचा सर्च घेऊ.



### क्लाउड कॉम्प्युटिंग म्हणजे काय?

ZDNet च्या मते, "क्लाउड कॅम्प्युटिंग म्हणजे मागणीनुसार संगणकीय सर्विस - ऍप्लिकेशन्सपासून स्टोरेज आणि प्रोसेसिंग पॉवरपर्यंत - स्पेशल त: इंटरनेटवर आणि तुम्ही जाता-जाता पगाराच्या आधारावर डिलिव्हरी केली जाते."

सोप्या भाषेत, क्लाउड इंटरनेटचा रेफरेन्स देते. जेव्हा संस्था त्यांच्या डिव्हाइसच्या हार्ड ड्राइव्हवर किंवा ऑन-प्रिमाइसेस IT इन्फ्रास्ट्रक्चरवर अवलंबून न राहता व्हर्च्युअल डेटा सेंटरमध्ये डेटा स्टोर करतात किंवा इंटरनेट कनेक्शन वापरून प्रोग्राममध्ये एक्सेस करतात, तेव्हा याचा अर्थ ते क्लाउडमध्ये कार्यरत आहेत.

क्लाउड कॅम्प्युटिंग "थर्डपार्टी डेटा सेंटरमधील सर्व्हर" किंवा संपूर्ण सर्व्हरलेस वर्कलोड्स इतके सोपे असू शकते जे अमर्यादपणे वाढवता येण्याजोगे आणि भू-रिडंडंट आहे. क्लाउड सर्व्हर आणि सर्विस स्केलेबल आणि लवचिक आहेत.

### क्लाउड कॉम्प्युटिंग कसे कार्य करते?

क्लाउड कॉम्प्युटिंग हे संगणकीय संसाधनांचे वितरण आहे, जसे की IT पायाभूत सुविधा किंवा इंटरनेटवर डेटा सेंटर. हे मॉडेल व्यवसायांना स्टोरेज स्पेस भाड्याने देण्याची किंवा क्लाउड सर्विस प्रदात्याकडून सॉफ्टवेअर प्रोग्राममध्ये एक्सेस करण्याची परवानगी देते, त्याऐवजी त्यांची स्वतःची IT इन्फ्रास्ट्रक्चर किंवा डेटा सेंटर तयार करणे आणि देखरेख करणे. क्लाउड कॉम्प्युटिंग सर्विस वापरण्याचा एक मोठा फायदा म्हणजे कंपन्या केवळ ते वापरत असलेल्या संसाधनांसाठी पैसे देतात.

त्याच्या तांत्रिक बाबी चांगल्या प्रकारे समजून घेण्यासाठी, क्लाउड संगणन प्रक्रिया फ्रंटएंड आणि बॅकएंडमध्ये विपार्ट ली जाऊ शकतात. फ्रंटएंड एलिमेंट वापरकर्त्यांना इंटरनेट ब्राउझरद्वारे किंवा क्लाउड संगणन एप्लिकेशन वापरून क्लाउडमध्ये स्टोर डेटा आणि प्रोग्राममध्ये एक्सेस करण्याची परवानगी देतो. बॅकएंडमध्ये सर्व्हर, संगणक आणि डेटाबेस असतात जे डेटा स्टोर करतात.

### क्लाउड कॉम्प्युटिंगचा इतिहास

टेक्नॉलॉजी रिव्हनुसार, "क्लाउड कॅम्प्युटिंग" या वाक्यांशाचा प्रथम उल्लेख 1996 मध्ये कॉम्पॅक इंटरनल डॉक्युमेंट्स त करण्यात आला होता.

1999 हे वर्ष क्लाउड कॉम्प्युटिंगसाठी मैलाचा दगड ठरले जेव्हा सेल्सफोर्स ही इंटरनेटवर ऍंटरप्राइझ ऍप्लिकेशन्स वितरीत करणारी पहिली कंपनी बनली. सॉफ्टवेअर-अ-सर्व्हिस (सास) चीही ही सुरुवात होती.

2002 मध्ये, Amazon ने Amazon Web Services (AWS) लॉंच केले, जी क्लाउड कॉम्प्युटिंगमधील आणखी एक महत्त्वपूर्ण प्रगती होती. क्लाउड-आधारित सर्विस च्या संचमध्ये स्टोरेज, गणना आणि अगदी मानवी बुद्धिमत्ता समाविष्ट आहे. 2006 मध्ये, Amazon ने इलास्टिक कॉम्प्युट क्लाउड (EC2)

लॉच केले, ज्यामुळे व्यवसायांना तसेच व्यक्तींना व्हर्च्युअल कॉम्प्युटर भाड्याने देण्याची आणि त्यांचे स्वतःचे संगणक एप्लिकेशन चालवण्याची परवानगी दिली.

Google Workspace (आता Google Workspace) ने ब्राउझर-आधारित एंटरप्राइझ ऍप्लिकेशन प्रदान करण्यास सुरुवात केल्याने 2009 साली क्लाउड संगणनामध्ये आणखी एक मोठा टप्पा गाठला गेला. त्याच वर्षी मायक्रोसॉफ्टने मायक्रोसॉफ्ट ॲझ्युरसह क्लाउड कॉम्प्युटिंग क्षेत्रात एक्सेस केला आणि लवकरच ओरॅकल आणि एचपी सारख्या कंपन्यांनी त्याचे अनुकरण केले.

### क्लाउड कॉम्प्युटिंगची उदाहरणे काय आहेत?

क्लाउड कंप्युटिंगमध्ये व्हर्च्युअल मशीन्सपासून डेटाबेसपर्यंत सर्व सर्व्हरलेस ऍप्लिकेशन्सचा समावेश होतो. क्लाउड कॉम्प्युटिंगच्या काही उदाहरणांमध्ये हे समाविष्ट आहे:

**Salesforce:** Salesforce.com एक SaaS प्रदाता आहे जो ग्राहक संबंध मॅनेजमेंट (CRM) मध्ये माहिर आहे. कंपनी विपणन, विक्री, ग्राहक सर्विस इ. एलाइनमेंट करण्यात मदत करण्यासाठी एंटरप्राइझ ऍप्लिकेशन प्रदान करते आणि वापरकर्त्यांना कुठूनही काम करण्याची परवानगी देते.

**डिजिटल ओशन:** ही कंपनी सॉफ्टवेअर डेव्हलपरसाठी न्यूयॉर्क-आधारित इन्फ्रास्ट्रक्चर-ए-ए-सर्व्हिस (IaaS) प्रदाता आहे. मल्टिपल क्लाउड सर्व्हरवर एकाच वेळी चालणारे एप्लिकेशन तैनात आणि स्केल करण्यासाठी व्यवसाय डिजिटल ओशन वापरतात.

**Microsoft Azure:** Microsoft Azure हे प्लॅटफॉर्म-एज-ए-सर्व्हिस (PaaS) चे उत्तम उदाहरण आहे जे संपूर्ण ऍप्लिकेशन डेव्हलपमेंट लाइफसायकलला सपोर्ट देते, अगदी डेव्हलपमेंटपासून डिप्लॉयमेंटपर्यंत आणि पुढे. Azure विकसकांना अनेक साधने, भाषा आणि फ्रेमवर्क प्रदान करते.

**ड्रॉपबॉक्स:** ड्रॉपबॉक्स ही क्लाउड-आधारित फाइल होस्टिंग सर्विस आहे जी वापरकर्त्यांना त्यांच्या डिव्हाइसवर फायली संचयित आणि समक्रमित करण्यास अनुमती देते जेणेकरून ते त्यांना कुठूनही एक्सेस करू शकतील. हे वापरकर्त्यांना इंटरनेटद्वारे प्रतिमा आणि व्हिडिओसह मोठ्या फायली सामायिक करण्याची परवानगी देते, प्रभावी कोलायब्रेशन सुलभ करते.

### क्लाउड कॉम्प्युटिंगचे महत्त्व काय आहे?

क्लाउड कंप्युटिंग अस्तित्वात येण्यापूर्वी, कंपन्यांना त्यांच्या फिजिकल पीसी किंवा ऑन-प्रीमाइसेस सर्व्हरवर एप्लिकेशन किंवा प्रोग्राम डाउनलोड करणे आवश्यक होते जेणेकरून ते वापरता येतील. कोणत्याही संस्थेसाठी, स्वतःचे आयटी इन्फ्रास्ट्रक्चर किंवा डेटा सेंटर तयार करणे आणि मॅनेज करणे हे एक मोठे आव्हान आहे. ज्यांच्याकडे स्वतःचे डेटा सेंटर आहेत त्यांच्यासाठीही, मोठ्या प्रमाणात आयटी एडमिनिस्ट्रेटर आणि रिसोर्सस वाटप करणे हे एक संघर्ष आहे.

क्लाउड कॉम्प्युटिंग आणि व्हर्च्युअलायझेशनचा परिचय तंत्रज्ञान उद्योगाच्या इतिहासात एक नमुना बदल होता. त्यांची स्वतःची IT इन्फ्रास्ट्रक्चर तयार करणे आणि मॅनेज करणे आणि सर्व्हर, पॉवर आणि रिअल इस्टेट

इत्यादीसाठी पैसे देण्याऐवजी, क्लाउड कॉम्प्युटिंग व्यवसायांना क्लाउड सर्विस प्रदात्यांकडून संगणकीय रिसोर्सस भाड्याने देण्याची परवानगी देते. हे व्यवसायांना मोठ्या आगाऊ खर्च आणि त्यांची स्वतःची डेटा केंद्रे मॅनेज करण्याची जटिलता टाळण्यास मदत करते. क्लाउड सर्विस भाड्याने देऊन, कंपन्या केवळ संगणकीय रिसोर्सस आणि डिस्क स्पेस यांसारख्या गोष्टींसाठी पैसे देतात. हे कंपन्यांना अधिक अचूकतेसह खर्चाचा अंदाज लावू देते.

क्लाउड सर्विस प्रदाते आयटी पायाभूत सुविधांचे मॅनेजमेंट आणि देखरेख करण्याचे खूप मोठे काम करत असल्याने, यामुळे व्यवसायांसाठी बराच वेळ, मेहनत आणि पैसा वाचतो. क्लाउड संस्थांना आवश्यकतेनुसार त्यांची संगणकीय पायाभूत सुविधा अखंडपणे वाढवण्याची किंवा कमी करण्याची क्षमता देखील देते. पारंपारिक ऑन-प्रीमाइसेस डेटा सेंटर मॉडेलच्या तुलनेत, क्लाउड इंटरनेट कनेक्टिव्हिटीसह कोठूनही आणि कोणत्याही डिव्हाइसवर डेटामध्ये सहज एक्सेस प्रदान करते, ज्यामुळे प्रभावी कोलायब्रेशन आणि वर्धित उत्पादकता सक्षम होते.

### क्लाउड कॉम्प्युटिंगचे सर्वात सामान्य यूजेस कोणते आहेत?

स्टार्टअपपासून मोठ्या कॉर्पोरेशन आणि सरकारी एजन्सीपर्यंत, प्रत्येक संस्था वर्कफ्लो सुव्यवस्थित करण्यासाठी, कॅम्प्युनिकेशन, उत्पादकता, सर्विस वितरण आणि बरेच काही सुधारण्यासाठी तंत्रज्ञान सर्विस मध्ये एक्सेस करण्यासाठी क्लाउडचा वापर करते. क्लाउड कॉम्प्युटिंगचे काही सामान्य यूजेस खाली लिस्ट बद्ध केले आहेत.

- **स्टोरेज:** क्लाउड कॉम्प्युटिंगचा सर्वात सामान्य वापर म्हणजे फाइल स्टोरेज. डेटा संचयित करण्यासाठी आणि त्यात एक्सेस करण्यासाठी अनेक पर्याय आहेत, जसे की PC वर हार्ड ड्राइव्ह, एक्सटर्नल हार्ड ड्राइव्ह्स, USB ड्राइव्ह इ., क्लाउड स्टोरेज व्यवसायांना इंटरनेट कनेक्शनसह कोठूनही आणि कोणत्याही डिव्हाइसवर डेटा अखंडपणे एक्सेस करण्यास सक्षम करते. Amazon S3, DropBox किंवा One-Drive सारख्या क्लाउड स्टोरेज सर्विस डेटावर सुरक्षित एक्सेस प्रदान करतात आणि व्यवसायांना त्यांच्या गरजेनुसार स्टोरेज स्पेस अपस्केल आणि डाउनस्केल करण्याची परवानगी देतात.
- **डेटाबेस:** क्लाउड डेटाबेस हे आणखी एक लोकप्रिय व्यवसाय वापर प्रकरण आहे. IBM क्लाउड डेटाबेसची व्याख्या "क्लाउड प्लॅटफॉर्मद्वारे तयार केलेली आणि एक्सेस केलेली डेटाबेस सर्विस" म्हणून करते. क्लाउड डेटाबेस पारंपारिक डेटाबेस प्रमाणेच बहुतेक फंक्शन प्रदान करतो, परंतु अतिरिक्त फायदे जसे की लवचिकता, खर्च बचत, फेलओव्हर सपोर्ट, स्पेशल कौशल्य आणि बरेच काही.
- **वेब एप्लिकेशन:** वेब ऍप्लिकेशन्स हे आज व्यवसायांसाठी आवश्यक साधन आहे. क्लाउड तंत्रज्ञानाद्वारे समर्थित, कोणीही वेब ब्राउझर वापरून वेब-आधारित ऍप्समध्ये एक्सेस करू शकतो, इनफार्मेशन वर त्वरित दूरस्थ एक्सेस प्रदान करतो. हे व्यवसाय व्यावसायिकांना ग्राहकांशी संवाद साधण्यास आणि प्रवासात असताना त्यांना आवश्यक इनफार्मेशन प्रदान करण्यास अनुमती देते आणि त्यांना कोठूनही सहकाऱ्यांसोबत कोलायब्रेशन करण्यास मदत करते.

- **कोलायब्रेशन** : त्याच्या सुलभ एक्सेस योग्यतेमुळे, एकत्रीकरण, लवचिकता, सुरक्षितता आणि वापरणी सुलभतेमुळे, Microsoft 365 आणि Google Workspace सारखी क्लाउड-आधारित साधने, विभागांमध्ये इंटरनल आणि बाहेरून क्लायंटसह कोलायब्रेशन करू पाहणाऱ्या व्यवसायांसाठी स्पष्ट पर्याय बनले आहेत. जीमेल, गुगल डॉक्स, मायक्रोसॉफ्ट आउटलुक, मायक्रोसॉफ्ट वर्ड, टीम्स इ., कोलायब्रेशन आणि उत्पादकता वाढविण्यासाठी डिझाइन केलेली शक्तिशाली व्यवसाय साधने आहेत.
- **SaaS एप्लिकेशन** : सेल्सफोर्स सारखे सॉफ्टवेअर-एज-ए-सर्व्हिस (सास) ॲप्लिकेशन्स, व्यवसायांना डेटा संचयित, मॅनेज आणि देखरेख करण्यास तसेच ऑटोमॅटिक विपणन आणि क्लायंटचे कार्यक्षमतेने मॅनेजमेंट करण्यास अनुमती देतात. SaaS सोल्यूशन्स अत्यंत कार्यक्षम आहेत आणि त्यांना सॉफ्टवेअर आणि/किंवा हार्डवेअर मॅनेजमेंट ची आवश्यकता नाही.

### क्लाउड कॉम्प्युटिंगचे विविध टाइप कोणते आहेत?

क्लाउड कॉम्प्युटिंगचे चार मुख्य टाइप आहेत: सार्वजनिक, प्राइवेट, संकरित आणि मल्टीक्लाउड.

सार्वजनिक क्लाउड

**VMware** पब्लिक क्लाउडची व्याख्या "एक IT मॉडेल आहे जिथे ऑनडिमांड कंप्युटिंग सर्विस आणि पायाभूत सुविधा तृतीय-पक्ष प्रदात्याद्वारे मॅनेज केल्या जातात आणि सार्वजनिक इंटरनेट वापरून मल्टिपल संस्थांसह सामायिक केल्या जातात." क्लाउड सर्विस प्रदाते विविध ऑफर देतात

इन्फ्रास्ट्रक्चर-एज-ए-सर्व्हिस (IaaS), प्लॅटफॉर्म-ए-ए-सर्व्हिस (PaaS) आणि SaaS सारख्या सर्विस ज्या व्यक्ती आणि व्यवसायांना या सर्विस मासिक भाड्याने देतात किंवा प्रति-वापर आधारावर देतात. Amazon Web Services (AWS), Microsoft Azure, Google Cloud, Alibaba Cloud आणि IBM Cloud हे शीर्ष पाच क्लाउड प्रदाते आहेत.

### प्राइवेट क्लाउड

प्राइवेट क्लाउड किंवा इंटरनल क्लाउड हे आहे जेथे IT पायाभूत सुविधा (हार्डवेअर आणि सॉफ्टवेअर रिसोर्सेस) पूर्णपणे एकाच संस्थेसाठी समर्पित असतात, सार्वजनिक क्लाउडच्या विपरीत जेथे संगणकीय रिसोर्सेस मल्टिपल भाडेकरूंमध्ये सामायिक केली जातात. प्राइवेट क्लाउड वातावरण ज्या व्यवसायांसाठी नियामक आवश्यकता पूर्ण करणे, सुरक्षा आणि कंट्रोल याला प्राधान्य आहे त्यांच्यासाठी आदर्श आहे. पारंपारिकपणे, प्राइवेट क्लाउड कंपनीच्या डेटा सेंटरमध्ये होस्ट केला जातो आणि स्वतःचे हार्डवेअर वापरतो. तथापि, एखादी संस्था संगणकीय रिसोर्सेस दूरस्थपणे मॅनेज करणाऱ्या तृतीय पक्ष प्रदात्याकडे होस्टिंग आउटसोर्स करू शकते.

### हायब्रिड क्लाउड

हायब्रीड क्लाउड हे सार्वजनिक क्लाउड आणि प्राइवेट क्लाउड दोन्ही वातावरणाचे संयोजन आहे. व्यवसाय त्यांच्या गणना क्षमतेला पूरक म्हणून या मॉडेलचा वापर करतात. जेव्हा प्राइवेट क्लाउडची क्षमता शिखरावर

पोहोचते तेव्हा प्राइवेट क्लाउडची क्षमता वाढविण्यासाठी व्यवसाय सार्वजनिक क्लाउडचा फायदा घेऊ शकतात. हायब्रीड क्लाउड व्यवसायांना रहदारी किंवा सेवेच्या मागणीनुसार गणना क्षमता वर किंवा कमी करण्यास सक्षम करते. हे नवीन सर्व्हर खरेदी करण्याची आणि देखरेख करण्याची गरज दूर करते, ज्यामुळे व्यवसायांना खर्च, वेळ आणि मेहनत वाचवता येते.

### मल्टीक्लाउड

मल्टीक्लाउड हे ढगांचे संयोजन वापरण्याची प्रथा आहे - दोन किंवा अधिक सार्वजनिक किंवा प्राइवेट क्लाउड, किंवा अनेक क्लाउड प्रदात्यांकडून दोन्हीचे संयोजन. मल्टीक्लाउड दृष्टीकोन व्यवसायांना विविध क्लाउड विक्रेत्यांकडून त्यांचे बजेट, तांत्रिक आवश्यकता, भौगोलिक पोजिशन इत्यादींच्या आधारावर सर्वोत्तम सर्विस निवडण्याची परवानगी देतो. हे मॉडेल व्यवसायांना वेगवेगळ्या उद्देशांसाठी वेगवेगळे क्लाउड वापरण्यास सक्षम करते. उदाहरणार्थ, एखादी संस्था सॉफ्टवेअर डेव्हलपमेंट आणि टेस्टिंग साठी एक क्लाउड, डेटा बँकअप आणि आपत्ती पुनर्प्राप्तीसाठी दुसरा क्लाउड आणि डेटा विश्लेषणासाठी दुसरा क्लाउड वापरू शकते.

### क्लाउड कॉम्प्युटिंग सर्विस चे तीन भिन्न टाइप कोणते आहेत?

क्लाउड कॉम्प्युटिंग सर्विस चे तीन टाइप म्हणजे इन्फ्रास्ट्रक्चर-एज-ए-सर्व्हिस (आयएएएस), प्लॅटफॉर्म-एज-ए-सर्व्हिस (पीएएस) आणि सॉफ्टवेअर-एज-ए-सर्व्हिस (सास).

### सर्विस म्हणून पायाभूत सुविधा (IaaS)

IaaS ही क्लाउड कंप्युटिंग सर्विस आहे जिथे क्लाउड प्रदाते इंटरनेटवर व्हर्च्युअलाइज्ड कॉम्प्युटिंग इन्फ्रास्ट्रक्चर वितरित करतात आणि मॅनेज करतात. इन-हाऊस आयटी इन्फ्रास्ट्रक्चर तयार करण्याऐवजी, व्यवसाय आवश्यक संसाधनांमध्ये एक्सेस करू शकतात, जसे की ऑपरेटिंग

मागणीनुसार सिस्टम, नेटवर्किंग, स्टोरेज स्पेस, डेव्हलपमेंट टूल्स इ. हे हार्डवेअर आणि सॉफ्टवेअरच्या खर्चात बचत करते तसेच आयटी कर्मचार्यांचा भार कमी करते.

### प्लॅटफॉर्म-ए-ए-सर्व्हिस (PaaS)

PaaS व्यवसायांना बेसिक पायाभूत सुविधांबद्दल काळजी न करता सॉफ्टवेअर ॲप्लिकेशन्स आणि सर्विस च्या विकास, सबयोजन आणि मॅनेजमेंट वर लक्ष केंद्रित करण्यास अनुमती देते कारण क्लाउड प्रदाते हेवी लिफ्टिंग करतात. PaaS सह, डेव्हलपर आणि प्रोग्रामर केवळ IT इन्फ्रास्ट्रक्चरच नाही तर ॲप्लिकेशन/सॉफ्टवेअर प्लॅटफॉर्म आणि सोल्यूशन स्टॅकमध्ये एक्सेस मिळवतात. PaaS च्या काही उदाहरणांमध्ये AWS Elastic Beanstalk, Google App Engine आणि Microsoft Azure यांचा समावेश आहे.

### सर्विस म्हणून सॉफ्टवेअर (SaaS)

SaaS व्यवसायांना वापरण्यास तयार सॉफ्टवेअर प्रदान करते जे इंटरनेटद्वारे वापरकर्त्यांना वितरित केले जाते. हार्डवेअर, सॉफ्टवेअर, डेटा स्टोरेज, पॅच मॅनेजमेंट आणि हार्डवेअर/सॉफ्टवेअर अपडेट्ससह सर्व बेसिक पायाभूत सुविधा SaaS प्रदात्यांद्वारे मॅनेज केल्या जातात. SaaS हे सबस्क्रिप्शन-

आधारित मॉडेल आहे, ज्यासाठी व्यवसायांना ते वापरू इच्छित असलेल्या सर्विस चे मेम्बरशिप घेणे आवश्यक आहे. वापरकर्ते वेब ब्राउझरद्वारे थेट SaaS ऍप्लिकेशन्समध्ये एक्सेस करू शकतात, जे त्यांना डाउनलोड किंवा इंस्टाल करण्याची आवश्यकता दूर करते. SaaS वापरकर्त्यांना ऍक्टिव्ह इंटरनेट कनेक्शनसह कोटूनही आणि कधीही वेब-आधारित सोल्यूशन्समध्ये एक्सेस करण्यास अनुमती देते. काही लोकप्रिय SaaS उपायांमध्ये Microsoft 365, Google Workspace आणि Salesforce यांचा समावेश आहे.

### क्लाउड कॉम्प्युटिंगचे फायदे काय आहेत?

क्लाउड कंप्युटिंग व्यवसायांना अक्षरशः कुठूनही आणि अधिक कार्यक्षमतेने ऑपरेट करण्यास सक्षम करते. क्लाउड कॉम्प्युटिंगच्या काही फायद्यांमध्ये हे समाविष्ट आहे:

- **खर्च बचत:** क्लाउड कॉम्प्युटिंगचा सर्वात मोठा फायदा म्हणजे कमी खर्च. व्यवसायांना त्यांची स्वतःची IT पायाभूत सुविधा तयार करण्याची किंवा हार्डवेअर किंवा उपकरणे खरेदी करण्याची आवश्यकता नसल्यामुळे, ते कंपन्यांना भांडवली खर्च लक्षणीयरीत्या कमी करण्यास मदत करते.
- **लवचिकता/स्केलेबिलिटी:** क्लाउड कंप्युटिंग सर्व आकारांच्या व्यवसायांना अधिक लवचिकता प्रदान करते. त्यांना अतिरिक्त बँडविड्थ, कॉम्प्युटिंग पॉवर किंवा स्टोरेज स्पेसची आवश्यकता असली तरीही, ते त्यांच्या गरजा आणि बजेटनुसार संगणकीय रिसोर्सेस अखंडपणे वाढवू शकतात किंवा कमी करू शकतात.
- **सेक्युरिटी:** डेटा सुरक्षा ही आज व्यवसायांसाठी एक प्रमुख चिंता आहे. क्लाउडमधील सेन्सिटिव्ह डेटा सुरक्षितपणे हाताळला जातो आणि स्टोर केला जातो याची खात्री करण्यासाठी क्लाउड विक्रेते एडवान्स सुरक्षा वैशिष्ट्ये जसे की प्रमाणीकरण, एक्सेस मॅनेजमेंट, डेटा एन्क्रिप्शन इ. प्रदान करतात.
- **गतिशीलता:** क्लाउड कंप्युटिंग वापरकर्त्यांना इंटरनेट वापरून कोणत्याही डिव्हाइसवरून, कोठूनही आणि कधीही कॉर्पोरेट डेटामध्ये एक्सेस करण्याची परवानगी देते. सोयीस्करपणे सबलब्ध इनफार्मेशन मुळे कर्मचारी प्रवासातही उत्पादक राहू शकतात.
- **वाढलेले कोलायब्रेशन :** क्लाउड ऍप्लिकेशन्स व्यवसायांना अखंडपणे संवाद साधण्याची आणि सुरक्षितपणे इनफार्मेशन ऍक्सेस आणि शेअर करण्याची परवानगी देतात, कोलायब्रेशन करून साधे आणि त्रासओपन . क्लाउड कॉम्प्युटिंग मल्टिपल वापरकर्त्यांना डॉक्युमेंट्स एडिट करण्यास किंवा फायलींवर एकाच वेळी आणि पारदर्शक मेथड्स ने कार्य करण्यास सक्षम करते.
- **डिजास्टर रिकव्हरी:** डेटा गमावणे आणि डाउनटाइममुळे कोणत्याही आकाराच्या व्यवसायांचे कधीही भरून न येणारे नुकसान होऊ शकते. मुख्य क्लाउड विक्रेते उच्च ऍप्लिकेशन उपलब्धता आणि व्यवसाय सातत्य सुनिश्चित करण्यासाठी हार्डवेअर/सॉफ्टवेअर अयशस्वी, नैसर्गिक आपत्ती आणि वीज खंडित यासारख्या अनपेक्षित विघटनकारी घटनांना तोंड देण्यासाठी सुसज्ज आहेत.

- **ऑटोमॅटिक अपडेट :** मॅन्युअल ऑर्गनायझेशनव्यापी सॉफ्टवेअर अपडेट्स करण्यासाठी आयटी कर्मचार्यांचा खूप मौल्यवान वेळ लागू शकतो. तथापि, क्लाउड कंप्युटिंगसह, सर्विस प्रदाते व्यवसायांना अद्ययावत सॉफ्टवेअर आवृत्त्या, नवीनतम सर्व्हर आणि अपग्रेड केलेली प्रक्रिया शक्ती प्रदान करण्यासाठी नवीनतम तंत्रज्ञानासह नियमितपणे रीफ्रेश आणि अपडेट करतात.

### क्लाउड कॉम्प्युटिंगचे तोटे काय आहेत?

क्लाउडमध्ये काम करण्याचे फायदे प्रचंड आहेत. तथापि, क्लाउडमध्ये ट्रांजिशन करण्याचा निर्णय घेण्यापूर्वी कंपन्यांनी जागरूक असले पाहिजे असे काही तोटे आहेत. क्लाउड कॉम्प्युटिंगचे शीर्ष पाच तोटे खाली लिस्ट बद्ध आहेत.

- 1 **डाउनटाइम:** क्लाउड संगणन सिस्टिम इंटरनेटवर पूर्णपणे अवलंबून असल्याने, ऍक्टिव्ह इंटरनेट कनेक्शनशिवाय, व्यवसाय क्लाउडमध्ये होस्ट केलेल्या डेटा किंवा ऍप्लिकेशन मध्ये एक्सेस करू शकत नाहीत. 2020 मध्ये Google ला तीन गंभीर आउटेजचा सामना करावा लागला ज्याचा रिझल्ट जगभरातील त्याच्या बहुतांश सर्विस आणि वापरकर्त्यांवर झाला.
- 2 **विक्रेता लॉक-इन:** कंपनीचे वर्कलोड आणि सर्विस एका क्लाउड प्रदात्याकडून दुसऱ्याकडे स्थलांतरित करणे हे क्लाउड कॉम्प्युटिंगमधील एक मोठे आव्हान आहे. क्लाउड वातावरणातील फरकांमुळे सुसंगतता किंवा एकत्रीकरण समस्या उद्भवू शकतात. जर ट्रांजिशन योग्यरित्या हाताळले गेले नाही, तर ते एखाद्या संस्थेच्या डेटाला अनावश्यक सुरक्षा भेद्यतेसाठी उघड करू शकते.
- 3 **मर्यादित कंट्रोल :** क्लाउड इन्फ्रास्ट्रक्चर पूर्णपणे क्लाउड विक्रेत्याच्या मालकीची आणि मॅनेज असल्याने, क्लाउड कॉम्प्युटिंग सर्विस वापरणाऱ्या व्यवसायांचे त्यांच्या डेटा, ऍप्लिकेशन्स आणि सर्विस वर मर्यादित कंट्रोल असते. त्यामुळे, क्लाउड इन्फ्रास्ट्रक्चरमध्ये व्यवसाय काय करू शकतो आणि काय करू शकत नाही हे समजून घेण्यासाठी योग्य एंड-यूजर परवाना करार (EULA) असणे महत्त्वाचे आहे.
- 4 **सेक्युरिटी:** क्लाउड मध्ये कंपनीचा सेन्सिटिव्ह डेटा संचयित करण्याच्या प्रमुख चिंतेपैकी एक सुरक्षा आहे. जरी क्लाउड सर्विस प्रदाते एडवान्स सुरक्षा उपायांची अंमलबजावणी करत असले तरी, पूर्णपणे तृतीय पक्षाच्या मालकीच्या आणि ऑपरेट केलेल्या रिमोट सर्व्हरवर गोपनीय फाइल्स संचयित केल्याने नेहमीच सुरक्षा धोके उघडतात. जेव्हा एखादी संस्था क्लाउड कॉम्प्युटिंग मॉडेल स्वीकारते, तेव्हा IT सुरक्षा जबाबदारी क्लाउड विक्रेता आणि युजर यांच्यात सामायिक केली जाते. यामुळे, प्रत्येक पक्ष त्यांच्या नियंत्रणात असलेल्या प्रॉपर्टी, प्रक्रिया आणि कार्यासाठी जबाबदार आहे.
- 5 **डेटा हरवणे किंवा चोरी:** व्हर्च्युअल डेटा सेंटर्समध्ये महत्त्वपूर्ण डेटा संचयित केल्याने डेटाचे नुकसान होऊ शकते अशा विविध जोखमींचे दरवाजे उघडू शकतात, जसे की क्लाउड चुकीचे कॉन्फिगरेशन, इनफार्मेशन ची चोरी, सुरक्षा भंग, चोरी झालेली क्रेडेन्शियल्स इ. शिवाय, क्लाउड सर्विस प्रदाते, जसे की Microsoft आणि Google, सामायिक

जबाबदारीचे मॉडेल फॉलो करतात, जेथे विक्रेत्याने ॲप्लिकेशनची उपलब्धता आणि त्यात समाविष्ट असलेल्या सर्व गोष्टींची जबाबदारी स्वीकारली आहे, तर ग्राहक ॲप्लिकेशन डेटाची जबाबदारी राखून ठेवतो. , एडमिनिस्ट्रेशन आणि युजर मॅनेजमेन्ट .

### स्पॅनिंग बॅकअपसह SaaS डेटा प्रोटेक्शन सुधारा

स्टॅटिस्टाच्या मते, 2021 पर्यंत, सर्व कॉर्पोरेट डेटापैकी सुमारे 50% क्लाउडमध्ये स्टोर आहे. डेटा सूचित करतो की व्यवसाय ग्लोबल लेयर वर त्यांच्या क्लाउड सर्विस प्रदात्यांना त्यांच्या सेन्सिटिव्ह डेटासह विश्वास ठेवतात. कंपनी कोणता क्लाउड विक्रेता निवडत असला तरीही, क्लाउडचे फायदे बेसिक सुरक्षा जोखमीपेक्षा जास्त होणार नाहीत याची काळजी घेतली पाहिजे.

Microsoft 365, Google Workspace आणि Salesforce सारखे SaaS सोल्यूशन्स वापरणारे व्यवसाय दररोज डेटा गमावतात. अनेक कंपन्यांचा असा विश्वास आहे की सास विक्रेते त्यांच्या डेटाचे प्रोटेक्शन करण्यासाठी जबाबदार आहेत. तथापि, तसे नाही. SaaS प्रदाते ॲप्लिकेशन अपटाइम आणि उपलब्धता सुनिश्चित करतात, डेटा प्रोटेक्शन ही ग्राहकांची जबाबदारी आहे.

अशा प्रकारे, व्यवसायांना विश्वासाह SaaS बॅकअप सोल्यूशनची आवश्यकता असते जे फिशिंग, रॅन्समवेअर आणि मालवेअर हल्ला, मानवी त्रुटी, दुर्भावनापूर्ण वर्तन आणि कॉन्फिगरेशन आणि सिंक त्रुटी यासारख्या डेटा गमावण्याच्या सर्वात सामान्य कारणांपासून त्यांच्या मौल्यवान डेटाचे प्रोटेक्शन करू शकतात.

मायक्रोसॉफ्ट 365, Google Workspace आणि Salesforce साठी बॅकअप आणि एंड-टू-एंड प्रोटेक्शन उपायांचा एकसंपान्शन करत, या धोक्यांमुळे गंभीर डेटाचे नुकसान होण्यापासून प्रोटेक्शन करण्यासाठी, तडजोडीचा धोका कमी करण्यासाठी आणि एन्ड यूजर आणि प्रशासकांना त्वरीत सर्च प्यात आणि सक्षम करण्यासाठी मूळ कार्यक्षमतेतील अंतर भरून काढतात. काही क्लिकमध्ये डेटा त्याच्या मूळ स्थितीत रिस्टोर करा.

### क्लाउड कॉम्प्युटिंग शिकण्यासाठी सर्वोत्तम रिसोर्सस

आज अनेक व्यावसायिक क्लाउड कॉम्प्युटिंगला त्यांचे करिअर म्हणून अवलंबत आहेत. या कारकिर्दीत यशस्वी होण्यासाठी, त्यांना त्यांची कौशल्ये सुधारण्यासाठी आणि क्लाउड कॉम्प्युटिंग शिकण्यासाठी नवीनतम बातम्या आणि ट्रेड्सह अद्ययावत राहणे आवश्यक आहे. क्लाउड कॉम्प्युटिंग शिकण्यासाठी आणि उद्योगातील नवीनतम ट्रेडच्या संपर्कात राहण्यासाठी खालील काही शीर्ष रिसोर्सस आहेत:

### ऑनलाइन क्लाउड कॉम्प्युटिंग कोर्सेस

क्लाउड कॉम्प्युटिंग शिकण्याचा ऑनलाइन कोर्स हा एक उत्तम पाथ आहे. अनेक ऑनलाइन अभ्यासक्रम शिकणाऱ्यांना मदत करण्यासाठी व्हिडिओ, लेख, ईपुस्तके, रिपोर्ट , हँड-ऑन लॅब आणि असेसमेंट ऑफर करतात. ऑनलाइन काही सर्वोत्तम क्लाउड संगणन अभ्यासक्रम खाली लिस्ट बद्ध आहेत:

- Udemy वर नवशिक्यांसाठी Amazon AWS वर क्लाउड कॉम्प्युटिंगचा परिचय
- Coursera वर क्लाउड कॉम्प्युटिंगचा परिचय
- इंटरनॅशनल वर AWS प्रशिक्षणासह क्लाउड संगणन
- क्लाउड कॉम्प्युटिंग बेसिक : Udemy वर क्लाउड अभियंता म्हणून तुमची कारकीर्द वाढवा • edX वर Microsoft द्वारे क्लाउड कंप्युटिंगचा परिचय
- NPTEL द्वारे क्लाउड संगणन
- कोर्सेरा वर क्लाउड कॉम्प्युटिंग बेसिक्स (क्लाउड 101).
- एकूण: Cloud Computing / CompTIA Cloud+ प्रमाणपत्र. (CV0-002) Udemy वर

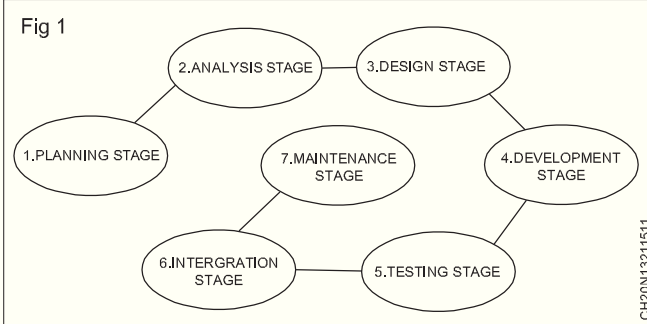


## एप्लिकेशन डेव्हलपमेंट लाइफ साइकल टप्पे ओळखा (Identify Phases of the Application Development Life Cycle)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एप्लिकेशन डेव्हलपमेंट लाइफ साइकल टप्पे ओळखा
- SDLC चे फायदे.

एप्लिकेशन डेव्हलपमेंट लाइफ साइकल ओळखा सायकल (पिक्चर 1)



### सिस्टिम

सॉफ्टवेअर डेव्हलपमेंट प्रक्रिया सामान्यतः लांब आणि त्रासदायक असते. परंतु प्रकल्प व्यवस्थापक आणि सिस्टिम विश्लेषक अधिक नियमितता, कार्यक्षमता आणि एकूण गुणवत्तेसह इनफार्मेशन सिस्टिम किंवा सॉफ्टवेअर उत्पादनांची रूपरेषा, डिझाइन, विकास, टेस्टिंग आणि शेवटी तैनात करण्यासाठी सॉफ्टवेअर विकास जीवन चक्राचा लाभ घेऊ शकतात.

### सिस्टिम डेव्हलपमेंट लाइफ सायकल म्हणजे काय?

सिस्टिम डेव्हलपमेंट लाइफ सायकल किंवा SDLC हे मूलतः एक प्रोजेक्ट मॅनेजमेंट मॉडेल आहे. एखाद्या प्रकल्पाच्या सुरुवातीच्या कल्पना किंवा संकल्पनेपासून ते तैनाती आणि नंतरच्या देखभालीपर्यंत विविध टप्प्यांची व्याख्या केली जाते.

### सिस्टिम डेव्हलपमेंट लाइफ साइकल यूएस पाथ दर्शक (पिक्चर 2)

या पाथ दर्शकामध्ये, आम्ही तुम्हाला सिस्टिम डेव्हलपमेंट लाइफ सायकल, त्यातील सर्व टप्प्यांबद्दल जाणून घेण्यासाठी आवश्यक असलेल्या सर्व गोष्टींचा समावेश करू. आम्ही सिस्टिम अनलिसिस्ट च्या भूमिका आणि SDLC स्वीकारून तुमच्या प्रकल्पाला मिळू शकणारे फायदे देखील पाहू.

सिस्टिम विकास जीवन चक्राचे 7 टप्पे



आधुनिक सिस्टिम विकासाच्या जीवनचक्राचे सात प्राथमरी टप्पे आहेत. येथे एक संक्षिप्त ब्रेकडाउन आहे:

- नियोजन स्टेज
- विश्लेषण स्टेजची व्यवहार्यता किंवा आवश्यकता
- डिझाइन आणि प्रोटोटाइपिंग स्टेज
- सॉफ्टवेअर डेव्हलपमेंट स्टेज
- सॉफ्टवेअर टेस्टिंग स्टेज
- अंमलबजावणी आणि एकत्रीकरण
- ऑपरेशन्स आणि मॅटेनन्स स्टेज

आता प्रत्येक टप्प्यावर वैयक्तिकरित्या जवळून पाहू.

### नियोजन स्टेज

आम्ही प्लॅनिंग स्टेजपासून सुरुवात करण्यापूर्वी, आम्ही तुम्हाला देऊ शकणारी सर्वोत्कृष्ट टीप म्हणजे वेळ काढणे आणि अॅप डेव्हलपमेंट लाइफ सायकलची योग्य इनफार्मेशन घेणे.

नियोजनाचा टप्पा (ज्याला व्यवहार्यता टप्पा देखील म्हटले जाते) अगदी असे दिसते: विकासक ज्या टप्प्यात आगामी प्रकल्पाची योजना आखतील.

हे कोणत्याही विद्यमान सिस्टिम ची समस्या आणि व्याप्ती परिभाषित करण्यात मदत करते, तसेच त्यांच्या नवीन प्रणालींसाठी उद्दिष्टे निश्चित करण्यात मदत करते.

आगामी विकास चक्रासाठी एक प्रभावी रूपरेषा विकसित करून, विकासावर रिझल्ट होण्याआधी ते सैद्धांतिकदृष्ट्या समस्या पकडतील.

आणि त्यांची योजना पूर्ण करण्यासाठी आवश्यक निधी आणि रिसोर्सेस सुरक्षित करण्यात मदत करा.

कदाचित सर्वात महत्त्वाचे म्हणजे, नियोजनाचा टप्पा प्रकल्पाचे वेळापत्रक सेट करतो, जे एखाद्या व्यावसायिक उत्पादनासाठी असेल जे स्पेसिफिक वेळेपर्यंत बाजारात पाठवले जाणे आवश्यक असेल तर ते महत्त्वाचे ठरू शकते.

### विश्लेषण स्टेज

विश्लेषण स्टेजमध्ये नवीन सिस्टिम साठी आवश्यक असलेले सर्व स्पेसिफिक तपशील एकत्र करणे तसेच प्रोटोटाइपसाठी प्रथम कल्पना निश्चित करणे समाविष्ट आहे.

विकासक हे करू शकतात:

- कोणत्याही प्रोटोटाइप सिस्टम आवश्यकता परिभाषित करा
- विद्यमान प्रोटोटाइपच्या पर्यायांचे असेसमेंट करा
- अंतिम वापरकर्त्यांच्या गरजा निश्चित करण्यासाठी रिसर्च न आणि विश्लेषण करा

शिवाय, विकासक अनेकदा सॉफ्टवेअर आवश्यकता तपशील किंवा SRS डॉक्युमेंट्स तयार करतात.

यामध्ये सॉफ्टवेअर, हार्डवेअर आणि नेटवर्कची आवश्यकता असलेल्या सिस्टीमसाठी ते तयार करण्याची योजना असलेल्या सर्व वैशिष्ट्यांचा समावेश आहे. हे त्यांना इतर विकास संघांप्रमाणे त्याच ठिकाणी काम करताना निधी किंवा रिसोर्सस ओव्हरड्रॉ करण्यापासून प्रतिबंधित करेल.

### डिझाइन स्टेज

डिझाइन स्टेज हा मुख्य डेव्हलपर स्टेजसाठी आवश्यक अग्रदूत आहे.

विकासक प्रथम संपूर्ण अनुप्रयोगासाठी तपशीलांची रूपरेषा तयार करतील, स्पेसिफिक पैलूसह, जसे की:

- युजर इंटरफेस
- सिस्टम इंटरफेस
- नेटवर्क आणि नेटवर्क आवश्यकता
- डेटाबेस

ते सामान्यतः त्यांनी तयार केलेले SRS डॉक्युमेंट्स अधिक लॉजिकल संरचनेत बदलतील जे नंतर प्रोग्रामिंग भाषेत लागू केले जाऊ शकते. ऑपरेशन, प्रशिक्षण आणि देखभाल योजना या सर्व तयार केल्या जातील जेणेकरून विकासकांना सायकलच्या प्रत्येक टप्प्यावर पुढे जाण्यासाठी काय करण्याची आवश्यकता आहे हे समजेल.

एकदा पूर्ण झाल्यावर, विकास व्यवस्थापक SDLC च्या पुढील टप्प्यांमध्ये संदर्भित करण्यासाठी डिझाइन डॉक्युमेंट्स तयार करतील.

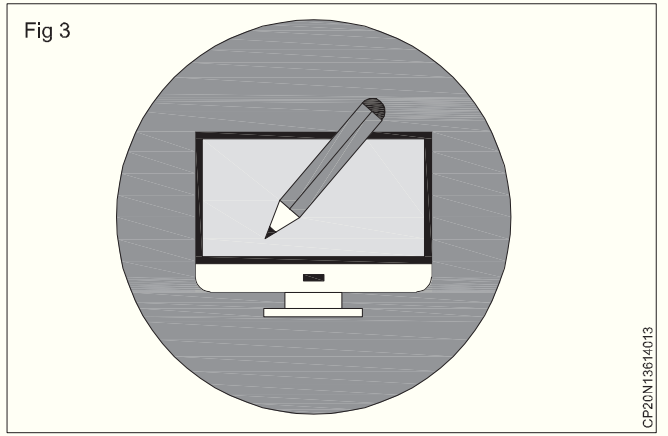
### विकासाचा टप्पा (पिव्चर 3)

डेव्हलपमेंट स्टेज हा एक पार्ट आहे जिथे डेव्हलपर प्रत्यक्षात कोड लिहितात आणि पूर्वीच्या डिझाइन डॉक्युमेंट्स आणि एक्सटर्नल रेखा दिलेल्या वैशिष्ट्यांनुसार एप्लिकेशन तयार करतात.

इथेच स्टॅटिक ऍप्लिकेशन सिक्युरिटी टेस्टिंग किंवा SAST टूल्सचा यूजेस होतो.

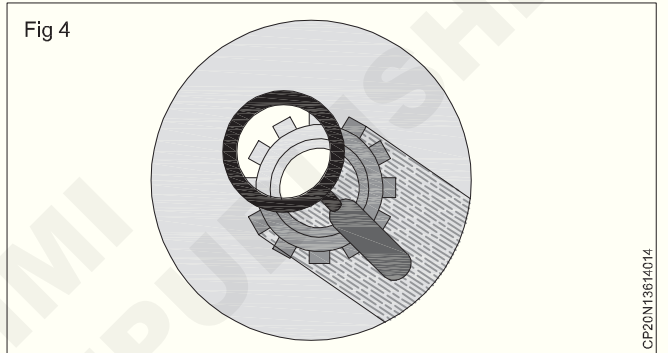
उत्पादन प्रोग्राम कोड डिझाइन डॉक्युमेंट्स च्या वैशिष्ट्यांनुसार तयार केला जातो. सैद्धांतिकदृष्ट्या, सर्व पूर्वनियोजन आणि आराखड्याने वास्तविक विकासाचा टप्पा तुलनेने सरळ केला पाहिजे.

डेव्हलपर संस्थेने परिभाषित केल्यानुसार कोणत्याही कोडिंग पाथ दर्शक तत्वांचे पालन करतील आणि कंपाइलर, डीबगर आणि इंटरप्रिटर यासारख्या विविध साधनांचा वापर करतील.



प्रोग्रामिंग भाषांमध्ये C++, PHP आणि बरेच काही यासारख्या स्टेपल्सचा समावेश असू शकतो. विकासक प्रकल्प तपशील आणि आवश्यकतांवर आधारित वापरण्यासाठी योग्य प्रोग्रामिंग कोड निवडतील.

### टेस्टिंग स्टेज (पिव्चर 4)



बिल्टिंग सॉफ्टवेअर शेवट नाही.

आता कोणतेही दोष नाहीत आणि अंतिम-यूजर च्या अनुभवावर कोणत्याही वेळी नकारात्मक रिझल्ट होणार नाही याची खात्री करण्यासाठी त्याची टेस्टिंग करणे आवश्यक आहे.

टेस्टिंग स्टेज दरम्यान, विकासक त्यांच्या सॉफ्टवेअरवर बारीक-दात असलेल्या कंगव्याने जातील, कोणत्याही दोष किंवा दोष लक्षात घेतील ज्यांचा मागोवा घेणे, निराकरण करणे आणि नंतर पुन्हा टेस्टिंग करणे आवश्यक आहे. हे महत्त्वाचे आहे की सॉफ्टवेअर संपूर्णपणे SRS डॉक्युमेंट्स त पूर्वी परिभाषित केलेल्या गुणवत्ता मानकांची पूर्तता करते.

डेव्हलपरचे कौशल्य, सॉफ्टवेअरची जटिलता आणि अंतिम यूजर च्या आवश्यकतांवर अवलंबून, टेस्टिंग एकतर अत्यंत लहान टप्पा असू शकते किंवा खूप वेळ लागू शकतो. अधिक इनफार्मेशन साठी सॉफ्टवेअर टेस्टिंग प्रकल्पांसाठी आमच्या शीर्ष 10 सर्वोत्तम मेथड्स पहा.

### अंमलबजावणी आणि एकत्रीकरण स्टेज

टेस्टिंग केल्यानंतर, सॉफ्टवेअरची संपूर्ण रचना एकत्र येईल. पुढील त्रुटी किंवा दोष सर्चण्यासाठी सामान्यतः प्रशिक्षण वातावरणाचा फायदा घेऊन विकासक प्रयत्नांद्वारे प्रायमरी सोर्स कोडमध्ये भिन्न मॉड्यूल किंवा डिझाइन एकत्रित केले जातील.

इनफार्मेशन सिस्टिम त्याच्या वातावरणात समाकलित केली जाईल आणि अखेरीस इंस्टाल केली जाईल. हा टप्पा पार केल्यानंतर, सॉफ्टवेअर

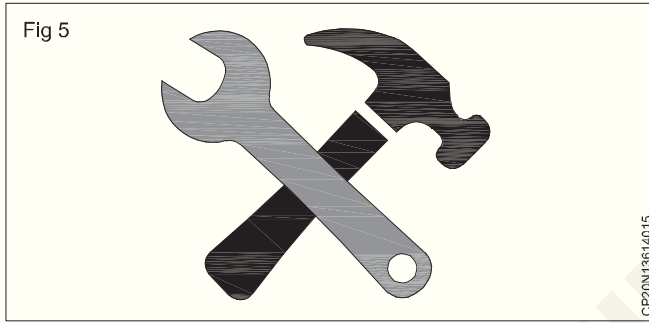
सैद्धांतिकदृष्ट्या बाजारपेठेसाठी तयार आहे आणि कोणत्याही अंतिम वापरकर्त्यांना प्रदान केले जाऊ शकते.

### मेंटेनेंस स्टेज (पिक्चर 5)

जेव्हा सॉफ्टवेअर बाजारात पोहोचते तेव्हा SDLC संपत नाही. विकसकांनी आता देखभाल मोडमध्ये जाणे आवश्यक आहे आणि अंतिम वापरकर्त्यांद्वारे नोंदवलेल्या समस्या हाताळण्यासाठी आवश्यक असलेल्या कोणत्याही क्रियाकलापांचा सराव सुरू करणे आवश्यक आहे.

शिवाय, सबयोजनांतर सॉफ्टवेअरला आवश्यक असलेले कोणतेही बदल लागू करण्यासाठी विकासक जबाबदार आहेत.

यामध्ये अवशिष्ट बग हाताळणे समाविष्ट असू शकते जे लाँच करण्यापूर्वी पॅच केले जाऊ शकत नव्हते किंवा यूजर च्या अहवालांमुळे उद्भवलेल्या नवीन समस्यांचे निराकरण करू शकतात. लहान सिस्टीमच्या तुलनेत मोठ्या सिस्टीमला दीर्घ देखभाल टप्पे आवश्यक असू शकतात.



### सिस्टम अनलिसिस्ट ची भूमिका

SDLC चा सिस्टीम विश्लेषक हा काही प्रकारे संपूर्ण सिस्टीमचा पर्यवेक्षक असतो. त्यांना सिस्टीम आणि त्याच्या सर्व हलत्या भागांबद्दल पूर्णपणे इनफार्मेशन असली पाहिजे आणि योग्य दिशानिर्देश देऊन प्रकल्पाला पाठ दर्शन करण्यास मदत करू शकतात.

### सिस्टम अनलिसिस्ट असावा:

- प्रकल्पासाठी आवश्यक असलेल्या कोणत्याही तांत्रिक कौशल्यातील तज्ञ
- त्याच्या किंवा तिच्या टीमला यश मिळवून देण्यासाठी मदत करण्यासाठी एक चांगला संवादक
- एक चांगला नियोजक जेणेकरून विकास फंक्शन विकास चक्राच्या प्रत्येक टप्प्यावर वेळेवर पार पाडता येतील

अशाप्रकारे, सिस्टम अनलिसिस्ट मध्ये परस्पर, तांत्रिक, मॅनेजमेन्ट आणि विश्लेषणात्मक कौशल्ये यांचे मिश्रण असले पाहिजे. ते अष्टपैलू व्यावसायिक आहेत जे SDLC बनवू किंवा खंडित करू शकतात.

दिलेल्या प्रकल्पाच्या अंतिम यशासाठी त्यांच्या जबाबदाऱ्या खूप वैविध्यपूर्ण आणि महत्त्वपूर्ण आहेत. सिस्टम अनलिसिस्ट कडून अनेकदा अपेक्षा केली जाईल:

- तथ्ये आणि इनफार्मेशन गोळा करा

- कोणत्या बग्सना प्राधान्य द्यायचे किंवा कोणती वैशिष्ट्ये कापायची याविषयी निर्णय घ्या • पर्यायी उपाय सुचवा
- वापरकर्ते आणि प्रोग्रामर दोघांनाही सहज समजू शकतील अशी वैशिष्ट्ये काढा
- नंतरच्या एकीकरणासाठी मॉड्यूलरिटी ठेवताना लॉजिकल सिस्टिम लागू करा
- प्रकल्पाच्या उद्दिष्टानुसार आवश्यक असलेल्या परिणामी सिस्टिम चे असेसमेंट आणि सुधारणा करण्यास सक्षम व्हा
- यूजर च्या गरजा परिभाषित करून आणि समजून घेऊन प्रकल्पाच्या गरजा आणि उद्दिष्टांचे नियोजन करण्यात मदत करा

### 6 बेसिक SDLC मेथड्स

जरी सिस्टीम डेव्हलपमेंट लाइफ सायकल हे व्यापक अर्थाने प्रोजेक्ट मॅनेजमेंट मॉडेल असले तरी, स्पेसिफिक रिझल्ट साध्य करण्यासाठी किंवा विविध गुणधर्मांसह अधिक SDLC प्रदान करण्यासाठी आणखी सहा स्पेसिफिक पद्धतींचा वापर केला जाऊ शकतो.

### वॉटरफॉल मॉडेल (पिक्चर 6)

वॉटरफॉल मॉडेल हे सर्व SDLC पद्धतींपैकी सर्वात जुने आहे. हे रेखीय आणि सरळ आहे आणि विकास कार्यसंघांनी पुढील टप्प्यावर जाण्यापूर्वी प्रकल्पाचा एक टप्पा पूर्णपणे पूर्ण करणे आवश्यक आहे.

प्रत्येक टप्प्याचा वेगळा प्रकल्प आराखडा असतो आणि समान समस्या टाळण्यासाठी (जर आढळल्यास) मागील टप्प्यातील इनफार्मेशन घेते. तथापि, हे लवकर विलंब होण्यास असुरक्षित आहे आणि नंतर रस्त्याच्या खाली विकास कार्यसंघांसाठी मोठ्या समस्या उद्भवू शकतात.

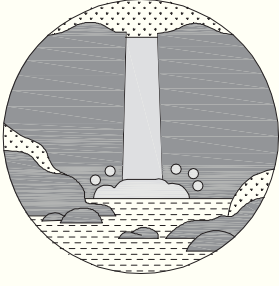


### पुनरावृत्ती (Iterative) मॉडेल (पिक्चर 7)

पुनरावृत्ती मॉडेल पुनरावृत्ती आणि पुनरावृत्ती टेस्टिंग वर लक्ष केंद्रित करते. सॉफ्टवेअर प्रकल्पाच्या नवीन आवृत्त्या प्रत्येक टप्प्याच्या शेवटी तयार केल्या जातात ज्यामुळे संभाव्य त्रुटी पकडल्या जातात आणि विकसकांना अंतिम उत्पादन बाजारात येईपर्यंत ते सतत सुधारण्यास अनुमती देते.

या मॉडेलचा एक फायदा असा आहे की विकासक त्यांच्या विकासाच्या जीवन चक्रात तुलनेने लवकर प्रकल्पाची कार्यरत आवृत्ती तयार करू शकतात, त्यामुळे बदल अंमलात आणणे अनेकदा कमी खर्चिक असते.

Fig 7



### स्पाइरल मॉडेल

स्पाइरल मॉडेल इतर पद्धतींच्या तुलनेत लवचिक असतात. प्रकल्प चार मुख्य टप्प्यांतून पुनःपुन्हा रूपकात्मक स्पाइरल गतीने जातात.

मोठ्या प्रकल्पांसाठी हे फायदेशीर आहे कारण विकास कार्यसंघ अतिशय कस्टमाइज उत्पादने तयार करू शकतात आणि जीवन चक्रात तुलनेने लवकर प्राप्त झालेला कोणताही अभिप्राय समाविष्ट करू शकतात.

### व्ही मॉडेल

व्ही-मॉडेल (जो पडताळणी आणि प्रमाणीकरणासाठी लहान आहे) हे धबधबा मॉडेलसारखेच आहे. संभाव्य दोष आणि दोष सर्चण्यासाठी प्रत्येक विकासाच्या टप्प्यात एक टेस्टिंग टप्पा समाविष्ट केला जातो.

हे आश्चर्यकारकपणे शिस्तबद्ध आहे आणि एक कठोर टाइमलाइन आवश्यक आहे. परंतु सैद्धांतिकदृष्ट्या, ते मोठ्या बग्सना नियंत्रणाबाहेर जाण्यापासून रोखून मुख्य धबधबा मॉडेलच्या उणीवांवर प्रकाश टाकते.

### बिग बॅग मॉडेल

बिग बॅग मॉडेल आश्चर्यकारकपणे लवचिक आहे आणि कठोर प्रक्रिया किंवा प्रक्रियेचे अनुसरण करत नाही. हे अगदी तपशीलवार नियोजन मागे सोडते. जेव्हा ग्राहक किंवा क्लायंटला त्यांना काय हवे आहे याची खात्री नसते तेव्हा ते बहुतेक व्यापक कल्पना विकसित करण्यासाठी वापरले जाते. विकासक फक्त पैसा आणि रिसोर्सस घेऊन प्रकल्प सुरू करतात.

त्यांचे आऊटपुट क्लायंटला शेवटी जे हवे आहे त्याच्यापेक्षा जवळ किंवा दूर असू शकते. हे मुख्यतः लहान प्रकल्पांसाठी वापरले जाते आणि त्याच कंपनीमधील इतर प्रकल्पांची इनफार्मेशन देण्यासाठी डिझाइन केलेले प्रायोगिक जीवन चक्र. (पिक्चर 8)

Fig 8



### चपळ(Agile) मॉडेल

चपळ मॉडेल तुलनेने सुप्रसिद्ध आहे, स्पेशल तः सॉफ्टवेअर विकास उद्योगात.

चपळ कार्यमेथड्स जलद आणि चालू असलेल्या प्रकाशन चक्रांना प्राधान्य देते, रीलिझ दरम्यान लहान परंतु वाढीव बदलांचा वापर करते. यामुळे इतर मॉडेलच्या तुलनेत अधिक पुनरावृत्ती आणि अनेक चाचण्या होतात.

सैद्धांतिकदृष्ट्या, हे मॉडेल संघांना प्रकल्पाच्या अधिक जटिल टप्प्यापर्यंत ते चुकवण्याऐवजी उद्भवलेल्या लहान समस्यांचे निराकरण करण्यात मदत करते. (पिक्चर 9)

Fig 9



**SDLC चे फायदे:** SDLC विकास कार्यसंघांना अनेक फायदे प्रदान करते जे ते योग्यरित्या अंमलात आणतात.

**टार्गेट डिस्क्रिप्शन क्लियर करा:** वेळ आणि रिसोर्सस वाया जाण्याची रिस्क कमी करून, विकासकांना त्यांना पूर्ण करण्यासाठी आवश्यक असलेली उद्दिष्टे आणि डिलिव्हेरेबल्स निश्चितपणे माहित असतात.

**स्थापनेपूर्वी योग्य टेस्टिंग :** SDLC मॉडेल अधिक सोर्स कोडमध्ये इंस्टाल होण्यापूर्वी सर्व सॉफ्टवेअरची टेस्टिंग केली गेली आहे याची खात्री करण्यासाठी चेक आणि बॅलन्स लागू करतात.

**स्पष्ट स्टेज प्रगती:** जोपर्यंत आधीचे पूर्ण होत नाही आणि व्यवस्थापकाने साइन ऑफ केले नाही तोपर्यंत विकासक पुढच्या वयात जाऊ शकत नाहीत. सदस्य लवचिकता: SDLC कडे प्रकल्पाची उद्दिष्टे आणि कार्यपद्धतीसाठी सु-संरचित डॉक्युमेंट्स असल्याने, टीम सदस्य सोडू शकतात आणि तुलनेने वेदनारहित नवीन सदस्य बदलू शकतात.

**परिपूर्णता प्राप्त करण्यायोग्य आहे:** सर्व SDLC टप्पे एकमेकांना परत देण्यासाठी आहेत. त्यामुळे SDLC मॉडेल मूलतः परिपूर्ण होईपर्यंत पुनरावृत्ती आणि स्वतःमध्ये सुधारणा करण्यासाठी प्रकल्पांना मदत करू शकतात.

### कोणताही सदस्य प्रकल्प बनवत नाही किंवा तोडत नाही

पुन्हा, SDLCs विस्तृत कागदपत्रे आणि पाथ दर्शक डॉक्युमेंट्स चा वापर करत असल्याने, हा एक सांघिक प्रयत्न आहे आणि एकही मोठा सदस्य गमावल्याने प्रकल्पाची वेळ धोक्यात येणार नाही.

सिस्टम डेव्हलपमेंट लाइफ सायकलबद्दल आपल्याला काय माहित असणे आवश्यक आहे

## SDLC कुठे वापरले जाते?

आयटी प्रकल्प विकसित करताना सिस्टिम विकास जीवन चक्र सामान्यतः वापरले जातात.

सॉफ्टवेअर डेव्हलपमेंट मॅनेजर SDLCs चा वापर विकासाच्या विविध टप्प्यांची रूपरेषा करण्यासाठी करतील, प्रत्येकाने टप्पे वेळेवर आणि योग्य क्रमाने पूर्ण केले आहेत याची खात्री करा आणि प्रकल्प शक्य तितक्या लवकर आणि दोषओपन वितरित केला जाईल.

SDLCs सिस्टिम अनलिसिस्ट द्वारे देखील अधिक स्पेसिफिक पणे वापरले जाऊ शकतात कारण ते नवीन इनफार्मेशन सिस्टिम विकसित करतात आणि नंतर लागू करतात.

## कोणते SDLC मॉडेल सर्वोत्तम आहे?

तुमच्या कार्यसंघाची उद्दिष्टे आणि संसाधनांची आवश्यकता काय आहे यावर हे मुख्यत्वे अवलंबून असते.

बहुतेक IT विकास संघ त्यांच्या SDLC साठी चपळ मेथड्स चा वापर करतात. तथापि, इतर पुनरावृत्ती किंवा स्पाइरल पद्धतींना प्राधान्य देऊ शकतात.

या तिन्ही मेथड्स लोकप्रिय आहेत कारण ते उत्पादन अधिक सोर्स कोडसह एकत्रित करण्यापूर्वी किंवा बाजारात वितरित करण्यापूर्वी विस्तृत पुनरावृत्ती आणि बग टेस्टिंग साठी परवानगी देतात.

DevOps मेथड्स देखील लोकप्रिय पर्याय आहेत. आणि जर तुम्हाला कधी DevOps काय आहे यावर रीफ्रेशर कोर्सची आवश्यकता असेल, तर तुम्हाला काळजी करण्याची गरज नाही कारण क्लाउडडिफेन्सच्या आमच्या टीमने तुम्हाला कव्हर केले आहे!

## SDLC काय विकसित करते?

SDLC चा वापर सॉफ्टवेअर, सिस्टिम आणि अगदी इनफार्मेशन सिस्टिम विकसित करण्यासाठी किंवा अभियंता करण्यासाठी केला जाऊ शकतो. हे एकाच वेळी हार्डवेअर किंवा सॉफ्टवेअर आणि हार्डवेअर या दोन्हीचे संयोजन विकसित करण्यासाठी देखील वापरले जाऊ शकते.

## वारंवार विचारले जाणारे प्रश्न (FAQ)

सिस्टिम डेव्हलपमेंट लाइफ सायकलचे 5 मूळ टप्पे काय होते?

सिस्टिम डेव्हलपमेंट लाइफ सायकलमध्ये मूलतः सात एवजी पाच टप्पे होते. यामध्ये नियोजन, निर्मिती, विकास, टेस्टिंग आणि तैनाती यांचा समावेश होता. लक्षात घ्या की याने विश्लेषण आणि देखभालीचे प्रमुख टप्पे सोडले आहेत.

## SDLC चे 7 टप्पे काय आहेत?

SDLC च्या नवीन सात टप्प्यांमध्ये नियोजन, विश्लेषण, डिझाइन, विकास, टेस्टिंग, अंमलबजावणी आणि देखभाल यांचा समावेश आहे.

## MIS मध्ये सिस्टिम डेव्हलपमेंट लाइफ सायकल म्हणजे काय?

मॅनेजमेन्ट इनफार्मेशन सिस्टिम किंवा MIS च्या मोठ्या संदर्भात, SDLC व्यवस्थापकांना टारगेट उद्दिष्टे पूर्ण करण्यासाठी इनफार्मेशन सिस्टिम डिझाइन, विकसित, टेस्टिंग आणि तैनात करण्यात मदत करते.

## निष्कर्ष

शेवटी, आयटी आणि इतर दोन्ही उद्योगांमधील कोणत्याही विकास संडन्सर्ट त्यांच्या प्रकल्पांमध्ये सिस्टिम विकास जीवन चक्र लागू करून फायदा होऊ शकतो. सर्वोत्तम परिणामांसाठी तुम्हाला तुमच्या SDLC सोबत कोणती मेथड्स वापरायची आहे हे ओळखण्यासाठी वरील पाथ दर्शक वापरा.

## प्रोग्रामिंग भाषा (पायथन) कमांड लाइन वरून पायथन वापरा (Programming language (Python) Use Python from command line)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- पायथनचा परिचय
- पायथनची वैशिष्ट्ये
- पायथनच्या कमेंट
- व्हेरिएबल.

पायथन नेदरलँड्समधील नॅशनल रिसर्च इन्स्टिट्यूट फॉर मॅथमॅटिक्स अँड कॉम्प्युटर सायन्स येथे 1980 च्या दशकाच्या उत्तरार्धात गुइडो व्हॅन रोसम यांनी विकसित केलेली एक सामान्य उद्देश, उच्च-स्तरीय, व्याख्या केलेली प्रोग्रामिंग भाषा आहे.

पायथन ही सर्वात लोकप्रिय आणि मोठ्या प्रमाणावर वापरली जाणारी प्रोग्रामिंग भाषा आहे जी कन्सोल आधारित, GUI आधारित, वेब प्रोग्रामिंग आणि डेटा विश्लेषणासह कार्याच्या सेटसाठी वापरली जाते.

पायथन ही शिकण्यास सोपी आणि सोपी प्रोग्रामिंग भाषा आहे त्यामुळे तुम्ही प्रोग्रामिंगमध्ये नवीन असलात तरीही, तुम्ही कोणत्याही अडचणींना सामोरे न जाता पायथन शिकू शकता.

वस्तुस्थिती :मॉन्टी पायथन फ्लाइंग सर्कस या कॉमेडी टेलिव्हिजन शोच्या नावावरून पायथन हे नाव देण्यात आले आहे.

- पायथन एप्लिकेशन
- पायथनची इन्स्टॉलेशन

### पायथनची वैशिष्ट्ये

पायथन खाली लिस्ट बद्ध केलेली बरीच वैशिष्ट्ये प्रदान करते.

- शिकण्यास आणि वापरण्यास सोपे

इतर प्रोग्रामिंग भाषांच्या तुलनेत पायथन शिकणे आणि वापरणे सोपे आहे. ही डेव्हलपर -अनुकूल आणि उच्च स्तरीय प्रोग्रामिंग भाषा आहे.

- व्याख्या केलेली भाषा

Python ही व्याख्या केलेली भाषा आहे कारण संकलनाची गरज नाही. हे डीबगिंग सोपे करते आणि त्यामुळे नवशिक्यांसाठी योग्य आहे.

- क्रॉस-प्लॅटफॉर्म भाषा

पायथन विंडोज, लिनक्स, युनिक्स आणि मॅकिंटॉश इत्यादी वेगवेगळ्या प्लॅटफॉर्मवर समान रीतीने चालू शकते. त्यामुळे, पायथन ही पोर्टेबल भाषा आहे असे आपण म्हणू शकतो.

- मोफत आणि ओपन सोर्स

पायथन इंटरप्रीटर हे ओपन -सोर्स परवान्याइंटरनल विकसित केले गेले आहे, ज्यामुळे ते इंस्टाल करणे, वापरणे आणि वितरित करणे विनाव्हॅल्यु आहे.

- ऑब्जेक्ट ओरिएंटेड भाषा

पायथन ऑब्जेक्ट ओरिएंटेड भाषा आणि संकल्पनांना सपोर्ट देते क्लास आणि वस्तू अस्तित्वात येतात.

- GUI प्रोग्रामिंग सपोर्ट

पायथन वापरून ग्राफिकल यूजर इंटरफेस विकसित केले जाऊ शकतात.

- इंटरप्रेटेड

हे C, C++, आणि JAVA इत्यादी भाषांसह सहजतेने एकत्रित केले जाऊ शकते.

### Python मधील कमेंट

सर्वसाधारणपणे, कमेंट प्रोग्रामिंग भाषेत प्रोग्रामचे डिस्क्रिप्शन करण्यासाठी किंवा दुभाष्यापासून कोडचा काही पार्ट लपवण्यासाठी वापरल्या जातात.

पायथनमधील कमेंट कोणत्याही प्रोग्राम कोडचे स्पष्टीकरण देण्यासाठी वापरल्या जाऊ शकतात. हे कोड लपवण्यासाठी देखील वापरले जाऊ शकते.

कमेंट हा कार्यक्रमाचा पार्ट नाही, परंतु ते कार्यक्रमाची संवादात्मकता वाढवते आणि प्रोग्रॅम वाचनीय (रिडेबल) बनवते.

### पायथन दोन टाइप च्या कमेंट्स ना सपोर्ट देतो:

- सिंगल लाइन कमेंट
- मल्टी लाइन कमेंट

#### 1 सिंगल लाइन कमेंट

जर यूजर ला एक ओळीची कमेंट स्पेसिफाइड करायची असेल तर कमेंट '# ' ने सुरू झाली पाहिजे

#### फॉर्मॅट :

# ही सिंगल लाइन कॉमेंट आहे

#### उदाहरण: "scomment.py"

# ही सिंगल लाइन कॉमेंट आहे.

प्रिंट ("हॅलो पायथन")

## आउटपुट: \$python3 scomment.py

हॅलो पायथन

## 2 मल्टी लाइन कमेंट

तिहेरी अवतरणांमध्ये बहु-रेखा असलेली कमेंट दिली जाऊ शकते. ओळीच्या सुरुवातीस स्टार्ट करणे आवश्यक आहे.

फॉर्मॅट :

```
“ This
```

```
is
```

```
Multi line comment”
```

## उदाहरण: “mcomment.py”

```
‘ ‘ ‘ This
```

```
is
```

```
Multi line comment”
```

```
print(“Hello Python”)
```

## आउटपुट: \$python3 mcomment.py

Hello Python

उदाहरण:

## उदाहरण: “comments.py”

```
# हे उदाहरण टिप्पण्यांचा वापर दर्शवते
```

```
... print() वापरले
```

```
प्रिंट/डिस्प्ले
```

```
स्क्रीनवरील मजकूर “
```

```
प्रिंट (“Python मध्ये आपले स्वागत आहे”)
```

```
#व्हेरिबल्सना मूल्य नियुक्त करा
```

```
a=20
```

```
b=30
```

```
दोन संख्यांची बेरीज प्रिंट करा
```

```
मुद्रित करा(“गोष्टी आहे”)
```

```
छापा(a b)
```

## आउटपुट: \$python3 comments.py

Welcome to Python

Sum is

50

## Python मध्ये व्हेरिबल

व्हेरिबल हे नामांकित मेमरी पोजिशन आहे ज्यामध्ये आपण स्पेसिफिक प्रोग्रामसाठी व्हॅल्यू साठवू शकतो. दुसऱ्या शब्दांत, व्हेरिबल हे नाव आहे जे मेमरी स्थानाचा रेफरन्स देण्यासाठी वापरले जाते. व्हेरिबल आयडेंटिफायर म्हणूनही ओळखले जाते आणि व्हॅल्यू ठेवण्यासाठी वापरले जाते.

## व्हेरिबल्स तयार करणे

पायथनमध्ये, आम्हाला पायथनमध्ये स्पष्टपणे व्हेरिबल घोषित करण्याची आवश्यकता नाही. जेव्हा आपण व्हेरिबलला कोणतेही व्हॅल्यू नियुक्त करतो तेव्हा ते व्हेरिबल ऑटोमॅटिक पणे घोषित केले जाते.

पायथनमध्ये, आम्हाला व्हेरिबलचा टाइप स्पेसिफाइड करण्याची आवश्यकता नाही कारण पायथन ही एक सहज टाइप केलेली भाषा आहे. म्हणजे शिथिलपणे टाइप केलेल्या भाषेत व्हेरिबलचा टाइप स्पेसिफाइड करण्याची आवश्यकता नाही कारण व्हेरिबल नियुक्त केलेल्या मूल्यावर आधारित डेटाटाइप ऑटोमॅटिक बदलतो.

उदाहरणार्थ:

```
?? a=10????Where variable a is an integer datatype.
```

```
?? b=“Glance”????Where variable b is string datatype.
```

## व्हेरिबलचे नामकरण करण्याचे नियम:

- परिवर्तनीय नावे दोन्ही अक्षरे आणि अंकांचा समूह असू शकतात, परंतु त्यांना अक्षर किंवा अंडरस्कोरने सुरुवात करावी लागेल.
- व्हेरिबल नावासाठी लोअरकेस अक्षरे वापरण्याची शिफारस केली जाते. ‘SUM’ आणि ‘sum’ दोन्ही दोन भिन्न चल आहेत.

## उदाहरण: “vardemo.py”

```
a=10 #पूर्णांक
```

```
b=“StudyGlance” #स्ट्रिंग
```

```
c=12.5 फ्लोट
```

```
छापा(a)
```

```
छापा(b)
```

```
प्रिंट(c)
```

## आउटपुट: \$python3 vardemo.py

10

StudyGlance

12.5

मल्टिपल व्हेरिबल्सना व्हॅल्यू नियुक्त करा

Python आम्हाला एकाच स्टेटमेंटमध्ये मल्टिपल व्हेरिबल्ससाठी एक व्हॅल्यू आणि मल्टिपल व्हेरिबल्ससाठी मल्टिपल व्हॅल्यू नियुक्त करण्याची परवानगी देते ज्याला मल्टिपल असाइनमेंट म्हणून देखील ओळखले जाते.

मल्टिपल व्हेरिएबल्ससाठी सिंगल व्हॅल्यू नियुक्त करा

उदाहरण: "vardemo1.py"

```
x=y=z=50
```

```
print x
```

```
print y
```

```
print z
```

आउटपुट: `$python3 vardemo1.py`

```
50
```

```
50
```

```
50
```

एकाधिक व्हेरिएबल्सना एकाधिक मूल्ये नियुक्त करा

उदाहरण: "vardemo2.py"

```
a,b,c=5,10,15
```

```
छापा a
```

```
छापा b
```

```
छापणे c
```

आउटपुट: `$python3 vardemo2.py`

```
५
```

```
10
```

```
१५
```

© NIMI  
NOT TO BE REPUBLISHED



## डेटा टाइप आणि ऑपरेटर वापरून ऑपरेशन्स करा (Perform Operations using Data Types and Operators)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- डेटा टाइप , Python मधील ऑपरेटर.

सर्वसाधारणपणे, डेटा टाइप हे स्पेसिफाइड करतात की व्हेरिएबल्समध्ये कोणत्या टाइप चा डेटा स्टोर केला जाईल. व्हेरिएबल्समध्ये विविध डेटा प्रकारांची व्हॅल्यू असू शकतात. पायथन ही डायनॅमिकली टाईप केलेली किंवा सैल टाईप केलेली भाषा आहे, म्हणून ती घोषित करताना व्हेरिएबलचा टाइप परिभाषित करण्याची गरज नाही. इंटरप्रीटर अस्पष्टपणे व्हॅल्यू त्याच्या प्रकारासह बांधतो.

म्हणजे शिथिलपणे टाइप केलेल्या भाषेत व्हेरिएबलचा टाइप स्पेसिफाइड करण्याची आवश्यकता नाही कारण व्हेरिएबल नियुक्त केलेल्या मूल्यावर आधारित डेटाटाइप ऑटोमेटिक बदलतो.

पायथन टाइप () फंक्शन प्रदान करतो जे आम्हाला व्हेरिएबलचा टाइप तपासण्यास सक्षम करते.

पायथन खालील स्टॅण्डर्ड डेटा टाइप प्रदान करतो, ते आहेत

1 नंबर

2 स्ट्रिंग

1 नंबर

नंबर नंबर त्मक व्हॅल्यू साठवते. जेव्हा व्हेरिएबलला नंबर नियुक्त केली जाते तेव्हा पायथन नंबर टाइप व्हेरिएबल तयार करतो.

**पायथनमध्ये तीन न्यूमरिक टाइप आहेत**

a int

b float(फ्लोट)

c Complex (कॉम्प्लेक्स)

**a int**

इंट, किंवा इंटीजर (पूर्णांक), दशांश नसलेली, अमर्यादित लांबीची पूर्ण नंबर , पॉसिटिव्ह किंवा निगेटिव्ह आहे.

**फॉर्मॅट :**

a = 10

b = -12

c = १२३४५६७८९

**b फ्लोट**

फ्लोट किंवा "फ्लोटिंग पॉइंट नंबर" ही एक नंबर आहे, पॉसिटिव्ह किंवा निगेटिव्ह, ज्यामध्ये एक किंवा अधिक दशांश असतात.

**फॉर्मॅट :**

X = 1.0

Y = 12.3

Z = -13.4

**c कॉम्प्लेक्स**

जटिल नंबर काल्पनिक पार्ट म्हणून "j" ने लिहिल्या जातात.

**फॉर्मॅट :**

A = 2+5j

B = -3+4j

C = -6j

**उदाहरण: "datatypedemo1.py"**

a = 10

b = 10.5

c = 2.14j

प्रिंट ("व्हेरिएबल a चा डेटाटाइप :", प्रकार(अ))

प्रिंट("व्हेरिएबल b चा डेटाटाइप :", type(b))

प्रिंट("व्हेरिएबल c चा डेटाटाइप :", type(c))

**आउटपुट: \$python3 datatypedemo1.py**

व्हेरिएबल a चा डेटाटाइप : <class 'int'>

व्हेरिएबल b चा डेटाटाइप : <वर्ग 'float'>

व्हेरिएबल c चा डेटाटाइप : <वर्ग 'complex'>

**2 स्ट्रिंग**

स्ट्रिंगची व्याख्या अवतरण चिन्हांमध्ये दर्शविलेल्या कॅरक्टर चा क्रम म्हणून केली जाऊ शकते. पायथनमध्ये, स्ट्रिंग परिभाषित करण्यासाठी आपण सिंगल, डबल किंवा ट्रिपल कोट्स वापरू शकतो.

स्ट्रिंग हाताळणीच्या बाबतीत, ऑपरेटर + चा वापर दोन स्ट्रिंग जोडण्यासाठी केला जातो कारण ऑपरेशन "hello" + "python" "hello python" परत करते.

फॉर्मॅट :

S1='Welcome'

S2="to Python"

S3=""world"" उदाहरण:

### उदाहरण

a = "स्वागत" #दुहेरी अवतरण वापरून

b = 'पायथन' #सिंगल कोट्स वापरून

c = ""वर्ल्ड"" #तिहेरी अवतरण वापरून

प्रिंट ("व्हेरिएबल a चा डेटाटाइप :", प्रकार(अ))

दोन तार जोडण्यासाठी प्रिंट(a b) करा

### आउटपुट:

व्हेरिएबल a :<class 'str'> चा डेटाटाइप

स्वागत पायथन

### पायथनमधील ऑपरेटर

ऑपरेटरला सिम्बॉल्स म्हणून परिभाषित केले जाऊ शकते जे दोन ऑपरेंडमधील स्पेसिफिक ऑपरेशनसाठी जबाबदार आहे.

Python खालीलप्रमाणे डिस्क्रिप्शन केलेले विविध ऑपरेटर प्रदान करते.

- अर्थमेटिक ऑपरेटर
- असाइनमेंट ऑपरेटर
- कम्प्यारीजन ऑपरेटर
- बिटवाइज ऑपरेटर
- आइडेंटिटी ऑपरेटर
- लॉजिकल ऑपरेटर
- मेम्बरशिप ऑपरेटर

### 1 अर्थमेटिक ऑपरेटर

अर्थमेटिक ऑपरेटर दोन ऑपरेंडमधील अर्थमेटिक ऑपरेशन्स करण्यासाठी वापरले जातात.

ऑपरेटर	डिस्क्रिप्शन
+ (addition) - (subtraction)	दोन ऑपरेंड ऑड करा डावीकडून उजवी ऑपरेंड वजा करा
*(गुणाकार) / (स्प्लिट )	दोन ऑपरेंड गुणाकार करा डाव्या ऑपरेंडला राइटोनने स्प्लिट करा (नेहमी फ्लोटमध्ये रिझल्ट होतो)
%( reminder)	मॉड्यूलस - उजव्या बाजूने डाव्या ऑपरेंडच्या विपार्त णीचा उर्वरित पार्त
// (floor division) results	फ्लोर डिवीजन - डिवीजन की पूर्ण संख्येमध्ये समायोजित केले क्रमांक ओळीत डावीकडे
** (exponent)	एक्सपोनंट - डावा ऑपरेंड वाढवला उजव्या पॉवर पर्यंत

### उदाहरण:

x =20

y =10

print('x + y =',x+y)

print('x - y =',x-y)

print('x \* y =',x\*y)

print('x / y =',x/y)

print('x // y =',x//y)

print('x \*\* y =',x\*\*y)

### आउटपुट:

x+y=30

x-y = 10

x \* y = 200

x / y =2

x//y=0

x \*\* y = 8

### 2 असाइनमेंट ऑपरेटर

असाइनमेंट ऑपरेटर डाव्या ऑपरेंडला उजव्या अभिव्यक्तीचे व्हॅल्यू नियुक्त करण्यासाठी वापरले जातात.

ऑपरेटर	डिस्क्रिप्शन
= (याला असाइन करते)	उजव्या बाजूच्या ऑपरेंडपासून डावीकडील ऑपरेंडला मूल्ये नियुक्त करते.
+= (अॅडिशन नंतर असाइनमेंट)	ते डाव्या ऑपरेंडमध्ये उजवे ऑपरेंड जोडते आणि रिझल्ट टोलेफ्ट ऑपरेंड नियुक्त करते.
-= (वजाबाकीनंतर असाइनमेंट)	ते डाव्या ऑपरेंडमधून उजवे ऑपरेंड वजा करते आणि डाव्या ऑपरेंडला निकाल देते.
*= (गुणा नंतर असाइनमेंट)	हे उजव्या ऑपरेंडला डाव्या ऑपरेंडसह गुणाकार करते आणि निकाल डाव्या ऑपरेंडला नियुक्त करते.
/= (विभागानंतर असाइनमेंट)	ते डाव्या ऑपरेंडला उजव्या ऑपरेंडसह स्लिट करते आणि रिझल्ट डाव्या ऑपरेंडला नियुक्त करते.
%= (मॉड्यूलस नंतर असाइनमेंट)	हे दोन ऑपरेंड वापरून मॉड्यूलस घेते आणि निकाल डाव्या ऑपरेंडला नियुक्त करते.
**= (घातांकानंतर नियुक्ती)	ऑपरेटर्सवर घातांक (शक्ती) गणना करते आणि डाव्या ऑपरेंडला व्हॅल्यू नियुक्त करते.
//= (फ्लोर डिवीजन असाइनमेंट)	हे फ्लोर डिवीजन करते ऑपरेटर आणि त्यांना व्हॅल्यू नियुक्त करा डावी कार्यमथड्स .

**उदाहरण:**

**आउटपुट:**

### 3 कम्प्यारीजन ऑपरेटर

कम्प्यारीजन ऑपरेटर दोन ऑपरेंडच्या मूल्याची कम्प्यारीजन करण्यासाठी वापरले जातात आणि त्यानुसार बुलियन खरे किंवा खोटे परत करतात.

ऑपरेटर	डिस्क्रिप्शन
== (समान ते)	समान - दोन्ही ऑपरेंड समान असल्यास खरे.
!= (नाही च्या समान)	समान नाही - ऑपरेंड समान नसल्यास खरे.
<= (कमी किंवा पेक्षा समान)	पेक्षा कमी किंवा समान - जर डावी ऑपरेंड उजवीपेक्षा कमी किंवा समान असेल तर खरे
>= (मोठे किंवा पेक्षा समान)	पेक्षा मोठे किंवा बरोबर - जर डावी ऑपरेंड उजवीपेक्षा मोठी किंवा समान असेल तर खरे
< (कमी पेक्षा)	त्यापेक्षा कमी - डावी ऑपरेंड उजवीपेक्षा कमी असल्यास खरे
> (मोठे पेक्षा)	मोठे ते - डावी ऑपरेंड उजवीपेक्षा मोठी असल्यास खरे

**उदाहरण:**

x = 20

y = 10

print('x > y is', x > y)

print('x < y is', x < y)

print('x == y is', x == y)

print('x != y is', x != y)

print('x >= y is', x >= y)

print('x <= y is', x <= y)

**आउटपुट:**

x > y is True

x < y is False

x == y is False

x != y is True

x >= y is True

x >= y is False

### 4 बिटवाइज ऑपरेटर

ऑपरेटर	डिस्क्रिप्शन
आणि (बायनरी आणि)	दोन्ही बिट 1 असल्यास प्रत्येक बिट 1 वर सेट करते
(बायनरी किंवा)	दोन बिटांपैकी एक 1 असल्यास प्रत्येक बिट 1 वर सेट करते
^ (बायनरी xor)	दोनपैकी फक्त एक बिट 1 असल्यास प्रत्येक बिट 1 वर सेट करते
~ (negation)	सर्व बिट्स उलटे करतो

<< (डावीकडे शिफ्ट)	उजवीकडून शून्य आत ढकलून डावीकडे सरकवा आणि सर्वात डावीकडे खाली पडू द्या
>> (उजवीकडे शिफ्ट)	प्रती ढकलून उजवीकडे हलवा डावीकडून सर्वात डावीकडे, आणि सर्वात उजवीकडील बिट खाली पडू द्या

बिटवाइज ऑपरेटर दोन ऑपरेंडच्या व्हॅल्यू वर बिट बाय बिट ऑपरेशन करतात. उदाहरण:

#### उदाहरण

```
a =9# equal to 1001
b =12# equal to 1100
print("a & b = ",a&b)
print("a | b = ",a|b)
print("a ^ b = ",a ^ b)
print("~a = ",~a)
print("a << 2 = ",a <<2)
print("a >> 2 = ",a >>2)
```

#### आउटपुट:

```
a & b =8
a | b =13
a ^ b =5
~a =-10
a <<2=36
a >>2=2
```

#### 5 ओळख ऑपरेटर

आयडेंटिटी ऑपरेटरचा वापर ऑब्जेक्ट्सची कम्प्यारीजन करण्यासाठी केला जातो, जर ते समान असतील तर नाही, परंतु जर ते प्रत्यक्षात समान ऑब्जेक्ट असतील तर त्याच मेमरी स्थानासह.

ऑपरेटर	डिस्क्रिप्शन
is	दोन्ही व्हेरिएबल्स समान ऑब्जेक्ट असल्यास सत्य मिळवते (जर id(x) id(y) बरोबर असेल तर )
is not	दोन्ही व्हेरिएबल्स समान ऑब्जेक्ट नसल्यास खरे मिळवते

#### उदाहरण:

```
x=10
y=10
print(x is y)
print("id(x)= %d id(y)= %d"%(id(x),id(y)))
y=20
print(x isnot y)
print("id(x)= %d id(y)= %d"%(id(x),id(y)))
```

#### आउटपुट:

```
True
id(x)=10105376id(y)=10105376
True
id(x)=10105376id(y)=10105696
```

#### 6 लॉजिकल ऑपरेटर

निर्णय घेण्यासाठी लॉजिकल ऑपरेटर प्रामुख्याने अभिव्यक्ती व्हॅल्यू मापनात वापरले जातात.

ऑपरेटर	डिस्क्रिप्शन
and (logical and)	दोन्ही स्टेटमेंट असल्यास सत्य मिळवते खरे आहेत
or (logical or)	पैकी एक असल्यास खरे मिळवते स्टेटमेंट सत्य आहेत
not (logical not)	रिझल्ट उलटा, returns निकाल खरा असेल तर खोटे

#### उदाहरण:

```
x =True
y =False
print('x and y is',x and y)
print('x or y is',x or y)
print('not x is',not x)
```

#### आउटपुट:

```
x आणि y असत्य आहे
x किंवा y खरे आहे
x असत्य नाही
```

## 7 मेम्बरशिप ऑपरेटर

निर्णय घेण्यासाठी लॉजिकल ऑपरेटर प्रामुख्याने अभिव्यक्ती व्हॅल्यू मापनात वापरले जातात.

ऑपरेटर	डिस्क्रिप्शन
in	ऑब्जेक्टमध्ये (लिस्ट , ट्यूपल किंवा डिक्शनरी) स्पेसिफाइड मूल्यासह सिक्वेंस सबस्थित असल्यास ते खरे होते
not in)	सह क्रम असल्यास ते सत्य परत करते स्पेसिफाइड व्हॅल्यू सबस्थित नाही ऑब्जेक्टमध्ये (लिस्ट , Tuple किंवा डिक्शनरी)

## उदाहरण:

```
x ='Hello world'  
print('H'in x)  
print('hello'notin x)
```

## आउटपुट:

```
खरे  
खरे
```

**डिसीजन आणि लूपसह कंट्रोल फ्लो करा (Control Flow with Decisions and Loops)**

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- if, if-else, if-elif-else.

पायथनमधील कंडीशनल स्टेटमेंट परिस्थितीनुसार भिन्न गणना किंवा क्रिया करतात.

पायथनमध्ये, खालील कंडीशनल स्टेटमेंट आहेत

- if
- if-else
- if - elif -else

**इंडेंटेशन**

प्रोग्रामिंगच्या सुलभतेसाठी आणि साधेपणा प्राप्त करण्यासाठी, पायथन ब्लॉक लेव्हल कोडसाठी कंस वापरण्याची परवानगी देत नाही. Python मध्ये, इंडेंटेशन ब्लॉक घोषित करण्यासाठी वापरले जाते. जर दोन स्टेटमेंट समान इंडेंटेशन लेयर वर असतील, तर ती एकाच ब्लॉकचा पार्ट आहेत.

साधारणपणे, विधानांना इंडेंट करण्यासाठी चार स्पेस दिली जातात जी पायथनमध्ये स्पेसिफिक प्रमाणात इंडेंटेशन असतात.

इंडेंटेशन हा पायथन भाषेचा सर्वात जास्त वापरला जाणारा पार्ट आहे कारण तो कोडचा ब्लॉक घोषित करतो. एका ब्लॉकची सर्व स्टेटमेंट समान पातळीच्या इंडेंटेशनवर अभिप्रेत आहेत.

**if statements**

स्पेसिफिक स्थिती तपासण्यासाठी if स्टेटमेंट वापरले जाते. जर कंडीशन सत्य असेल, तर कोडचा एक ब्लॉक (if-block) कार्यान्वित केला जाईल.

**सिन्टेक्स :**

जर अभिव्यक्ती:

विधान:

**उदाहरण:**

```
a =33
```

```
b =200
```

```
if b > a:
```

```
    print("b is greater than a")
```

**आउटपुट:**

b a पेक्षा मोठा आहे

**If - else स्टेटमेंट**

if-else स्टेटमेंट हे इफ स्टेटमेंटसह इतर ब्लॉक प्रदान करते जे कंडीशनच्या खोल्या केसमध्ये अंमलात आणले जाते.

जर कंडीशन सत्य असेल, तर if-block कार्यान्वित होईल. अन्यथा, इतर-ब्लॉक कार्यान्वित केला जातो.

**सिन्टेक्स:**

if expression:

#block of statements

else:

#another block of statements

**उदाहरण:**

```
age =int(input("Enter your age : "))
```

```
if age >=18:
```

```
    print("You are eligible to vote !!")
```

```
else:
```

```
    print("Sorry! you have to wait !!")
```

**आउटपुट:**

Enter your age:19

You are eligible to vote!!

**If - elif - else स्टेटमेंट**

एलिफ स्टेटमेंट आम्हाला अनेक अटी तपासण्यास आणि त्यातील खऱ्या स्थितीनुसार स्टेटमेंटचे स्पेसिफिक ब्लॉक कार्यान्वित करण्यास सक्षम करते.

आमच्या गरजेनुसार आमच्या प्रोग्राममध्ये आमच्याकडे कितीही एलिफ स्टेटमेंट असू शकतात. तथापि, एलिफ वापरणे ऐच्छिक आहे.

**सिन्टेक्स :**

if expression 1:

# block of statements

elif expression 2:

# block of statements

elif expression 3:

```
# block of statements
```

else:

```
# block of statements
```

#### उदाहरण:

```
marks =int(input("Enter the marks :"))
```

```
if marks >85and marks <=100:
```

```
print("Congrats! you scored grade A..")
```

```
elif marks >60and marks <=85:
```

```
print("You scored grade B + ..")
```

```
elif marks >40and marks <=60:
```

```
print("You scored grade B ..")
```

```
elif(marks >30and marks <=40):
```

```
print("You scored grade C ..")
```

```
else:
```

```
print("Sorry you are fail ?")
```

#### आउटपुट:

गुण प्रविष्ट करा:70

तुम्ही B + ग्रेड मिळवला.

#### उदाहरण प्रोग्रॅम:

उद्दिष्ट: तीन नंबर मधील जास्तीत जास्त सर्च करण्यासाठी पायथन प्रोग्रॅम

#### उदाहरण:

```
a=int(input("Enter a value : "))
```

```
b=int(input("Enter b value : "))
```

```
c=int(input("Enter c value : "))
```

```
if(a>b)and(a>c):
```

```
print("Maximum value is :.",a)
```

```
elif(b>c):
```

```
print("Maximum value is :.",b)
```

```
else:
```

```
print("Maximum value is :.",c)
```

#### आउटपुट:

a व्हॅल्यु प्रविष्ट करा: 10

b व्हॅल्यु प्रविष्ट करा: 14

c व्हॅल्यु प्रविष्ट करा:9

कमाल व्हॅल्यु आहे: 14

## Python मध्ये लूप स्टेटमेंट (Loop Statements in Python)

उद्दिष्टे:या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

### • लूप करताना, लूपसाठी.

कधीकधी आम्हाला प्रोग्रॅमचा प्रवाह बदलण्याची आवश्यकता असू शकते. स्पेसिफिक कोडच्या अंमलबजावणीसाठी अनेक वेळा पुनरावृत्ती करण्याची आवश्यकता असल्यास आपण लूप स्टेटमेंटसाठी जाऊ शकतो.

या उद्देशासाठी, पायथन विविध टाइप चे लूप प्रदान करतात जे काही स्पेसिफिक कोड अनेक वेळा पुनरावृत्ती करण्यास सक्षम असतात. ते आहेत,

- लूप असताना (while loop)
- लूपसाठी (For loop)

#### लूप असताना (while loop)

व्हाईल लूपच्या सहाय्याने आपण स्टेटमेंट्सचा संच कार्यान्वित करू शकतो जोपर्यंत एक कंडिशन सत्य आहे. व्हाईल लूप बहुतेकदा अशा बाबतीत वापरला जातो जेथे पुनरावृत्तीची नंबर आगाऊ माहित नसते.

#### सिन्टेक्स

```
while expression:
```

```
Statement(s)
```

#### उदाहरण:

```
i=1;
```

```
while i<=3:
```

```
print(i);
```

```
i=i+1;
```

#### आउटपुट:

१

2

3

## while loop सह else वापरणे

पायथन आपल्याला इतर ब्लॉकसह while लूप वापरण्यास सक्षम करतो. while स्टेटमेंटमध्ये दिलेली कंडीशन खोटी ठरल्यावर else ब्लॉक कार्यान्वित केला जातो.

### सिन्टेक्स :

while expression:

Statements

else:

Statements

### उदाहरण:

i=1;

while i<=3:

print(i)

i=i+1;

else:print("The while loop terminated");

### आउटपुट:

1

2

3

while लूप संपुष्टात आला

## लूपसाठी (for loop)

Python मधील फॉर लूप अनेक वेळा स्टेटमेंट किंवा प्रोग्रामचा पार्ट पुनरावृत्ती करण्यासाठी वापरला जातो. लिस्ट, ट्यूपल किंवा डिक्शनरी सारख्या डेटा स्ट्रक्चर्समधून पाथ काढण्यासाठी हे वारंवार वापरले जाते.

### सिन्टेक्स :

for iterating\_var in sequence:

statement(s)

### उदाहरण:

i=1

n=int(input("Enter n value : "))

for i in range(i,n+1):

print(i,end = ' ')

### आउटपुट:

Enter n value:5

१ २ ३ ४ ५

## लूपसह else वापरणे

Python आम्हाला लूपसह else स्टेटमेंट वापरण्याची परवानगी देते जे सर्व पुनरावृत्ती संपल्यावरच कार्यान्वित केले जाऊ शकते. येथे, आपण हे लक्षात घेतले पाहिजे की जर लूपमध्ये ब्रेक स्टेटमेंट असेल तर else स्टेटमेंट कार्यान्वित केले जाणार नाही.

### सिन्टेक्स :

for iterating\_var in sequence:

statements

else:

statements

### उदाहरण:

for i in range(0,5):

print(i)

else:

print("for loop completely exhausted");

### आउटपुट:

0

1

2

3

4

for loop completely exhausted

### नोंट

रेंज () फंक्शन डीफॉल्टनुसार 0 पासून सुरू होणारी नंबर चा क्रम मिळवते आणि 1 ने वाढवते (डीफॉल्टनुसार), आणि एका स्पेसिफाइड संख्येवर समाप्त होते.

### उदाहरण:

range(6) means the values from 0 to 5.

range(2,6) means the values from 2 to 5.

### उदाहरण प्रोग्रॅम:

ध्येय: दिलेल्या क्रमांकाचा चार्ट छापण्याचा प्रोग्रॅम

### उदाहरण:

i=1;

num =int(input("Enter a number:"));

for i in range(1,11):



```
print("%d X %d = %d"%(num,i,num*i))
```

### आउटपुट:

क्रमांक प्रविष्ट करा:10

Enter a number:10

10 X 1=10

10 X 2=20

10 X 3=30

10 X 4=40

10 X 5=50

10 X 6=60

10 X 7=70

10 X 8=80

10 X 9=90

10 X 10=100

### पायथनमध्ये नेस्टेड लूप

Python आम्हाला a for loop च्या आत कितीही for loops नेस्ट करू देते. आतील लूप एक्सटर्नल लूपच्या प्रत्येक पुनरावृत्तीसाठी n वेळा अंमलात आणला जातो.

### सिन्टेक्स :

for iterating\_var1 in sequence:

for iterating\_var2 in sequence:

#block of statements

### उदाहरण प्रोग्रॅम:

**ध्येय:** दिलेल्या संख्येच्या ताऱ्यांच्या री प्रिंटेड करण्याचा प्रोग्रॅम

उदाहरण:

```
=int(input("Enter the no.of rows you want to print : "))
```

```
for i inrange(0,n):
```

```
for j inrange(0,i+1):
```

```
print("**",end="")
```

```
print()
```

### आउटपुट:

तुम्हाला प्रिंटेड करायचे असलेल्या री ची नंबर एंटर करा:4

```
*
```

```
**
```

```
***
```

```
****
```

## डॉक्युमेंट्स आणि स्ट्रक्चर कोड (Document and Structure Code)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

• डॉक्युमेंट्स आणि स्ट्रक्चर कोड.

## Python मध्ये क्रम लिस्ट

python मध्ये, एक लिस्ट व्हॅल्यू किंवा विविध टाइप च्या वस्तूंचा संग्रह म्हणून परिभाषित केली जाऊ शकते. लिस्ट तील आयटम स्वल्पविरामाने विभक्त केले आहेत (,) आणि चौरस कंस [] सह संलग्न आहेत.

## सिन्टेक्स

```
List =[value1, value2, value3,....]
```

## उदाहरण

```
L0 = [] #creates empty list
```

```
L1 =[123,"python",3.7]
```

```
L2 =[1,2,3,4,5,6]
```

```
L3 =["C Programing","Java","Python"]
```

```
print(L0)
```

```
print(L1)
```

```
print(L2)
```

```
print(L3)
```

## ऑपरेटरची यादी करा

ऑपरेटर	डिस्क्रिप्शन
+	दोन याद्या जोडण्यासाठी वापरला जाणारा कॉन्कटेनेशन ऑपरेटर म्हणून ओळखला जातो
*	हे पुनरावृत्ती ऑपरेटर म्हणून ओळखले जाते. हे एकाच लिस्ट च्या अनेक प्रती एकत्र करते.
[]	तो स्लाइस ऑपरेटर म्हणून ओळखला जातो. हे लिस्ट मधून लिस्ट आयटममध्ये एक्सेस करण्यासाठी वापरले जाते.
[:]	हे रेंज स्लाइस ऑपरेटर म्हणून ओळखले जाते. हे लिस्ट मधून लिस्ट आयटमच्या रेन्ज मध्ये एक्सेस करण्यासाठी वापरले जाते.
in	हे मेम्बरशिप ऑपरेटर म्हणून ओळखले जाते. स्पेसिफाइड लिस्ट मध्ये स्पेसिफिक आयटम सबस्थित असल्यास ते परत येते.
not in	हे मेम्बरशिप ऑपरेटर देखील आहे आणि लिस्ट मध्ये स्पेसिफिक लिस्ट आयटम सबस्थित नसल्यास ते सत्य परत करते.

## उदाहरण:

```
num=[1,2,3,4,5]
```

```
lang=['python','c','java','php']
```

```
print(num+lang)
```

```
print(num*2)
```

```
print(lang[2])
```

## आउटपुट

```
[]
```

```
[123,'python',3.7]
```

```
[1,2,3,4,5,6].
```

```
['C Programing','Java','Python']
```

## लिस्ट इंडेक्स

इंडेक्सिंगची प्रक्रिया स्ट्रिंगच्या प्रमाणेच केली जाते. स्लाइस ऑपरेटर [] वापरून लिस्ट तील एलिमेंटमध्ये एक्सेस केला जाऊ शकतो.

इंडेक्स 0 पासून सुरू होते, लिस्ट चा पहिला एलिमेंट 0व्या सिक्सेस गिकेवर स्टोर केला जातो, लिस्ट चा दुसरा एलिमेंट 1व्या निर्देशांकावर स्टोर केला जातो आणि असेच.

Python आम्हाला नकारात्मक इंडेक्स वापरण्याची लवचिकता देखील प्रदान करते. ऋण निर्देशांक उजवीकडून मोजले जातात. लिस्ट च्या शेवटच्या एलिमेंटमध्ये (सर्वात उजवीकडे) आहे इंडेक्स -1, त्याचा समीप डावा एलिमेंट निर्देशांक -2 वर सबस्थित असतो आणि डावीकडे सर्वात जास्त एलिमेंट येईपर्यंत.

```
print(lang[1:4])
```

```
print('cpp'in lang)
```

```
print(6notin num)
```

## आउटपुट:

```
[1,2,3,4,5,'python','c','java','php']
```

```
[1,2,3,4,5,1,2,3,4,5]
```

java

['c', 'java', 'php']

False

True

### लिस्ट मध्ये एलिमेंट कसे ऑड करायचे किंवा बदलायचे?

Python आम्हाला स्लाइस आणि असाइनमेंट ऑपरेटर वापरून लिस्ट आयटम सुधारण्याची परवानगी देते. एखादी वस्तू किंवा आयटमची रेन्ज बदलण्यासाठी आम्ही असाइनमेंट ऑपरेटर (=) वापरू शकतो.

#### उदाहरण:

```
num=[1,2,3,4,5]
```

```
print(num)
```

```
num[2]=30
```

```
print(num)
```

```
num[1:3]=[25,36]
```

```
print(num)
```

```
num[4]="Python"
```

```
print(num)
```

#### आउटपुट:

```
[1,2,3,4,5]
```

```
[1,2,30,4,5]
```

```
[1,25,36,4,5]
```

```
[1,25,36,4,'Python']
```

### लिस्ट मधून एलिमेंट कसे डिलेट करायचे किंवा काढायचे?

Python आम्हाला del कीवर्ड वापरून यादीतील एक किंवा अधिक आयटम हटविण्याची परवानगी देतो.

#### उदाहरण:

```
num=[1,2,3,4,5]
```

```
print(num)
```

```
delnum[1]
```

```
print(num)
```

```
delnum[1:3]
```

```
print(num)
```

#### आउटपुट:

```
[1,2,3,4,5]
```

```
[1,3,4,5]
```

```
[1,5]
```

### फंक्शन्सची यादी करा

Python खालील अंगभूत फंक्शन्स प्रदान करते जे लिस्ट सह वापरले जाऊ शकतात.

- len()
- max()
- min()
- list()
- sum()
- sorted()
- append()
- remove()
- sort()
- reverse()
- count()
- index()
- insert()
- pop()
- clear()

#### len()

Python len() मध्ये यादीची लांबी सर्च प्यासाठी वापरली जाते, म्हणजे ती यादीतील आयटमची नंबर परत करते.

#### सिन्टेक्स :

```
len(list)
```

#### उदाहरण:

```
num=[1,2,3,4,5,6]
```

```
print("length of list:",len(num))
```

#### आउटपुट:

```
यादीची लांबी: 6
```

```
?max()
```

Python max() मध्ये लिस्ट तील कमाल व्हॅल्यू सर्च करण्यासाठी वापरले जाते

#### सिन्टेक्स :

```
max(list)
```

**उदाहरण:**

```
num=[1,2,3,4,5,6]
lang=['java','c','python','cpp']
print("Max of list :",max(num))
print("Max of list :",max(lang))
```

**आउटपुट:**

लिस्ट ची कमाल: 6

लिस्ट ची कमाल: पायथन

```
min()
```

Python मध्ये min() चा वापर यादीतील किमान व्हॅल्यू सर्च करण्यासाठी केला जातो

**सिन्टेक्स :**

```
min(list)
```

**उदाहरण:**

```
num=[1,2,3,4,5,6]
lang=['java','c','python','cpp']
print("Min of list :",min(num))
print("Min of list :",min(lang))
```

**आउटपुट:**

लिस्ट ची किमान: 1

यादीतील किमान: c

**sum()**

python मध्ये, sum(list) फंक्शन लिस्ट मधील सर्व व्हॅल्यू ची बेरीज देते. लिस्ट व्हॅल्यू नंबर प्रकारात असणे आवश्यक आहे.

**सिन्टेक्स :**

```
sum(list)
```

**उदाहरण:**

```
num=[1,2,3,4,5,6]
print("sum of list items :",sum(num))
```

**आउटपुट:**

लिस्ट आयटमची बेरीज: 21

**sorted()**

python मध्ये, क्रमबद्ध (लिस्ट) फंक्शनचा वापर लिस्ट तील सर्व आयटमची चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावण्यासाठी केला जातो.

**सिन्टेक्स :**

```
sorted(list)
```

**उदाहरण:**

```
num=[1,3,2,4,6,5]
lang=['java','c','python','cpp']
print(sorted(num))
print(sorted(lang))
```

**आउटपुट:**

```
[1,2,3,4,5,6]
['c','cpp','java','python']
```

**list()**

list() मेथड्स सिक्सेस टाइप घेते आणि त्यांना लिस्ट मध्ये रूपांतरित करते. हे दिलेल्या स्ट्रिंग किंवा Tupleला लिस्ट मध्ये रूपांतरित करण्यासाठी वापरले जाते.

**सिन्टेक्स :**

```
list(sequence)
```

**उदाहरण:**

```
str="python"
list1=list(str)
print(list1)
```

**आउटपुट:**

```
['p','y','t','h','o','n']
['p', 'y', 't', 'h', 'o', 'n']
```

**append ()**

python append() मेथड्स मध्ये लिस्ट च्या शेवटी एक आयटम जोडला जातो.

**सिन्टेक्स :**

```
list.append(item)
```

आयटम नंबर, स्ट्रिंग, लिस्ट आणि इत्यादी असू शकतात

**उदाहरण:**

```
num=[1,2,3,4,5]
lang=['python','c','java','php']
num.append(6)
print(num)
```

```
lang.append("cpp")
```

```
print(lang)
```

### आउटपुट:

```
[1,2,3,4,5,6]
```

```
['python','c','java','php','cpp']
```

### remove()

python remove() मेथड्स मध्ये यादीतील पहिला आयटम काढून टाकला जातो जो पास केलेल्या मूल्याप्रमाणे असतो. जर आयटम लिस्ट मध्ये नसेल तर तो एक त्रुटी टाकतो.

### सिन्टेक्स :

```
list.remove(item)
```

आयटम नंबर, स्ट्रिंग, लिस्ट आणि इत्यादी असू शकतात

### उदाहरण:

```
num=[1,2,3,4,5]
```

```
lang=['python','c','java','php','c']
```

```
num.remove(2)
```

```
print(num)
```

```
lang.remove("c"># first occurrence will remove
```

```
print(lang)
```

```
lang.remove("cpp")
```

```
print(lang)
```

### आउटपुट:

```
[1,3,4,5]
```

```
['python','java','php','c']
```

```
ValueError:list.remove(x):x not in list
```

### sort()

python sort() मेथड्स मध्ये यादीतील एलिमेंटची क्रमवारी (सॉर्टिंग) लावली जाते. हे आयटमची उतरत्या आणि चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते. हे पर्यायी पॅरामीटर 'रिव्हर्स' घेते जे लिस्ट ला उतरत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते. डीफॉल्टनुसार, लिस्ट एलिमेंटची चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते.

### सिन्टेक्स :

```
list.sort([reverse=true])
```

reverse, which displays items in descending

### उदाहरण:

```
lang = ['p','y','t','h','o','n']# Char list
```

```
even = [6,8,2,4,10]# int list
```

```
print(lang)
```

```
print(even)
```

```
lang.sort()
```

```
even.sort()
```

```
print("\nAfter Sorting:\n")
```

```
print(lang)
```

```
print(even)
```

```
print("In Descending Order :\n")
```

```
even.sort(reverse=True)
```

```
print(even)
```

### आउटपुट:

```
['p','y','t','h','o','n']
```

```
[6,8,2,4,10]
```

```
After Sorting:
```

```
['h','n','o','p','t','y']
```

```
[2,4,6,8,10]
```

```
In Descending Order :
```

```
[10,8,6,4,2]
```

### reverse()

python reverse() मेथड्स मध्ये यादीतील एलिमेंट उलटे केले जातात. लिस्ट रिकामी असल्यास, ती फक्त रिकामी यादी परत करते. उलट केल्यानंतर लिस्ट चे शेवटचे इंडेक्स व्हॅल्यू 0 इंडेक्सवर असेल.

### सिन्टेक्स :

```
list.reverse ()
```

### उदाहरण:

```
lang = ['p','y','t','h','o','n']
```

```
lang = ['p','y','t','h','o','n']
```

```
print("After reverse")
```

```
lang.reverse()
```

```
print(lang)
```

### आउटपुट:

उलट केल्यानंतर

```
['n','o','h','t','y','p']
```

### count()

python count() मेथड्स मध्ये लिस्ट मध्ये एलिमेंट किती वेळा दिसला ते दाखवते. लिस्ट मध्ये एलिमेंट सबस्थित नसल्यास, ते 0 मिळवते.

### सिन्टेक्स :

```
list.count(item)
```

### उदाहरण:

```
num=[1,2,3,4,3,2,2,1,3,4,5,7,8]
```

```
cnt=num.count(2)
```

```
print("Count of 2 is:",cnt)
```

```
cnt=num.count(10)
```

```
print("Count of 10 is:",cnt)
```

### आउटपुट:

2 ची नंबर आहे:3

10 ची नंबर : 0

### index()

पायथन इंडेक्स () मेथड्स मध्ये पास केलेल्या एलिमेंटची इंडेक्स मिळते. ही मेथड्स एक आर्ग्यूमेंट्स घेते आणि त्याची इंडेक्स मिळवते. एलिमेंट सबस्थित नसल्यास, ते व्हॅल्यु त्रुटी वाढवते. लिस्ट मध्ये डुप्लिकेट एलिमेंट असल्यास, ते प्रथम आलेल्या एलिमेंटची इंडेक्स मिळवते.

या मेथड्स मध्ये आणखी दोन पर्यायी पॅरामीटर्स स्टार्ट आणि एंड लागतात जे एका मर्यादित इंडेक्स सर्च प्यासाठी वापरले जातात.

### सिन्टेक्स :

```
list.index(x[, start[, end]])
```

### उदाहरण:

```
lang=['p','y','t','h','o','n','p','r','o','g','r','a','m']
```

```
print("index of t is:",lang.index('t'))
```

```
print("index of p is:",lang.index('p'))
```

```
print("index of p is:",lang.index('p',3,10))
```

```
print("index of p is:",lang.index('z'))
```

### आउटपुट:

index of t is:2

index of p is:0

index of p is:6

ValueError:'z'isnotinlist

### (insert())

python insert() मेथड्स मध्ये लिस्ट तील स्पेसिफाइड निर्देशांकावर एलिमेंट समाविष्ट केला जातो. पहिला आर्ग्यूमेंट्स हा एलिमेंटची इंडेक्स आहे ज्याच्या आधी एलिमेंट इन्सर्ट यचा आहे.

### सिन्टेक्स :

```
list.insert(i,x)
```

i: कोणत्या एलिमेंटवर इंडेक्स टाकला जाईल.

x : एलिमेंट इन्सर्ट यचा आहे.

### उदाहरण:

```
num=[10,20,30,40,50]
```

```
num.insert(4,60)
```

```
print("updated list is :",num)
```

```
num.insert(7,70)
```

```
print("updated list is :",num)
```

### आउटपुट:

updated listis:[10,20,30,40,60,50]

updated listis:[10,20,30,40,60,50,70]

### pop()

python pop() मध्ये एलिमेंट लिस्ट मधून स्पेसिफाइड निर्देशांकावर सबस्थित असलेला एलिमेंट काढून टाकतो. हे पॉप केलेले एलिमेंट परत करते.

### सिन्टेक्स

```
list.pop([i])
```

### उदाहरण:

```
num=[10,20,30,40,50]
```

```
num.pop()
```

```
print("updated list is :",num)
```

```
num.pop(2)
```

```
print("updated list is :",num)
```

```
num.pop(7)
```

```
print("updated list is :",num)
```

### आउटपुट:

updated listis:[10,20,30,40]

updated listis:[10,20,40]

IndexError: pop index out of range

### clear()

python clear() मेथड्स मध्ये यादीतील सर्व एलिमेंट काढून टाकले जातात. हे यादी पूर्णपणे क्लियर करते आणि काहीही परत करत नाही.

### सिन्टेक्स :

```
list.clear()
```

### उदाहरण:

```
num=[10,20,30,40,50]
```

```
num.clear()
```

```
print("After clearing ",num)
```

### आउटपुट:

```
क्लियर केल्यानंतर [ ]
```

## Python मध्ये Tuple सिक्वेन्स (Tuple Sequence in Python)

**उद्दिष्ट:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- Tuple
- इंडेक्सिंग
- Tuple ऑपरेटर
- लिस्ट मध्ये एलिमेंट कसे ऑड करायचे किंवा काढायचे?
- Tuple फंक्शन्स.

Python मध्ये Tuple चा वापर अपरिवर्तनीय पायथन ऑब्जेक्ट्सचा क्रम स्टोर करण्यासाठी केला जातो. ट्यूपल हे लिस्ट सारखेच आहे कारण लिस्ट मध्ये स्टोर केलेल्या वस्तूंचे व्हॅल्यू बदलले जाऊ शकते तर ट्यूपल अपरिवर्तनीय आहे आणि ट्यूपलमध्ये स्टोर केलेल्या वस्तूंचे व्हॅल्यू बदलले जाऊ शकत नाही.

लहान कंसात बंद केलेल्या स्वल्पविराम विभक्त व्हॅल्यू चा संग्रह म्हणून Tuple लिहिता येते.

### सिन्टेक्स :

```
Tuple=(value1, value2...)
```

### उदाहरण:

```
T1 =()
```

```
T2 =(10,30,20,40,60)
```

```
T3 =(“C”,“Java”,“Python”)
```

```
T4 =(501,“abc”,19.5)
```

```
T5 =(90,)
```

```
print(T1)
```

```
print(T2)
```

```
print(T3)
```

```
print(T4)
```

```
print(T5)
```

### आउटपुट:

```
()
```

```
(10,30,20,40,60)
```

```
(‘C’,‘Java’,‘Python’)
```

```
(501,‘abc’,19.5)
```

```
(90,)
```

### Tuple इंडेक्स :

ट्यूपलमधील इंडेक्स आणि स्लाइसिंग याद्यांप्रमाणेच आहेत. ट्यूपलमधील इंडेक्स 0 पासून सुरू होते आणि लांबी (ट्यूपल) - 1 पर्यंत जाते.

स्लाइस ऑपरेटर वापरून ट्यूपलमधील आयटममध्ये एक्सेस केला जाऊ शकतो. पायथन आम्हाला कोलन ऑपरेटरचा वापर ट्यूपलमधील मल्टिपल आयटममध्ये एक्सेस करण्यासाठी देखील परवानगी देतो.

इतर भाषांप्रमाणे, python आम्हाला नकारात्मक इंडेक्स देखील वापरण्याची लवचिकता प्रदान करते. ऋण निर्देशांक उजवीकडून मोजले जातात. ट्यूपलच्या शेवटच्या एलिमेंटमध्ये (सर्वात उजवीकडे) निर्देशांक -1 असतो, त्याच्या जवळचा डावा एलिमेंट निर्देशांक -2 वर सबस्थित असतो आणि डावीकडे सर्वात जास्त एलिमेंट येईपर्यंत.

### Tuple ऑपरेटर

ऑपरेटर	डिस्क्रिप्शन
+	t ला दोन ट्यूपल्स एकत्र करण्यासाठी वापरला जाणारा कंकटेनेशन ऑपरेटर म्हणून ओळखले जाते
*	t ला पुनरावृत्ती ऑपरेटर म्हणून ओळखले जाते. हे एकाच ट्यूपलच्या अनेक प्रती एकत्र करते.

[ ]	तो स्लाइस ऑपरेटर म्हणून ओळखला जातो. हे tuple वरून आयटम ऍक्सेस करण्यासाठी वापरले जाते.
[ : ]	t ला रेंज स्लाइस ऑपरेटर म्हणून ओळखले जाते. हे ट्यूपलमधून आयटमच्या रेंज मध्ये ऍक्सेस करण्यासाठी वापरले जाते.
in	हे मेम्बरशिप ऑपरेटर म्हणून ओळखले जाते. स्पेसिफाइड Tuple मध्ये स्पेसिफिक आयटम सबस्थित असल्यास ते परत येते.
not in	हे एक मेम्बरशिप ऑपरेटर देखील आहे आणि ट्यूपलमध्ये स्पेसिफिक आयटम सबस्थित नसल्यास ते सत्य परत करते.

#### उदाहरण:

```
num=(1,2,3,4,5)
lang=('python','c','java','php')
print(num+lang)
print(num*2)
print(lang[2])
print(lang[1:4])
print('cpp'in lang)
print(6notin num)
```

#### आउटपुट:

```
(1,2,3,4,5,'पायथन','c','java','php')
(1,2,3,4,5,1,2,3,4,5)
java
('c', 'java', 'php')
True
True
```

#### लिस्ट मध्ये एलिमेंट कसे ऑड करायचे किंवा काढायचे?

लिस्ट च्या विपरीत, ट्यूपल आयटम अद्ययावत किंवा हटविले जाऊ शकत नाहीत कारण ट्यूपल्स अपरिवर्तनीय आहेत. संपूर्ण ट्यूपल डिलीट करण्यासाठी, आपण Tuple नावासह del कीवर्ड वापरू शकतो.

#### उदाहरण:

```
mytuple=('python','c','java','php')
mytuple[3]="html"
# 'tuple' object does not support item assignment
print(mytuple)
delmytuple[3]
# 'tuple' object doesn't support item deletion
print(mytuple)
del mytuple
#deletes entire tuple
```

#### आउटपुट:

```
'tuple'object आयटम असाइनमेंटला सपोर्ट देत नाही
'tuple'object आयटम हटवण्यास सपोर्ट देत नाही
```

#### Tuple फंक्शन

पायथन खालील अंगभूत फंक्शन्स पुरवतो जे टपल्ससह वापरले जाऊ शकतात.

- len()
- len()
- max()
- min()
- tuple()
- sum()
- sorted()
- index()
- count()

#### len()

पायथन len() मध्ये ट्यूपलची लांबी सर्च प्यासाठी वापरला जातो, म्हणजेच ते ट्यूपलमधील आयटमची नंबर परत करते.

#### सिन्टेक्स :

```
len(tuple)
```

#### उदाहरण:

```
num=(1,2,3,4,5,6)
print("length of tuple:",len(num))
```

#### आउटपुट:

```
Tupleची लांबी: 6
```

#### max()

Python max() मध्ये tuple मधील कमाल व्हॅल्यू सर्च प्यासाठी वापरले जाते.

#### सिन्टेक्स :

```
max(tuple)
```



### उदाहरण:

```
num=(1,2,3,4,5,6)
lang=('java','c','python','cpp')
print("Max of tuple :",max(num))
print("Max of tuple :",max(lang))
```

### आउटपुट:

Tupleची कमाल:6  
Tupleची कमाल: पायथन

### min()

Python min() मध्ये tuple मधील किमान व्हॅल्यू सर्च प्यासाठी वापरले जाते.

### सिन्टेक्स :

```
min(tuple)
```

### उदाहरण:

```
num=(1,2,3,4,5,6)
lang=('java','c','python','cpp')
print("Min of tuple :",min(num))
print("Min of tuple :",min(lang))
```

### आउटपुट:

Tupleची किमान: 1  
Tupleचे किमान: c

### sum()

python मध्ये, sum(tuple) फंक्शन ट्यूपलमधील सर्व व्हॅल्यू ची बेरीज मिळवते. Tuple व्हॅल्यू नंबर प्रकारात असणे आवश्यक आहे.

### सिन्टेक्स :

```
sum(tuple)
```

### उदाहरण:

```
num=(1,2,3,4,5,6)
print("sum of tuple items :",sum(num))
```

### आउटपुट:

Tuple आयटमची बेरीज: 21

### sorted()

पायथनमध्ये, सॉर्टेड (ट्यूपल) फंक्शनचा वापर ट्यूपलच्या सर्व वस्तू चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावण्यासाठी केला जातो. हे आयटमची उतरत्या

आणि चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते. हे एक पर्यायी पॅरामीटर 'रिव्हर्स' घेते जे Tupleला उतरत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते.

### सिन्टेक्स :

```
sorted(tuple[,reverse=True])
```

### उदाहरण:

```
num=(1,3,2,4,6,5)
lang=('java','c','python','cpp')
print(sorted(num))
print(sorted(lang))
print(sorted(num,reverse=True))
```

### आउटपुट:

```
(1,2,3,4,5,6)
('c','cpp','java','python')
(6,5,4,3,2,1)
```

### tuple (sequence)

tuple() मेथड्स सिक्सेस टाइप घेते आणि त्यांना tuples मध्ये रूपांतरित करते. हे दिलेल्या स्ट्रिंग किंवा लिस्ट ला ट्यूपलमध्ये रूपांतरित करण्यासाठी वापरले जाते.

### सिन्टेक्स :

```
uple(sequence)
```

### उदाहरण:

```
str="python"
tuple1=tuple(str)
print(tuple1)
num=[1,2,3,4,5,6]
tuple2=tuple(num)
print(tuple2)
```

### आउटपुट:

```
('p','y','t','h','o','n')
(1,2,3,4,5,6)
```

### count()

python count() मेथड्स मध्ये ट्यूपलमध्ये एलिमेंट किती वेळा दिसला ते दाखवते. ट्यूपलमध्ये एलिमेंट सबस्थित नसल्यास, ते 0 मिळवते.

## सिन्टेक्स :

```
tuple.count(item)
```

## उदाहरण:

```
num=(1,2,3,4,3,2,2,1,3,4,5,7,8)
cnt=num.count(2)
print("Count of 2 is:",cnt)
cnt=num.count(10)
print("Count of 10 is:",cnt)
```

## आउटपुट:

2 ची नंबर आहे:3  
10 ची नंबर : 0

## Index()

पायथन इंडेक्स () मेथड्स मध्ये पास केलेल्या एलिमेंटची इंडेक्स मिळते. ही मेथड्स एक आर्ग्युमेंट्स घेते आणि त्याची इंडेक्स मिळवते. एलिमेंट सबस्थित नसल्यास, ते व्हॅल्यू त्रुटी वाढवते.

ट्युपलमध्ये डुप्लिकेट एलिमेंट असल्यास, ते प्रथम आलेल्या एलिमेंटची इंडेक्स मिळवते.

## Python मध्ये सिक्सेस सेट करा (Set Sequence in Python)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- एक संच तयार करणे
- ऑपरेटर सेट करा
- फंक्शन्स सेट करा.

पायथनमध्ये, कुरळे ब्रेसेसमध्ये बंद केलेल्या विविध वस्तूंचा अक्रमित संग्रह म्हणून सेटची व्याख्या केली जाऊ शकते. सेटचे एलिमेंट डुप्लिकेट असू शकत नाहीत. पायथन सेटचे एलिमेंट अपरिवर्तनीय असणे आवश्यक आहे.

पायथनमधील इतर संग्रहांप्रमाणे, सेटच्या एलिमेंटशी कोणतीही इंडेक्स जोडलेली नाही, म्हणजे, आम्ही निर्देशांकाद्वारे सेटच्या कोणत्याही एलिमेंटमध्ये थेट एक्सेस करू शकत नाही. तथापि, आम्ही ते सर्व एकत्र प्रिंटेड करू शकतो किंवा आम्ही संचातून लूप करून एलिमेंटची यादी मिळवू शकतो.

## एक संच तयार करणे

स्वल्पविरामाने विभक्त केलेल्या वस्तूंना कुरळे ब्रेसेससह बंद करून सेट तयार केला जाऊ शकतो.

## सिन्टेक्स :

```
Set={value1, value2....}
```

या मेथड्स मध्ये आणखी दोन पर्यायी पॅरामीटर्स स्टार्ट आणि एंड लागतात जे एका मर्यादित इंडेक्स सर्च प्यासाठी वापरले जातात.

## सिन्टेक्स :

```
tuple.index(x[, start[, end]])
```

## उदाहरण:

```
lang = ('p','y','t','h','o','n','p','r','o','g','r','a','m')
print("index of t is:",lang.index('t'))
print("index of p is:",lang.index('p'))
print("index of p is:",lang.index('p',3,10))
print("index of p is:",lang.index('z'))
```

## आउटपुट:

t ची इंडेक्स आहे:2  
p ची इंडेक्स आहे:0  
p ची इंडेक्स आहे:6  
ValueError:'z'isnotin tuple.

## उदाहरण:

```
Days ={"Monday","Tuesday","Wednesday ","Thursday",
"Friday","Saturday","Sunday"}
print(Days)
print(type(Days))
print("Looping through the set elements ... ")
for i in Days:
print(i)
```

## आउटपुट:

```
{'बुधवार', 'मंगळवार', 'Sun', 'शुक्रवार', 'गुरुवार',
'शनिवार', 'सोमवार'}
सेट एलिमेंटमधून वळण घेत आहे...
बुधवार
```

मंगळवार

Sun

शुक्रवार

गुरुवार

शनिवार

सोमवार

### ऑपरेटर सेट करा

पायथनमध्ये, आम्ही पायथन सेटवर विविध गणिती ऑपरेटर्स करू शकतो जसे की युनियन, इंटरसेक्शन, डिफरन्स इ.

ऑपरेटर	डिस्क्रिप्शन
	युनियन ऑपरेटर
&	छेदनपॉइंट ऑपरेटर
-	फरक ऑपरेटर:

### युनियन (|) ऑपरेटर

किंवा (|) ऑपरेटर वापरून दोन संचांचे मिलन मोजले जाते. दोन संचांच्या मिलनामध्ये दोन्ही संचांमध्ये सबस्थित असलेल्या सर्व वस्तूंचा समावेश होतो.

### उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
print(Days1 | Days2)
```

### आउटपुट:

```
{'गुरु', 'शुक्र', 'रवि', 'मंगळ', 'बुध', 'सोम', 'शनि'}
```

### इंटरसेक्शन (&) ऑपरेटर

& (इंटरसेक्शन) ऑपरेटरचा वापर पायथनमधील दोन संचांच्या छेदनपॉइंट ची गणना करण्यासाठी केला जातो. दोन संचांचे छेदनपॉइंट दोन्ही संचांमध्ये समान असलेल्या एलिमेंटांचा संच म्हणून दिले आहेत.

### उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
print(Days1 & Days2)
```

### आउटपुट:

```
{'Mon','Sat'}
```

### डिफरन्स (-) ऑपरेटर

वजाबाकी (-) ऑपरेटर वापरून दोन संचातील फरक काढता येतो. संच 2 मध्ये सबस्थित असलेल्या संच 1 मधील सर्व एलिमेंट काढून परिणामी संच प्राप्त केला जाईल.

### उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
Print(Days1 - Days2)
```

### आउटपुट:

```
{'मंगळ', 'बुध'}
```

### फंक्शन्स सेट करा

Python मध्ये सेटसह वापरल्या जाणाऱ्या खालील मेथड्स आहेत. ते आहेत

- लें (सेट)
- len(set)
- max(set)
- min(set)
- sum(set)
- sorted(set)
- set()
- add()
- update()
- discard()
- remove()
- pop()
- clear()
- union()
- intersection()
- difference()
- issubset()
- issuperset()

### len(set)

Python len() मध्ये सेटची लांबी सर्च प्यासाठी वापरला जातो, म्हणजेच तो सेटमधील आयटमची नंबर देतो.

### सिन्टेक्स :

```
len(set)
```

### उदाहरण:

```
num={1,2,3,4,5,6}
print("length of set:",len(num))
```

### आउटपुट:

सेटची लांबी: 6

### max()

Python max() मध्ये सेटमधील कमाल व्हॅल्यू सर्च प्यासाठी वापरला जातो.

### सिन्टेक्स :

```
max(set)
```

### उदाहरण:

```
num={1,2,3,4,5,6}
lang={'java','c','python','cpp'}
print("Max of set :",max(num))
print("Max of set :",max(lang))
```

### आउटपुट:

सेटची कमाल: 6

सेटची कमाल: पायथन

### min()

Python मध्ये min() संचातील किमान व्हॅल्यू सर्च प्यासाठी वापरला जातो.

### सिन्टेक्स :

```
min(set)
```

### उदाहरण:

```
num={1,2,3,4,5,6}
lang={'java','c','python','cpp'}
print("Min of set :",min(num))
print("Min of set :",min(lang))
```

### आउटपुट:

सेटची किमान: 1

सेटची किमान: c

### sum()

python मध्ये, sum(set) फंक्शन सेटमधील सर्व व्हॅल्यू ची बेरीज देते. नंबर प्रकारात व्हॅल्यू सेट करणे आवश्यक आहे.

### सिन्टेक्स :

```
sum(set)
```

### उदाहरण:

```
num={1,2,3,4,5,6}
```

```
print("sum of set items :",sum(num))
```

### आउटपुट:

सेट आयटमची बेरीज: 21

### sorted()

python मध्ये, सॉर्टेड (सेट) फंक्शनचा वापर सेटच्या सर्व आयटमची चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावण्यासाठी केला जातो. हे आयटमची उतरत्या आणि चढत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते. हे पर्यायी पॅरामीटर 'रिव्हर्स' घेते जे सेटला उतरत्या क्रमाने क्रमवारी (सॉर्टिंग) लावते.

### सिन्टेक्स :

```
sorted(set[,reverse=True])
```

### उदाहरण:

```
num={1,3,2,4,6,5}
lang={'java','c','python','cpp'}
print(sorted(num))
print(sorted(lang))
print(sorted(num,reverse=True))
```

### आउटपुट:

```
{1,2,3,4,5,6}
{'c', 'cpp', 'java', 'python'}
{6,5,4,3,2,1}
```

### set()

सेट() मेथड्स सिक्सेस टाइप घेते आणि त्यांना सेटमध्ये रूपांतरित करते. हे दिलेली स्ट्रिंग किंवा लिस्ट किंवा Tuple सेटमध्ये रूपांतरित करण्यासाठी वापरले जाते.

### सिन्टेक्स :

```
set(sequence)
```

### उदाहरण:

```
set1=set("PYTHON")
print(set1)
days=["Mon","Tue","Wed","Thur","Fri","Sat","Sun"]
set2 =set(days)
print(set2)
days=("Mon","Tue","Wed","Thur","Fri","Sat","Sun")
set3 =set(days)
print(set3)
```

### आउटपुट:

```
{'N', 'O', 'T', 'H', 'P', 'Y'}
```

```
{'शुक्र', 'गुरु', 'मंगळ', 'रवि', 'सोम', 'शनि', 'बुध'}
```

```
{'शुक्र', 'गुरु', 'मंगळ', 'रवि', 'सोम', 'शनि', 'बुध'}
```

### add()

python मध्ये, add() मेथड्स सेटमध्ये काही स्पेसिफिक आयटम जोडण्यासाठी वापरली जाते.

### सिन्टेक्स :

```
set.add (item)
```

### उदाहरण:

```
Days = { "Monday", "Tuesday", "Wednesday",  
"Thursday", "Friday" }
```

```
print("\n printing the original set ... ")
```

```
print(Days)
```

```
Days.add("Saturday");
```

```
Days.add("Sunday");
```

```
print("\n Printing the modified set...");
```

```
print(Days)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

```
{'बुधवार', 'शुक्रवार', 'गुरुवार', 'मंगळवार', 'सोमवार'}
```

सुधारित संच प्रिंटेड करत आहे...

```
{'बुधवार', 'Sun', 'शुक्रवार', 'गुरुवार', 'मंगळवार'
```

```
,'शनिवार', 'सोमवार'}
```

### update()

Python सेटमध्ये एकापेक्षा जास्त आयटम जोडण्यासाठी अपडेट () मेथड्स प्रदान करते.

### सिन्टेक्स :

```
set.update([item1, item2...])
```

### उदाहरण:

```
Months={"Jan","Feb","Mar","Apr"}
```

```
print("\n Printing the original set ... ")
```

```
print(Months)
```

```
Months.update (["May","Jun","Jul"])
```

```
print("\n Printing the modified set...");
```

```
print(Months)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

```
{'मार', 'एप्रिल', 'जाने', 'फेब्रु'}
```

सुधारित संच प्रिंटेड करत आहे...

```
{'मार', 'एप्रिल', 'जाने', 'जून', 'मे', 'जुलै', 'फेब्रुवारी'}
```

### discard()

Python discard () मेथड्स प्रदान करते जी सेटमधून आयटम काढण्यासाठी वापरली जाऊ शकते. सेटमध्ये आयटम अस्तित्वात नसल्यास, पायथन त्रुटी देणार नाही. प्रोग्रॅम त्याचे कंट्रोल प्रवाह राखतो.

### सिन्टेक्स :

```
set.discard(item)
```

### उदाहरण:

```
Months={"Jan","Feb","Mar","Apr"}
```

```
print("\n printing the original set ... ")
```

```
print(Months)
```

```
Months.discard("Apr")
```

```
print("\n Printing the modified set...");
```

```
print(Months)
```

```
Months.discard("May") #doesn't give error
```

```
print("\n Printing the modified set...");
```

```
print(Months)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

```
{'जाने', 'एप्रिल', 'मार', 'फेब्रु'}
```

सुधारित संच प्रिंटेड करत आहे...

```
{'जाने', 'मार', 'फेब्रु'}
```

सुधारित संच प्रिंटेड करत आहे...

```
{'जाने', 'मार', 'फेब्रु'}
```

### remove()

पायथन रिमूव्ह () मेथड्स प्रदान करते जी सेटमधून आयटम काढण्यासाठी वापरली जाऊ शकते. सेटमध्ये आयटम अस्तित्वात नसल्यास, पायथन त्रुटी देईल.

### सिन्टेक्स :

```
set.remove(item)
```

### उदाहरण:

```
Months={"Jan","Feb","Mar","Apr"}
print("\n printing the original set ... ")
print(Months)
Months.remove("Apr")
print("\n Printing the modified set...");
print(Months)
Months.remove("May") #it give error
print("\n Printing the modified set...");
print(Months)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

{'फेब्रु', 'जाने', 'एप्रिल', 'मार'}

सुधारित संच प्रिंटेड करत आहे...

{'फेब्रु', 'जाने', 'मार'}

मुख्य त्रुटी: 'मे' अस्तित्वात नाही.

### pop()

पायथनमध्ये, आयटम काढण्यासाठी पॉप () मेथड्स वापरली जाते. तथापि, ही मेथड्स नेहमीच शेवटची वस्तू काढून टाकते.

### सिन्टेक्स :

```
set.pop ()
```

### उदाहरण:

```
Days = {"Monday","Tuesday", "Wednesday", "Thursday",
"Friday"}
print("\n printing the original set ... ")
print(Days)
Days.pop()
print("\n Printing the modified set...");
print(Days)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

{'सोमवार', 'बुधवार', 'शुक्रवार', 'मंगळवार', 'गुरुवार'} सुधारित संच प्रिंटेड करत आहे...

{'बुधवार', 'शुक्रवार', 'मंगळवार', 'गुरुवार'}

### clear()

पायथनमध्ये, सेटमधील सर्व आयटम काढण्यासाठी क्लियर () मेथड्स वापरली जाते.

### सिन्टेक्स :

```
set.clear()
```

### उदाहरण:

```
Days = {"Monday","Tuesday","Wednesday", "Thursday",
"Friday"}
print("\n printing the original set ... ")
print(Days)
Days.clear()
print("\n Printing the modified set...");
print(Days)
```

### आउटपुट:

मूळ संच प्रिंटेड करत आहे...

{'सोमवार', 'बुधवार', 'शुक्रवार', 'मंगळवार', 'गुरुवार'} सुधारित संच प्रिंटेड करत आहे...

सेट()

### union ()

पायथनमध्ये, युनियन () मेथड्स चा वापर दोन संचांचे एकत्रीकरण करण्यासाठी केला जातो. दोन संचांच्या मिलनामध्ये दोन्ही संचांमध्ये सबस्थित असलेल्या सर्व वस्तूंचा समावेश होतो.

### सिन्टेक्स :

```
set1.union (set2)
```

### उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
print(Days1.union(Days2))
```

### आउटपुट:

{'गुरु', 'शुक्र', 'रवि', 'मंगळ', 'बुध', 'सोम', 'शनि'}

### intersection ()

पायथनमध्ये, पायथनमधील दोन संचांच्या छेदनपॉइंट ची गणना करण्यासाठी छेदनपॉइंट () वापरला जातो. दोन संचांचे छेदनपॉइंट दोन्ही संचांमध्ये समान असलेल्या एलिमेंटचा संच म्हणून दिलेला आहे.

सिन्टेक्स :

```
set1.intersection (set2)
```

उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
print(Days1.intersection(Days2))
```

आउटपुट:

```
{'सोम', 'शनि'}
```

**difference ()**

फरक () मेथड्स चा वापर करून दोन संचांचा फरक काढता येतो. संच 2 मध्ये सबस्थित असलेल्या संच 1 मधील सर्व एलिमेंट काढून परिणामी संच प्राप्त केला जाईल.

सिन्टेक्स :

```
set1.difference (set2)
```

उदाहरण:

```
Days1={"Mon","Tue","Wed","Sat"}
Days2={"Thr","Fri","Sat","Sun","Mon"}
print(Days1.difference(Days2))
```

आउटपुट:

```
{'मंगळ', 'बुध'}
```

**issubset()**

जर सेटचे सर्व एलिमेंट दुसऱ्या सेटमध्ये असतील तर (आर्ग्यूमेंट्स म्हणून पास) असल्यास issubset() मेथड्स True मिळवते. नसल्यास, ते False परत करते.

सिन्टेक्स :

```
set1.issubset (set2)
```

उदाहरण:

```
set1={1,2,3,4}
set2={1,2,3,4,5,6,7,8,9}
print(set1.issubset(set2))
print(set2.issubset(set1))
```

आउटपुट:

```
खरे
```

```
खोटे
```

**issuperset()**

एखाद्या सेटमध्ये दुसऱ्या सेटचे प्रत्येक एलिमेंट असल्यास (आर्ग्यूमेंट्स म्हणून पास) असल्यास issuperset () मेथड्स True मिळवते. नसल्यास, ते False परत करते.

सिन्टेक्स :

```
set1.issuperset (set2)
```

उदाहरण:

```
set1={1,2,3,4}
set2={1,2,3,4,5,6,7,8,9}
print(set1.issuperset(set2))
print(set2.issuperset(set1))
```

आउटपुट:

```
खोटे
```

```
खरे
```

## Python मध्ये डिक्शनरी सिक्वेन्स (Dictionary Sequence in Python)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- डिक्शनरी व्हॅल्यू मध्ये एक्सेस करणे
- डिक्शनरी व्हॅल्यू अपडेट करणे
- डिक्शनरी व्हॅल्यू हटवणे
- लूप वापरून डिक्शनरी व्हॅल्यू एक्सेस करणे
- डिक्शनरी फंक्शन.

पायथनमध्ये डिक्शनरी म्हणजे की-व्हॅल्यू जोड्यांचा संग्रह आहे जेथे व्हॅल्यू कोणतेही पायथन ऑब्जेक्ट असू शकते तर की अपरिवर्तनीय पायथन ऑब्जेक्ट आहेत, म्हणजे नंबर, स्ट्रिंग किंवा Tuple.

स्वल्पविराम(,) ने विभक्त केलेल्या आणि कर्ली ब्रेसिस {} मध्ये बंद केलेल्या मल्टिपल की-व्हॅल्यू जोड्या वापरून डिक्शनरी तयार केला जाऊ शकतो.

सिन्टेक्स :

```
Dict={key1:value1, key2:value2, .....}
```

उदाहरण:

```
student ={"Name": "Kiran", "Age": 22, "Regno": 562, "Branch": "CSE"}
```

```
print(student)
```

### आउटपुट:

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

### डिक्शनरी व्हॅल्यू मध्ये एक्सेस करणे

लिस्ट मध्ये डेटा एक्सेस केला जाऊ शकतो आणि इंडेक्स वापरून Tuple केला जाऊ शकतो.

तथापि, किल्ली वापरून डिक्शनरीमध्ये व्हॅल्यू पाहिली जाऊ शकतात कारण की डिक्शनरीमध्ये अद्वितीय आहेत.

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562, "Branch": "CSE"}
```

```
print("Name : ", student["Name"])
```

```
print("Age : ", student["Age"])
```

```
print("RegNo : ", student["Regno"])
```

```
print("Branch : ", student["Branch"])
```

### आउटपुट:

नाव : किरण

वय: 22

राज्य : ५६२

शाखा: CSE

### डिक्शनरी व्हॅल्यू अपडेट करत आहे

डिक्शनरी हा एक परिवर्तनीय डेटा टाइप आहे आणि त्याची व्हॅल्यू स्पेसिफिक की वापरून अपडेट केली जाऊ शकतात.

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562, "Branch": "CSE"}
```

```
print("printing student data .... ")
```

```
print(student)
```

```
student["Name"] = "Kishore"
```

```
print("printing updated data .... ")
```

```
print(student)
```

### आउटपुट:

विद्यार्थ्यांचा डेटा प्रिंट करत आहे....

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

अपडेटेड डेटा प्रिंट करत आहे....

```
{'नाव': 'किशोर', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

### डिक्शनरी व्हॅल्यू डिलेट करणे

del कीवर्ड वापरून शब्दकोशातील आयटम हटवता येतात.

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562, "Branch": "CSE"}
```

```
print("printing student data .... ")
```

```
print(student)
```

```
student["Name"] = "Kishore"
```

```
print("printing updated data .... ")
```

```
print(student)
```

### आउटपुट:

विद्यार्थ्यांचा डेटा प्रिंट करत आहे....

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

सुधारित इनफार्मेशन प्रिंट करणे

```
{'नाव': 'किरण', 'वय': 22, 'राज्य': 562}
```

### लूप वापरून डिक्शनरी व्हॅल्यू मध्ये एक्सेस करणे

फॉर लूप वापरून डिक्शनरी पुनरावृत्ती केली जाऊ शकते. आम्ही फक्त की, फक्त व्हॅल्यू आणि दोन्ही की आणि व्हॅल्यू मध्ये एक्सेस करू शकतो.

शब्दकोशाच्या सर्व कळ्या प्रिंट करा

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562, "Branch": "CSE"}
```

```
print("Keys are :")
```

```
for x in student:
```

```
print(x)
```

### आउटपुट:

की आहेत:

नाव

वय

राज्य

शाखा

शब्दकोशाची सर्व व्हॅल्यू प्रिंट करा



**उदाहरण:**

```
print("values are :")
for x in student:
    print(student[x])
```

**आउटपुट:**

```
values are :
Kiran
22
562
CSE
print all the Keys & values of a dictionary
```

**उदाहरण:**

```
print("Key and values are :")
for x,y in student.items():
    print(x,y)
```

**आउटपुट:**

```
व्हॅल्यू आहेत:
कॉल
22
५६२
CSE
शब्दकोशाच्या सर्व की आणि व्हॅल्यू प्रिंटेड कराउदाहरण:
```

**उदाहरण**

```
print("Key and values are :")
for x,y in student.items():
    print(x,y)
```

**आउटपुट:**

```
की आणि व्हॅल्यू आहेत:
नाव किरण
वय 22
राजवट 562
शाखा CSE
```

**डिक्शनरी फंक्शन**

पायथन खालील इन-बुलिट फंक्शन्सना सपोर्ट करतो, ती म्हणजे •

- len()
- copy()
- get()
- keys()
- items()
- values()
- update()
- pop()
- clear()

**len()**

python मध्ये, len() फंक्शन दिलेल्या शब्दकोशाची लांबी सर्च प्यासाठी वापरले जाते.सिन्टेक्स :

**सिन्टेक्स**

```
len(dictionary)
```

**उदाहरण:**

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
           "Branch": "CSE"}
print("Length of Dictionary is:", len(student))
```

**आउटपुट:**

शब्दकोशाची लांबी आहे: 4

**copy()**

हे दिलेल्या शब्दकोशाची दुसरी प्रत परत करते.

**सिन्टेक्स :**

```
dictionary.copy()
```

**उदाहरण:**

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
           "Branch": "CSE"}
student2 = student.copy()
print(student2)
```

**आउटपुट:**

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

### get()

python मध्ये, get() हे स्पेसिफाइड की फॉर्म डिक्शनरीचे व्हॅल्यू मिळविण्यासाठी वापरले जाते.सिन्टेक्स :

### सिन्टेक्स

```
dictionary.get()
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
"Branch": "CSE"}

print("Name is :", student.get("Name"))

print("RegNo is :", student.get("Regno"))
```

### आउटपुट:

नाव : किरण

RegNo आहे:562

### keys()

python keys() मध्ये डिक्शनरीमधून सर्व की आणण्यासाठी मेथड्स वापरली जाते

### सिन्टेक्स :

```
dictionary.keys()
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
"Branch": "CSE"}

for x in student.keys():

print(x)
```

### आउटपुट:

नाव

वय

राज्य

शाखा

### Items()

python items() मेथडमध्ये डिक्शनरीचे नवीन व्ह मिळतात. हे व्ह की व्हॅल्यू ट्युपल्सचे संकलन आहे.

### सिन्टेक्स :

```
dictionary.items()
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
"Branch": "CSE"}

for x in student.items():

print(x)
```

### आउटपुट:

( 'नाव', 'किरण' )

( 'वय', 22 )

( 'राज्य', 562 )

( 'शाखा', 'CSE' )

### Values()

python values() मध्ये डिक्शनरीतील सर्व व्हॅल्यू गोळा करण्यासाठी मेथड्स वापरली जाते.

### सिन्टेक्स :

```
Dictionary.values()
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
"Branch": "CSE"}

for x in student.values():

print(x)
```

### आउटपुट:

किरण

22

५६२

CSE

### update()

python update() मेथड्स मध्ये की आणि व्हॅल्यू जोड्यांसह डिक्शनरी अपडेट करते. जर ते सबस्थित नसेल तर ते की/व्हॅल्यू समाविष्ट करते. की/व्हॅल्यू अद्ययावत करते जर ते आधीपासून डिक्शनरीमध्ये असेल.

### सिन्टेक्स :

```
Dictionary.update({key:value,...})
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno": 562,
"Branch": "CSE"}

student.update({"Regno": 590})
```

```
student.update({"phno":56895})
```

```
print(student)
```

### आउटपुट:

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 590, 'शाखा': 'CSE', 'phno': 56895}
```

### pop()

python pop() मेथड्स मध्ये डिक्शनरीमधून एक एलिमेंट काढून टाकला जातो. हे स्पेसिफाइड कीशी संबंधित एलिमेंट काढून टाकते.

डिक्शनरीमध्ये स्पेसिफाइड की सबस्थित असल्यास, ती काढून टाकते आणि त्याचे व्हॅल्यू परत करते. स्पेसिफाइड की सबस्थित नसल्यास, ती त्रुटी KeyError टाकते.

### सिन्टेक्स :

```
Dictionary.remove(key)
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno" : 562, "Branch": "CSE"}
```

```
student.pop('Age')
```

```
print(student)
```

```
student.pop('hallno')
```

```
print(student)
```

## पायथन अरे (Python Arrays)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

• अरेचे एलिमेंट.

टीप: Python मध्ये Arrays साठी अंगभूत सपोर्ट नाही, परंतु Python लिस्ट त्याऐवजी वापरल्या जाऊ शकतात.

### अरे

टीप: हे पृष्ठ तुम्हाला ARRAYS म्हणून लिस्ट कशी वापरायची हे दाखवते, तथापि, पायथनमध्ये अरेसह कार्य करण्यासाठी तुम्हाला NumPy लायब्ररीसारखी लायब्ररी इंपोर्ट करावी लागेल.

एका व्हेरिएबलमध्ये अनेक व्हॅल्यू संचयित करण्यासाठी अरेचा वापर केला जातो:

### उदाहरण

कारची नावे असलेली अरे तयार करा:

```
cars = ["Ford", "Volvo", "BMW"]
```

Try it Yourself "

### आउटपुट:

```
{'नाव': 'किरण', 'राज्य': 562, 'शाखा': 'CSE'}
```

मुख्य त्रुटी: 'हॉलनो'

### clear()

python मध्ये, clear() चा वापर डिक्शनरीतील सर्व आयटम हटवण्यासाठी केला जातो.

### सिन्टेक्स :

```
Dictionary.clear()
```

### उदाहरण:

```
student = {"Name": "Kiran", "Age": 22, "Regno" : 562, "Branch": "CSE"}
```

```
print(student)
```

```
student.clear()
```

```
print(student)
```

### आउटपुट:

```
{'नाव': 'किरण', 'वय': 22, 'रेग्नो': 562, 'शाखा': 'CSE'}
```

```
{}
```

### अरे म्हणजे काय?

अरे हे एक स्पेशल व्हेरिएबल आहे, जे एका वेळी एकापेक्षा जास्त व्हॅल्यू धारण करू शकते.

तुमच्याकडे आयटमची लिस्ट असल्यास (उदाहरणार्थ, कारच्या नावांची लिस्ट), कार सिंगल व्हेरिएबलमध्ये स्टोर करणे यासारखे दिसू शकते:

```
car1 = "फोर्ड"
```

```
car2 = "व्होल्वो"
```

```
car3 = "BMW"
```

तथापि, जर तुम्हाला कारमधून पळवाट काढायची असेल आणि एखादी स्पेसिफिक शोधायची असेल तर? आणि जर तुमच्याकडे 3 कार नसतील तर 300 असतील तर?

उपाय एक अरे आहे!

अरे एकाच नावाखाली अनेक व्हॅल्यू धारण करू शकते आणि तुम्ही इंडेक्स क्रमांकाचा रेफरन्स देऊन व्हॅल्यू मध्ये एक्सेस करू शकता.

## अरेच्या एलिमेंटमध्ये एक्सेस करा

तुम्ही इंडेक्स क्रमांकाचा रेफरेन्स देऊन अरे एलिमेंटचा रेफरेन्स घेता.

### उदाहरण

पहिल्या अरे आयटमचे व्हॅल्यू मिळवा:

```
x = कार[0]
```

स्वतः करून पहा”

### उदाहरण

पहिल्या अरे आयटमचे व्हॅल्यू सुधारित करा:

```
cars[0] = "टोयोटा"
```

### ते स्वतः वापरून पहा

#### अरेची लांबी

अरेची लांबी (अरेमधील एलिमेंटची नंबर ) परत करण्यासाठी len() मेथड्स वापरा.

### उदाहरण

कार अरेमधील एलिमेंटची नंबर परत करा:

```
x = लेन (कार)
```

स्वतः करून पहा”

**टीप:** अरेची लांबी नेहमीच सर्वाच्च अरे इंडेक्सपेक्षा एक जास्त असते.

### लूपिंग अरे एलिमेंट

अरेच्या सर्व एलिमेंटमधून लूप करण्यासाठी तुम्ही फॉर इन लूप वापरू शकता.

### उदाहरण

कार अरेमध्ये प्रत्येक आयटम प्रिंटेड करा:

कार मध्ये x साठी:

```
प्रिंट(x)
```

स्वतः करून पहा”

### अरे एलिमेंट जोडत आहे

अरेमध्ये एलिमेंट जोडण्यासाठी तुम्ही append() मेथड्स वापरू शकता.

### उदाहरण

कार अरेमध्ये आणखी एक एलिमेंट ऑड करा:

```
cars.append("होडा")
```

### स्वतः करून पहा”

अरे एलिमेंट काढून टाकत आहे अरेमधून एलिमेंट काढून टाकण्यासाठी तुम्ही pop() मेथड्स वापरू शकता.

### उदाहरण

कार अरेचा दुसरा एलिमेंट डिलेट करा:

```
cars.pop(1)
```

### ते स्वतः वापरून पहा

अरेमधून एलिमेंट काढून टाकण्यासाठी तुम्ही remove() मेथड्स देखील वापरू शकता.

### उदाहरण

“व्होल्वो” व्हॅल्यू असलेला एलिमेंट डिलेट करा:

```
cars.remove("Volvo")
```

स्वतः करून पहा”

**टीप:** लिस्ट ची remove() मेथड्स केवळ स्पेसिफाइड मूल्याची पहिली इवेंट्स काढून टाकते.

### अरे मेथड्स

पायथनमध्ये अंगभूत पद्धतींचा संच आहे जो तुम्ही लिस्ट /अरेवर वापरू शकता.

मेथड्स	डिस्क्रिप्शन
append()	लिस्ट च्या शेवटी एक एलिमेंट जोडते
clear()	लिस्ट मधून सर्व एलिमेंट काढून टाकते
copy()	लिस्ट ची एक प्रत परत करते
count()	स्पेसिफाइड मूल्यासह एलिमेंटची नंबर मिळवते
extend()	करंट लिस्ट च्या शेवटी लिस्ट चे एलिमेंट (किंवा कोणतेही पुनरावृत्ती करण्यायोग्य) ऑड करा
index()	नमूद केलेल्या मूल्यासह पहिल्या एलिमेंटची इंडेक्स मिळवते
insert()	स्पेसिफाइड स्थानावर एक एलिमेंट जोडते
pop()	स्पेसिफाइड स्थानावर एलिमेंट काढून टाकते
remove()	स्पेसिफाइड मूल्यासह प्रथम आयटम काढून टाकते
reverse()	लिस्ट चा क्रम उलट करतो
sort()	यादीची क्रमवारी (सॉर्टिंग) लावते

**टीप:** Python मध्ये Arrays साठी अंगभूत सपोर्ट नाही, परंतु Python लिस्ट त्याऐवजी वापरल्या जाऊ शकतात.

## मॉड्यूल्स आणि टूल्स वापरून ऑपरेशन्स करा (Perform Operations Using Modules and Tools)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- मॉड्यूल तयार करणे
- लोडिंग मॉड्यूल
- मॉड्यूलचे नाव बदला.

### Python मध्ये मॉड्यूल

पायथन मॉड्यूलमध्ये पायथन प्रोग्राम फाइल म्हणून परिभाषित केले जाऊ शकते ज्यामध्ये पायथन फंक्शन्स, क्लास किंवा व्हेरिएबल्ससह पायथन कोड असतो. दुसऱ्या शब्दांत, आम्ही म्हणू शकतो की आमची पायथन कोड फाइल एक्स्टेंशन (.py) सह सेव्ह केली आहे ती मॉड्यूल मानली जाते.

Python मधील मॉड्यूल्स लॉजिकल मेथड्स ने कोड व्यवस्थित करण्याची लवचिकता प्रदान करतात. एका मॉड्यूलची कार्यक्षमता दुसऱ्यामध्ये वापरण्यासाठी, आपल्याला स्पेसिफिक मॉड्यूल आयात करणे आवश्यक आहे.

मॉड्यूल तयार करणे

#### उदाहरण: demo.py

```
# Python Module example

def sum(a,b):

return a+b

defsub(a,b):

return a-b

defmul(a,b):

return a*b

defdiv(a,b):

return a/b
```

वरील उदाहरणात आपण डेमो नावाच्या मॉड्यूलमध्ये sum(), sub(), mul() आणि div() 4 फंक्शन्स परिभाषित केली आहेत.

आमच्या पायथन कोडमध्ये मॉड्यूल लोड करत आहे

त्याची कार्यक्षमता वापरण्यासाठी आम्हाला आमच्या पायथन कोडमध्ये मॉड्यूल लोड करणे आवश्यक आहे. पायथन खाली परिभाषित केल्याप्रमाणे दोन टाईप ची स्टेटमेंट प्रदान करतो.

- 1 इम्पोर्ट स्टेटमेंट
- 2 फ्रॉम इम्पोर्ट स्टेटमेंट

### 1 इम्पोर्ट स्टेटमेंट

इंपोर्ट स्टेटमेंटचा वापर एका मॉड्यूलची सर्व कार्यक्षमता दुसऱ्या मॉड्यूलमध्ये आयात करण्यासाठी केला जातो. येथे, आपण हे लक्षात घेतले पाहिजे की आपण कोणत्याही python सोर्स फाइलची कार्यक्षमता वापरू शकतो आणि ती फाइल मॉड्यूल म्हणून दुसऱ्या पायथन सोर्स फाइलमध्ये आयात करू शकतो.

आम्ही एकाच आयात विधानासह अनेक मॉड्यूल्स आयात करू शकतो.

**सिन्टेक्स :**

```
import module1,module2..
```

**उदाहरण:**

```
import demo           #importing entire Module

a=int(input("Enter a :"))
b=int(input("Enter b :"))

print("Sum is :",demo.sum(a,b))
print("Sub is :",demo.sub(a,b))
print("Mul is :",demo.mul(a,b))
print("Div is :",demo.div(a,b))
```

**आउटपुट:**

```
एंटर करा :12
b :6 एंटर करा
Sum is: 18
sub is:6
Mul is:72
Div is: 2.0
```

### 2 फ्रॉम इम्पोर्ट स्टेटमेंट

नेमस्पेसमध्ये संपूर्ण मॉड्यूल आयात करण्याऐवजी, python केवळ मॉड्यूलचे स्पेसिफिक प्रॉपर्टीज आयात करण्याची लवचिकता प्रदान करते. हे फ्रॉम - इंपोर्ट स्टेटमेंट वापरून केले जाऊ शकते. अशा परिस्थितीत आम्ही डॉट ऑपरेटर वापरत नाही.

## सिन्टेक्स :

```
from module-name import*
```

## उदाहरण:

```
from demo import*
a=int(input("Enter a :"))
b=int(input("Enter b :"))
print("Sum is :",sum(a,b))
print("Sub is :",sub(a,b))
print("Mul is :",mul(a,b))
print("Div is :",div(a,b))
```

## आउटपुट:

```
Enter a :12
Enter b :6
Sum is:18
Sub is:6
Mul is:72
Div is:2.0
```

आम्ही संपूर्ण मॉड्यूल आयात न करता मॉड्यूलमधून स्पेसिफिक कार्य आयात करू शकतो. येथे एक उदाहरण आहे.

## सिन्टेक्स :

```
from module-name import function1,function2...
```

## उदाहरण:

```
from demo importsub,mul
#importing specific functionality from Module
a=int(input("Enter a :"))
```

## पायथन इटरेटर्स (Python Iterators)

उद्दिष्टे:या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- इटरेटरद्वारे लूपिंग
- एक इटरेटर तयार करा
- इटरेटर थांबवा.

## पायथन इटरेटर्स

पुनरावृत्ती करणारा एक ऑब्जेक्ट आहे ज्यामध्ये मोजण्यायोग्य व्हॅल्यू ची नंबर असते.

पुनरावृत्ती करणारा एक ऑब्जेक्ट आहे ज्यावर पुनरावृत्ती केली जाऊ शकते, याचा अर्थ असा की आपण सर्व व्हॅल्यू मधून जाऊ शकता.

```
b=int(input("Enter b :"))
print("Sub is :",sub(a,b))
print("Mul is :",mul(a,b))
```

## आउटपुट:

```
एंटर करा :12
b :6 एंटर करा
सब आहे:6
Mul आहे:72
```

## मॉड्यूलचे नाव बदलत आहे

Python आम्हाला स्पेसिफिक नावाने काही मॉड्यूल आयात करण्याची लवचिकता प्रदान करते जेणेकरून आम्ही आमच्या पायथन सोर्स फाइलमध्ये ते मॉड्यूल वापरण्यासाठी हे नाव वापरू शकतो.

## सिन्टेक्स :

```
import module-name as specific-name
```

## उदाहरण:

```
import demo as c
a=int(input("Enter a :"))
b=int(input("Enter b :"))
print("Sum is :",c.sum(a,b))
print("Sub is :",c.sub(a,b))
```

## आउटपुट:

```
एंटर करा :25
b :12 एंटर करा
बेरीज आहे: 37
सब आहे:13
```

## इटरेटर वि इटरेबल

याद्या, ट्यूपल्स, डिक्शनरी आणि संच सर्व पुनरावृत्ती करण्यायोग्य वस्तू आहेत. ते पुनरावृत्ती करण्यायोग्य कंटेनर आहेत ज्यातून आपण पुनरावृत्ती मिळवू शकता.

या सर्व ऑब्जेक्ट्समध्ये एक iter() मेथड्स आहे जी इटरेटर मिळविण्यासाठी वापरली जाते:

### उदाहरण

Return an iterator from a tuple, and print each value:

```
mytuple = ("apple", "banana", "cherry")
```

```
myit = iter(mytuple)
```

```
print(next(myit))
```

```
print(next(myit))
```

```
print(next(myit))
```

### स्वतः करून पहा

अगदी स्ट्रिंग देखील पुनरावृत्ती करण्यायोग्य वस्तू आहेत आणि ते पुनरावृत्ती करणारा परत करू शकतात: उदाहरण

```
mystr = "banana"
```

```
myit = iter(mystr)
```

```
print(next(myit))
```

```
print(next(myit))
```

```
print(next(myit))
```

```
print(next(myit))
```

```
print(next(myit))
```

```
print(next(myit))
```

### स्वतः करून पहा

### इटरेटरद्वारे लूपिंग

पुनरावृत्ती करण्यायोग्य ऑब्जेक्टद्वारे पुनरावृत्ती करण्यासाठी आपण फॉर लूप देखील वापरू शकतो:

उदाहरण

Iterate the values of a tuple:

```
mytuple = ("apple", "banana", "cherry")
```

```
for x in mytuple:
```

```
    print(x)
```

स्वतः करून पहा

## उदाहरण

Iterate the characters of a string:

```
mystr = "banana"
```

```
for x in mystr:
```

```
    print(x)
```

### स्वतः करून पहा

फॉर लूप प्रत्यक्षात एक इटरेटर ऑब्जेक्ट तयार करतो आणि प्रत्येक लूपसाठी पुढील() मेथड्स कार्यान्वित करतो.

### इटरेटर तयार करा

इटरेटर म्हणून ऑब्जेक्ट/क्लास तयार करण्यासाठी तुम्हाला तुमच्या ऑब्जेक्टमध्ये \_\_iter\_\_() आणि \_\_next\_\_() मेथड्स लागू कराव्या लागतील.

तुम्ही पायथन क्लासेस/ऑब्जेक्ट्स चॅप्टरमध्ये शिकल्याप्रमाणे, सर्व क्लासेसमध्ये \_\_init\_\_() नावाचे फंक्शन असते, जे तुम्हाला ऑब्जेक्ट बनवताना काही इनिशियलायझिंग करण्यास अनुमती देते.

\_\_iter\_\_() मेथड्स सारखीच कार्य करते, तुम्ही ऑपरेशन्स करू शकता (स्टार्ट करणे इ.), परंतु नेहमी इटरेटर ऑब्जेक्ट स्वतःच परत करणे आवश्यक आहे.

\_\_next\_\_() मेथड्स तुम्हाला ऑपरेशन्स करण्यास देखील परवानगी देते, आणि अनुक्रमात पुढील आयटम परत करणे आवश्यक आहे.

### उदाहरण

एक पुनरावृत्ती तयार करा जो 1 ने सुरू होणारी नंबर परत करेल आणि प्रत्येक क्रम एक ने वाढेल (1,2,3,4,5 इ. परत करणे):

```
class MyNumbers:
```

```
    def __iter__(self):
```

```
        self.a = 1
```

```
        return self
```

```
    def __next__(self):
```

```
        x = self.a
```

```
        self.a += 1
```

```
        return x
```

```
myclass = MyNumbers()
```

```
myiter = iter(myclass)
```

```
print(next(myiter))
```

```
print(next(myiter))
```

```
print(next(myiter))
```

```
print(next(myiter))
```

```
print(next(myiter))
```

स्वतः करून पहा”

### थांबवणे

तुमच्याकडे पुरेशी पुढील() स्टेटमेंट असल्यास किंवा फॉर लूपमध्ये वापरली असल्यास वरील उदाहरण कायमचे चालू राहिल.

पुनरावृत्ती कायमचे चालू ठेवण्यासाठी, आम्ही StopIteration स्टेटमेंट वापरू शकतो.

\_\_next\_\_() मेथड्स मध्ये, जर पुनरावृत्ती स्पेसिफाइड वेळा केली गेली असेल तर त्रुटी वाढवण्यासाठी आम्ही समाप्ती स्थिती जोडू शकतो:

### उदाहरण

Stop after 20 iterations:

```
class MyNumbers:
    def __iter__(self):
        self.a = 1
        return self
    def __next__(self):
        if self.a <= 20:
            x = self.a
            self.a += 1
            return x
        else:
            raise StopIteration
myclass = MyNumbers()
myiter = iter(myclass)
for x in myiter:
    print(x)
```

### पायथन गणित

Python मध्ये अंगभूत गणित फंक्शन्सचा एक संच आहे, ज्यामध्ये विस्तृत गणित मॉड्यूल समाविष्ट आहे, जे तुम्हाला नंबर वर गणिती फंक्शन करण्यास अनुमती देते.

### अंगभूत गणित फंक्शन

min() आणि max() फंक्शन्सचा वापर पुनरावृत्ती करण्यायोग्य मध्ये सर्वात कमी किंवा सर्वोच्च व्हॅल्यू सर्च प्यासाठी केला जाऊ शकतो:

### उदाहरण

```
x = min(5, 10, 25)
y = max(5, 10, 25)
print(x)
print(y)
```

स्वतः करून पहा”

abs() फंक्शन स्पेसिफाइड संख्येचे परिपूर्ण (सकारात्मक) व्हॅल्यू मिळवते:उदाहरण

### उदाहरण

```
x = abs(-7.25)
print(x)
```

स्वतः करून पहा”

pow(x, y) फंक्शन x चे व्हॅल्यू y (xy) च्या बळावर मिळवते.उदाहरण

### उदाहरण

```
x = pow(4, 3)
print(x)
स्वतः करून पहा”
```

### गणित मॉड्यूल

पायथनमध्ये गणित नावाचे एक अंगभूत मॉड्यूल देखील आहे, जे गणितीय कार्याची लिस्ट विस्तृत करते.

ते वापरण्यासाठी, आपण गणित मॉड्यूल आयात करणे आवश्यक आहे:

### इम्पोर्ट गणित

तुम्ही गणित मॉड्यूल आयात केल्यावर, तुम्ही मॉड्यूलच्या मेथड्स आणि कॉन्स्टंट्स वापरणे सुरू करू शकता.

math.sqrt() मेथड्स उदाहरणार्थ, संख्येचे क्लास मूळ मिळवते:

### उदाहरण

```
import math
x = math.sqrt(64)
print(x)
```

स्वतः करून पहा”

math.ceil() मेथड्स एखाद्या संख्येला त्याच्या जवळच्या पूर्णांकापर्यंत पूर्ण करते, आणि math.floor() मेथड्स एखाद्या संख्येला त्याच्या जवळच्या पूर्णांकापर्यंत पूर्ण करते आणि रिझल्ट मिळवते:

### उदाहरण

```
import math
x = math.ceil(1.4)
y = math.floor(1.4)
print(x) # returns 2
print(y) # returns 1
```



स्वतः करून पहा”

math.pi कॉन्स्टंट्स, PI चे व्हॅल्यू मिळवते (3.14...):

**उदाहरण**

```
import math  
  
x = math.pi  
  
print(x)
```

**संपूर्ण गणित मॉड्यूल रेफरेन्स**

आमच्या मॅथ मॉड्यूल संदर्भामध्ये तुम्हाला मॅथ मॉड्यूलशी संबंधित सर्व मेथड्स आणि स्थिरांकांचा संपूर्ण रेफरेन्स मिळेल.

**पायथन युजर इनपुट**

**युजर इनपुट**

पायथन युजर इनपुटसाठी परवानगी देतो.

याचा अर्थ आम्ही यूजर ला इनपुटसाठी विचारण्यास सक्षम आहोत.

Python 2.7 पेक्षा Python 3.6 मध्ये मेथड्स थोडी वेगळी आहे.

Python 3.6 input() मेथड्स वापरते.

Python 2.7 raw\_input() मेथड्स वापरते.

खालील उदाहरण युजर नाव विचारते आणि जेव्हा तुम्ही युजर नाव प्रविष्ट केले तेव्हा ते स्क्रीनवर छापले जाते:

**पायथन ३.६**

युजर नाव = इनपुट (“युजर नाव प्रविष्ट करा:”)

प्रिंट (“युजर नाव आहे: ” + युजर नाव)

**रन उदाहरण “**

**पायथन 2.7**

युजर नाव = raw\_input(“युजर नाव प्रविष्ट करा:”)

प्रिंट (“युजर नाव आहे: ” + युजर नाव)

चालवा उदाहरण “

जेव्हा इनपुट() फंक्शन येतो तेव्हा पायथन कार्यान्वित करणे थांबवते आणि यूजर ने काही इनपुट दिल्यावर ते चालू राहते.

## ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग आणि जावा भाषा (Object Oriented Programming and JAVA Language)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- JAVA काय आहे
- OOPs च्या संकल्पना.

### JavaTutorial

#### Java म्हणजे काय

आमचे मुख्य Java प्रोग्रामिंग ट्यूटोरियल विद्यार्थी आणि कार्यरत व्यावसायिकांसाठी डिझाइन केलेले आहे. जावा ही ऑब्जेक्ट-ओरिएंटेड, क्लास-आधारित, समवर्ती, सुरक्षित आणि सामान्य-उद्देशीय संगणक-प्रोग्रामिंग भाषा आहे. हे मोठ्या प्रमाणावर वापरले जाणारे मजबूत तंत्रज्ञान आहे.

जावा ही एक प्रोग्रामिंग भाषा आणि एक व्यासपीठ आहे. जावा ही उच्च स्तरीय, मजबूत, ऑब्जेक्ट ओरिएंटेड आणि सुरक्षित प्रोग्रामिंग भाषा आहे.

जावा सन मायक्रोसिस्टम्सने (जी आता ओरॅकलची सबकंपनी आहे) सन 1995 मध्ये विकसित केली होती. जेम्सगोसलिंग हे जावाचे जनक म्हणून ओळखले जातात. जावापूर्वी त्याचे नाव ओक होते. ओक ही आधीच नोंदणीकृत कंपनी असल्याने, जेम्स गोसलिंगांच्या टीमने ओकचे नाव बदलून जावा केले.

प्लॅटफॉर्म: कोणतेही हार्डवेअर किंवा सॉफ्टवेअर वातावरण ज्यामध्ये प्रोग्राम चालतो, त्याला प्लॅटफॉर्म म्हणून ओळखले जाते. Javahasa runtimeenvironment(JRE) आणिAPI असल्याने, त्याला प्लॅट फॉर्म म्हणतात.

#### Java उदाहरण

जावा प्रोग्रामिंगचे उदाहरण पाहू या. हॅलो Java उदाहरणाचे तपशीलवार डिस्क्रिप्शन पुढील पानावर सबलब्ध आहे.

साधा जावा

```
classSimple{
```

```
    public static void main (String args[])
    {System.out.println("HelloJava");
    }
}
```

#### एप्लीकेशन

सनच्या मते, 3 अब्ज उपकरणे Java चालवतात. अशी अनेक उपकरणे आहेत जिथे जावा सध्या वापरला जातो. त्यापैकी काही एरिया खालीलप्रमाणे आहेत:

- 1 डेस्कटॉप ऍप्लिकेशन्स जसे की ऍक्रोबॅट रीडर, मीडिया प्लेयर, अँटीव्हायरस इ.
- 2 वेब ऍप्लिकेशन्स जसे की irctc.co.in,javatpoint.com, इ.

3 एंटरप्राइझ ऍप्लिकेशन्स जसे की बँकिंग ऍप्लिकेशन्स.

4 मोबाईल

5 एम्बेडेड सिस्टम

6 स्मार्टकार्ड

7 रोबोटिक्स

8 खेळ इ.

#### Java ऍप्लिकेशन्सचे टाइप

जावा प्रोग्रामिंग वापरून 4 टाइपचे ऍप्लिकेशन तयार केले जाऊ शकतात:

- 1 **स्टँडअलोन ऍप्लिकेशन** : स्वतंत्र एप्लिकेशन ना डेस्कटॉप एप्लिकेशन किंवा विंडो-आधारित एप्लिकेशन म्हणून देखील ओळखले जाते. हे पारंपारिक सॉफ्टवेअर आहेत जे आम्हाला प्रत्येक मशीनवर इंस्टाल करणे आवश्यक आहे. स्टँडअलोन ऍप्लिकेशनची उदाहरणे म्हणजे मीडियाप्लेअर, अँटीव्हायरस, इ. स्टँड अँप्लिकेशन्स तयार करण्यासाठी AWT आणि स्विंगरे वापरलेले जावा.
- 2 **वेब ऍप्लिकेशन** : जो ऍप्लिकेशन सर्व्हरच्या बाजूने चालतो आणि डायनॅमिक पेज तयार करतो त्याला वेब ऍप्लिकेशन म्हणतात. सध्या, सर्व्हेट, जेएसपी, स्ट्रट्स, स्पिंग, हायब्रनेट, जेएसएफ, इ. Java मध्ये वेब ऍप्लिकेशन तयार करण्यासाठी तंत्रज्ञानाचा वापर केला जातो.
- 3 **एंटरप्राइझ ऍप्लिकेशन** : बँकिंग ऍप्लिकेशन्स इत्यादी सारख्या निसर्गात वितरीत केलेल्या ऍप्लिकेशनला एंटरप्राइझ ऍप्लिकेशन म्हणतात. यात उच्च-स्तरीय सुरक्षा, लोड बॅलन्सिंग आणि क्लस्टरिंग सारखे फायदे आहेत. Java मध्ये, EJB चा वापर एंटरप्राइझ ऍप्लिकेशन तयार करण्यासाठी केला जातो.
- 4 **मोबाईल ऍप्लिकेशन** : मोबाईल उपकरणांसाठी तयार केलेल्या ऍप्लिकेशनला मोबाईल ऍप्लिकेशन म्हणतात. सध्या, Android आणि Java ME मोबाईल ऍप्लिकेशन तयार करण्यासाठी वापरले जातात.

#### जावा प्लॅटफॉर्म/आवृत्त्या

जावाचे 4 प्लॅटफॉर्म विकृती आहेत:

- 1 **JavaSE (जावा स्टँडर्ड एडिशन)** : हे जावा प्रोग्रामिंग प्लॅटफॉर्म आहे. यात Java प्रोग्रामिंग API जसे की java.lang, java.io, java.net, java.util, java.sql, java.math इत्यादींचा समावेश आहे. यामध्ये

OOPs, String, Regex, अपवाद, इनर क्लासेस, मल्टी थ्रेडिंग सारख्या मुख्य विषयांचा समावेश आहे ,I/OStream, नेटवर्किंग, AWT, स्विंग, रिफ्लेक्शन, कलेक्शन, इ.

**2 JavaEE (Java Enterprise Edition) :** हे एक एंटरप्राइझ प्लॅटफॉर्म आहे जे प्रामुख्याने वेब आणि एंटरप्राइझ ऍप्लिकेशन्स विकसित करण्यासाठी वापरले जाते. हे Java SEplatform च्या शीर्षस्थानी तयार केले आहे. यामध्ये सर्वलेट, JSP, वेब सर्विस , EJB, JPA, इत्यादी विषयांचा समावेश आहे.

**3 JavaME (जावा मायक्रो एडिशन) :** आयटीआय हे एक मायक्रो प्लॅटफॉर्म आहे जे मोबाईल ऍप्लिकेशन्ससाठी समर्पित आहे.

**4 JavaFX :** याचा यूजेस प्रचुर इंटरनेट ऍप्लिकेशन विकसित करण्यासाठी केला जातो. हे हलके वजन युजर इंटरफेस API वापरते.

### पूर्वतयारी

TolearnJava, तुम्हाला C/C++ प्रोग्रामिंग भाषेचे बेसिक ज्ञान असणे आवश्यक आहे.

### प्रेक्षक

आमचे Java प्रोग्रामिंग ट्यूटोरियल नवशिक्या आणि व्यावसायिकांना मदत करण्यासाठी डिझाइन केलेले आहे.

### जावाचा इतिहास

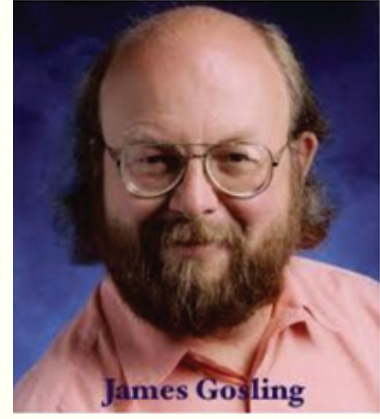
जावाचा इतिहास खूप रंजक आहे. Java मूलतः इंटरएक्टिव्ह टेलिव्हिजनसाठी डिझाइन केले गेले होते, परंतु त्या वेळी डिजिटल केबल टेलिव्हिजन उद्योगासाठी ते खूप एडवॉन्स तंत्रज्ञान होते. जावाचा इतिहास ग्रीन टीमने सुरू होतो. जावा टीमच्या सदस्यांनी (ग्रीन टीम म्हणूनही ओळखले जाते), सेट-टॉप बॉक्स, टेलिव्हिजन इत्यादीसारख्या डिजिटल उपकरणांसाठी भाषा विकसित करण्यासाठी हा प्रकल्प सुरू केला. तथापि, ते इंटरनेट प्रोग्रामिंगसाठी सर्वात अनुकूल होते. नंतर, जावा तंत्रज्ञान नेटस्केपने कॉर्पोरेट केले .

Java प्रोग्रामिंग तयार करण्यासाठीची तत्त्वे होती "साधे, मजबूत, पोर्टेबल, प्लॅटफॉर्म-स्वतंत्र, सुरक्षित, उच्च कार्यप्रदर्शन, मल्टीथ्रेडेड, आर्किटेक्चर न्यूट्रल, ऑब्जेक्ट ओरिएंटेड, इंटरप्रीटेड आणि डायनॅमिक". जावा 1995 मध्ये जेम्स गॉसलिंग, ज्यांना जावाचे जनक म्हणून ओळखले जाते, यांनी विकसित केले होते. जेम्स गॉसलिंग आणि त्यांच्या टीम सदस्यांनी 90 च्या दशकाच्या सुरुवातीला हा प्रकल्प सुरू केला.

सध्या, जावाचा वापर इंटरनेट प्रोग्रामिंग, मोबाईल उपकरणे, गेम्स, ई-बिझनेस सोल्यूशन्स इ. मध्ये केला जातो. जावाच्या इतिहासाचे डिस्क्रिप्शन करणारे महत्त्वाचे मुद्दे खाली दिले आहेत.

### जेम्स गोस्लिंग - जावाचे संस्थापक

1 जेम्स गोस्लिंग, माईक शेरिडन आणि पॅट्रिक नॉटन यांनी जून 1991 मध्ये जावा भाषा प्रकल्प सुरू केला. सन इंजिनियर्सची छोटी टीम ग्रीनटीम म्हणतात.



- 2 सुरुवातीला हे सेट-टॉप बॉक्ससारख्या इलेक्ट्रॉनिक उपकरणांमध्ये लहान, एम्बेडेड सिस्टमसाठी डिझाइन केले होते.
- 3 प्रथम, जेम्स गॉसलिंगने याला "Greentalk" म्हटले आणि फाइल एक्सपान्शन होता.gt.
- 4 त्यानंतर, त्याला ओक म्हटले गेले आणि ग्रीन प्रकल्पाचा एक पार्ट म्हणून विकसित केले गेले.जावाचे नाव "ओक" का ठेवले गेले? ओक ते जावा असा जावा इतिहास
- 5 ओक का? ओक हे शक्तीचे सिम्बॉल्स आहे आणि यू.एस.ए., फ्रान्स, जर्मनी, रोमानिया इत्यादी अनेक देशांचे राष्ट्रीय वृक्ष म्हणून निवडले जाते.
- 6 1995 मध्ये, ओकचे "जावा" असे नामकरण करण्यात आले कारण ते आधीपासूनच ओक टेक्नॉलॉजीजचे ट्रेडमार्क होते.

### Java Programming ला "Java" का नाव दिले?

- 7 त्यांनी जावा भाषेसाठी Java हे नाव का निवडले? नवीन नाव निवडण्यासाठी संघ जमला. सुचविलेले शब्द "गतिशील", "क्रांतिकारक", "सिल्क", "जोल्ड", "DNA", इत्यादी होते. तंत्रज्ञानाचे सार प्रतिबिंबित करणारे काहीतरी हवे होते: क्रांतिकारी, गतिमान, चैतन्यशील, मस्त, अद्वितीय आणि शब्दलेखन करणे सोपे, आणि funtosay. जेम्स गॉसलिंगच्या मते, "सिल्क सोबत जावा ही टॉप चॉईसपैकी एक होती". जावा खूप अनोखा असल्याने, बहुतेक टीम सदस्यांनी इतर नावांपेक्षा जावाला प्राधान्य दिले.
- 8 जावा हे इंडोनेशियातील एक बेट आहे जिथे पहिली कॉफी तयार झाली (ज्याला जावा कॉफी म्हणतात). हे एक टाइप चे एस्प्रेसोबीन आहे. जावाचे नाव जेम्स गॉसलिंगने त्याच्या कार्यालयाजवळ एक कप कॉफी घेत असताना.
- 9 लक्षात घ्या की जावा हे फक्त एक नाव आहे, संक्षिप्त रूप नाही.
- 10 सुरुवातीला जेम्स गॉसलिंगने सन मायक्रो सिस्टीम्स (जी आता ओरॅकल कॉर्पोरेशनची सबकंपनी आहे) येथे विकसित केली आणि 1995 मध्ये रिलीज केली.
- 11 1995 मध्ये, टाइममॅगझिनने 1995 च्या दहा सर्वोत्तम उत्पादनांपैकी जावोन म्हटले.

12 JDK 1.0 23 जानेवारी 1996 रोजी रिलीझ करण्यात आले. Java च्या पहिल्या रिलीझनंतर, भाषेमध्ये अनेक अतिरिक्त वैशिष्ट्ये जोडण्यात आली आहेत. आता जावा विंडोज ऍप्लिकेशन्स, वेब ऍप्लिकेशन्स, एंटरप्राइझ ऍप्लिकेशन्स, मोबाईल ऍप्लिकेशन्स, कार्ड्स इत्यादींमध्ये वापरला जात आहे. प्रत्येक नवीन आवृत्ती जावामध्ये नवीन वैशिष्ट्ये जोडते.

### जावा आवृत्ती इतिहास

manyjavaversionshavebeenreleased.tever Java ची करंट स्थिर रिलीझ JavaSE18 आहे.

- 1 JDK Alpha and Beta (1995)
- 2 JDK1.0 (23rdJan1996)
- 3 JDK1.1(19thFeb1997)
- 4 J2SE1.2 (8thDec1998)
- 5 J2SE1.3 (8thMay2000)
- 6 J2SE1.4(6thFeb2002)
- 7 J2SE5.0(30thSep2004)
- 8 JavaSE6(11thDec2006)
- 9 JavaSE7 (28 जुलै 2011)
- 10 JavaSE8 (18 मार्च 2014)
- 11 JavaSE 9 (21stSep2017)
- 12 JavaSE 10 (20thMar2018)
- 13 JavaSE11 (September2018)
- 14 JavaSE12(March2019)
- 15 JavaSE13(September2019)
- 16 JavaSE14(Mar2020)
- 17 JavaSE15(September2020)
- 18 JavaSE16(Mar2021)
- 19 JavaSE17(September 2021)
- 20 JavaSE 18 (tobe released by March2022)

Java SE 8 रिलीझ झाल्यापासून, ओरॅकल कॉर्पोरेशन एक पॅटर्न फॉलो करते ज्यामध्ये प्रत्येक सम आवृत्ती मार्चमंथंडनमध्ये रिलीज होते आणि सप्टेंबर महिन्यात रिलीज होते.

### Java ची वैशिष्ट्ये

जावा प्रोग्रामिंग भाषा निर्मितीचा मुख्य उद्देश ती पोर्टेबल, सोपी आणि सुरक्षित प्रोग्रामिंग भाषा बनवणे हा होता. याशिवाय, काही उत्कृष्ट वैशिष्ट्ये देखील आहेत जी या भाषेच्या लोकप्रियतेमध्ये महत्त्वपूर्ण भूमिका बजावतात. Java ची वैशिष्ट्ये Java buzzwords म्हणूनही ओळखली जातात.

जावा भाषेतील सर्वात महत्त्वाच्या वैशिष्ट्यांची यादी खाली दिली आहे.

- 1 साधे
- 2 ऑब्जेक्ट-ओरिएंटेड
- 3 पोर्टेबल
- 4 प्लॅटफॉर्म स्वतंत्र
- 5 सुरक्षित
- 6 मजबूत
- 7 आर्किटेक्चर तटस्थ
- 8 अर्थ लावला
- 9 उच्च कार्यक्षमता
- 10 मल्टीथ्रेडेड
- 11 वितरीत केले
- 12 डायनॅमिक



### Simple

Java शिकायला खूप सोपे आहे, आणि त्याची वाक्यरचना सोपी, स्वच्छ आणि समजण्यास सोपी आहे. सन मायक्रोसिस्टमच्या मते, जावा भाषा ही एक सोपी प्रोग्रामिंग भाषा आहे कारण:

- Java वाक्यरचना C++ वर आधारित आहे (प्रोग्रामर्सना C++ नंतर ते शिकणे सोपे आहे).
- Java ने अनेक क्लिष्ट आणि क्वचित वापरलेली वैशिष्ट्ये काढून टाकली आहेत, उदाहरणार्थ, स्पष्ट पॉइंटर्स, ऑपरेटर ओव्हरलोडिंग इ.
- रेफरेन्स नसलेल्या वस्तू काढून टाकण्याची गरज नाही कारण Java मध्ये ऑटोमॅटिक कचरा संकलन आहे.

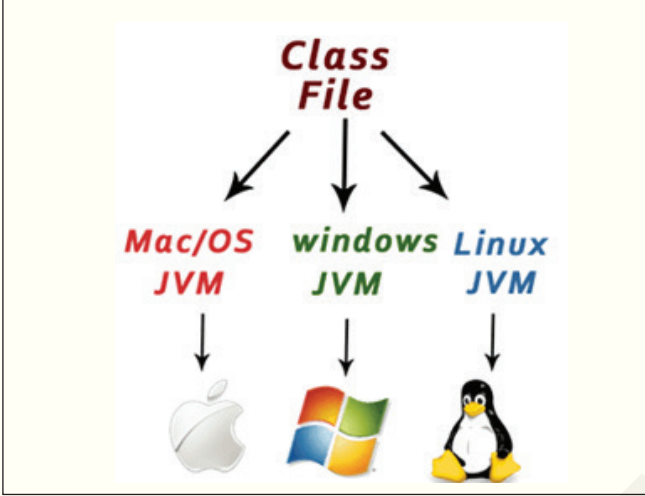
### ऑब्जेक्ट-ओरिएंटेड

जावा ही ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग भाषा आहे. Java मधील प्रत्येक गोष्ट एक वस्तू आहे. ऑब्जेक्ट-ओरिएंटेड म्हणजे आम्ही आमचे सॉफ्टवेअर विविध टाइप च्या ऑब्जेक्ट्सचे संयोजन म्हणून मॅनेज करतो ज्यामध्ये डेटा आणि वर्तन दोन्ही समाविष्ट असतात.

ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग (OOPs) ही एक मेथड्स आहे जी काही नियम प्रदान करून सॉफ्टवेअर विकास आणि देखभाल सुलभ करते.

OOPs च्या बेसिक संकल्पना आहेत:

- 1 Object
- 2 Class
- 3 Inheritance
- 4 Polymorphism
- 5 Abstraction
- 6 Encapsulation



1 रनटाइम वातावरण

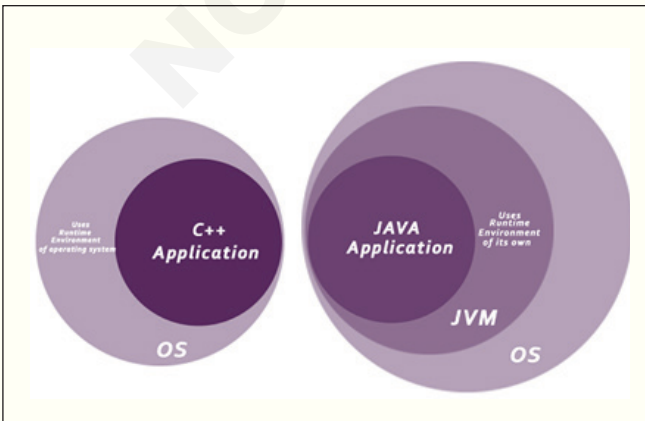
2 API (अॅप्लिकेशन प्रोग्रामिंग इंटरफेस)

जावा कोड मल्टिपल प्लॅटफॉर्मवर कार्यान्वित केला जाऊ शकतो, उदाहरणार्थ, विंडोज, लिनक्स, सन सोलारिस, मॅक/ओएस, इ. जावा कोड कंपाइलरद्वारे संकलित केला जातो आणि बायकोडमध्ये रूपांतरित केला जातो. हा बाइटकोड एक प्लॅटफॉर्म-स्वतंत्र कोड आहे कारण तो मल्टिपल प्लॅटफॉर्मवर चालवला जाऊ शकतो, म्हणजे एकदा लिहा आणि कुठेही चालवा (WORA).

## सुरक्षित

जावा त्याच्या सुरक्षिततेसाठी प्रसिद्ध आहे. जावा सह, आम्ही व्हायरस-ओपन सिस्टिम विकसित करू शकतो. जावा सुरक्षित आहे कारण:

- कोणतेही स्पष्ट सूचक नाही
- जावा प्रोग्राम वर्च्युअल मशीन सँडबॉक्समध्ये चालतात



- **क्लासलोडर:** जावा मधील क्लासलोडर हा जावा Runtime Environment (JRE) चा एक पार्ट आहे जो जावा क्लासेस जावा वर्च्युअल मशीनमध्ये गतिशीलपणे लोड करण्यासाठी वापरला जातो. हे नेटवर्क स्त्रोतांकडून आयात केलेल्या लोकल फाइल सिस्टमच्या वर्गासाठी पॅकेज वेगळे करून सुरक्षा जोडते.
- **बाइटकोड सत्यापनकर्ता:** हे बेकायदेशीर कोडसाठी कोडचे तुकडे तपासते जे ऑब्जेक्ट्सच्या एक्सेस अधिकारांचे उल्लंघन करू शकतात.
- **सुरक्षा व्यवस्थापक:** लोकल डिस्कवर वाचन आणि लेखन यांसारख्या वर्गाला कोणती रिसोर्सेस एक्सेस करता येईल हे ते ठरवते.

जावा भाषा ही सिव्क्युरिटीज बाय डीफॉल्ट पुरवते. SSL, JAAS, क्रिप्टोग्राफी इ. द्वारे काही सुरक्षा देखील एप्लिकेशन विकासकाद्वारे स्पष्टपणे प्रदान केली जाऊ शकते.

## मजबूत

रोबस्टची इंग्रजी खाण मजबूत आहे. जावा मजबूत आहे कारण:

- हे मजबूत मेमरी मॅनेजमेन्ट वापरते.
- सुरक्षेच्या समस्या टाळणाऱ्या पॉइंटरचा अभाव आहे.
- जावा ऑटोमॅटिक कचरा संकलन प्रदान करते जे जावा अनुप्रयोगाद्वारे वापरल्या जात नसलेल्या वस्तूंपासून ओपन होण्यासाठी जावा वर्च्युअल मशीनवर चालते.
- जावा मध्ये अपवाद हाताळणी आणि टाइप तपासण्याची यंत्रणा आहे. हे सर्व मुद्दे जावा मजबूत करतात.

## आर्किटेक्चर-तटस्थ

जावा आर्किटेक्चर तटस्थ आहे कारण तेथे अंमलबजावणीवर अवलंबून वैशिष्ट्ये नाहीत, उदाहरणार्थ, Primitive प्रकारांचा आकार निश्चित आहे.

सी प्रोग्रामिंगमध्ये, इंट डेटा टाइप 32-बिट आर्किटेक्चरसाठी 2 बाइट्स मेमरी आणि 64-बिट आर्किटेक्चरसाठी 4 बाइट्स मेमरी व्यापतो. तथापि, ते जावा मधील 32 आणि 64-बिट आर्किटेक्चर्ससाठी 4 बाइट्स मेमरी व्यापते.

## पोर्टेबल

जावा पोर्टेबल आहे कारण ते तुम्हाला जावा bytecode कोणत्याही प्लॅटफॉर्मवर घेऊन जाण्याची सुविधा देते. त्याला कोणत्याही अंमलबजावणीची आवश्यकता नाही.

## उच्च कार्यक्षमता

जावा इतर पारंपारिक व्याख्या केलेल्या प्रोग्रामिंग भाषांपेक्षा वेगवान आहे कारण जावा bytecode नेटिव्ह कोडच्या "जवळ" आहे. संकलित केलेल्या भाषेपेक्षा ती अजून थोडी हळू आहे (उदा. C++). जावा ही एक व्याख्या केलेली भाषा आहे म्हणूनच ती संकलित भाषांपेक्षा हळू आहे, उदा., C, C++, इ.

## वितरित केले

जावा वितरित केले जाते कारण ते वापरकर्त्यांना जावा मध्ये वितरित एप्लिकेशन तयार करण्यास सुलभ करते. वितरित एप्लिकेशन तयार

करण्यासाठी RMI आणि EJB वापरले जातात. Java चे हे वैशिष्ट्य आम्हाला इंटरनेटवरील कोणत्याही मशीनवरून कॉल करून फायलींमध्ये एक्सेस करण्यास सक्षम करते.

### मल्टी-थ्रेडेड

थ्रेड हा एका वेगळ्या प्रोग्रामसारखा असतो, जो एकाच वेळी कार्यान्वित होतो. आपण जावा प्रोग्राम लिहू शकतो जे एकाच वेळी अनेक थ्रेड्स परिभाषित करून अनेक फंक्शन हाताळतात. मल्टी-थ्रेडिंगचा मुख्य फायदा म्हणजे तो प्रत्येक थ्रेडसाठी मेमरी व्यापत नाही. हे एक सामान्य मेमरी एरिया सामायिक करते. मल्टी-मीडिया, वेब ऍप्लिकेशन्स इत्यादींसाठी थ्रेड महत्त्वाचे आहेत.

### डायनामिक

जावा ही डायनॅमिक भाषा आहे. हे वर्गांच्या डायनॅमिक लोडिंगला सपोर्ट देते. याचा अर्थ क्लास मागणीनुसार लोड केले जातात. हे त्याच्या मूळ भाषांमधील फंक्शन्सचे सपोर्ट करते, म्हणजे C आणि C++.

जावा डायनॅमिक संकलन आणि ऑटोमॅटिक मेमरी मॅनेजमेंट (कचरा संकलन) चे सपोर्ट करते

© NIMI  
NOT TO BE REPUBLISHED

**JAVA प्रोग्राम लिहिण्याचे प्रात्यक्षिक दाखवा (Demonstrate writing- Java programs)**

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- java व्हर्च्युअल मशीन
- JVM चे इंटरनल आर्किटेक्चर.

**JVM (जावा व्हर्च्युअल मशीन) आर्किटेक्चर**

JVM (Java Virtual Machine) एक ऑब्स्ट्रक्ट मशीन आहे. हे एक स्पेसिफिकेशन आहे जे रनटाइम वातावरण प्रदान करते ज्यामध्ये java bytecode कार्यान्वित केले जाऊ शकते.

JVM अनेक हार्डवेअर आणि सॉफ्टवेअर प्लॅटफॉर्मसाठी सबलम्ब आहेत (म्हणजे JVM हे प्लॅटफॉर्मवर अवलंबून आहे).

**जेव्हीएम म्हणजे काय**

हे आहे:

- 1 एक तपशील जावा व्हर्च्युअल मशीनचे काम जेथे स्पेसिफाइड केले आहे. परंतु अंमलबजावणी प्रदाता स्वतंत्र आहे अल्गोरिदम निवडण्यासाठी. त्याची अंमलबजावणी झाली आहे ओरॅकल आणि इतर कंपन्यांनी प्रदान केले.
- 2 एक अंमलबजावणीत्याची अंमलबजावणी JRE (Java Runtime Environment) म्हणून ओळखली जाते.
- 3 रनटाइम उदाहरणजेव्हा तुम्ही java क्लास चालवण्यासाठी कमांड प्रॉम्प्टवर java कमांड लिहा, तेव्हा JVM चे एक उदाहरण तयार केले जाईल.

**ते काय करते****JVM खालील ऑपरेशन करते:**

- कोड लोड करतो
- कोड सत्यापित करते
- कोड कार्यान्वित करतो
- रनटाइम वातावरण प्रदान करते

**JVM यासाठी व्याख्या प्रदान करते:**

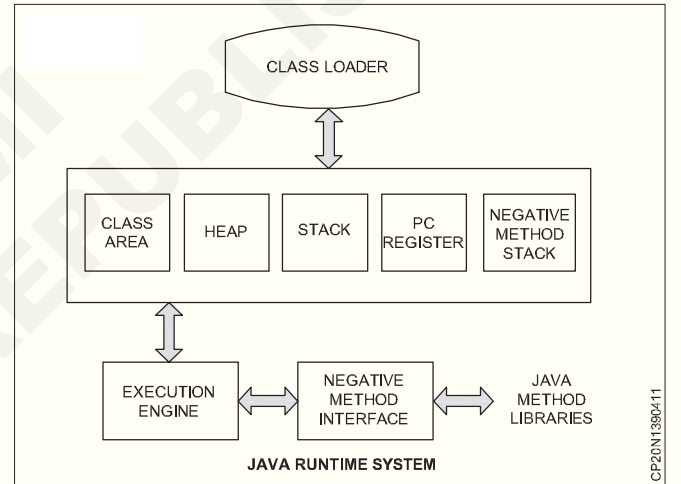
- मेमरी एरिया
- क्लास फाइल फॉर्मॅट
- नोंदणी संच
- कचऱ्याचा ढीग
- घातक त्रुटी रिपोर्ट इ.

**जेव्हीएम आर्किटेक्चर**

JVM ची इंटरनल रचना समजून घेऊ. यात क्लासलोडर, मेमरी एरिया, एक्झिक्युशन इंजिन इ.

**1 क्लासलोडर**

क्लासलोडर हे JVM चे सबसिस्टिम आहे जे क्लास फाईल्स लोड करण्यासाठी वापरले जाते. जेव्हाही आपण जावा प्रोग्राम रन करतो तेव्हा तो क्लासलोडरद्वारे प्रथम लोड केला जातो. Java मध्ये तीन अंगभूत क्लासलोडर आहेत.



- 1 **बूटस्ट्रॅप क्लासलोडर:** हा पहिला क्लासलोडर आहे जो एक्स्टेंशन क्लासलोडरचा सुपर क्लास आहे. हे rt.jar फाइल लोड करते ज्यामध्ये Java स्टॅण्डर्ड आवृत्तीच्या सर्व क्लास फाइल्स असतात जसे की java.lang पॅकेज क्लासेस, java.net पॅकेज क्लासेस, java.util पॅकेज क्लासेस, java.io पॅकेज क्लासेस, java.sql पॅकेज क्लासेस इ.
- 2 **एक्स्टेंशन क्लासलोडर:** हा बूटस्ट्रॅपचा चाइल्ड क्लासलोडर आणि सिस्टम क्लासलोडरचा पॅरेंट क्लासलोडर आहे. हे \$JAVA\_HOME/jre/lib/ext डिरेक्टरीमध्ये असलेल्या जार फाइल्स लोड करते.
- 3 **सिस्टम/अॅप्लिकेशन क्लासलोडर:** हा एक्स्टेंशन क्लासलोडरचा चाइल्ड क्लासलोडर आहे. हे क्लासपाथवरून क्लासफाईल्स लोड करते. डीफॉल्टनुसार, क्लासपाथ चालू डिरेक्टरी वर सेट केला जातो. तुम्ही "-cp" किंवा "-classpath" स्विच वापरून क्लासपाथ बदलू शकता. हे ऍप्लिकेशन क्लासलोडर म्हणून देखील ओळखले जाते.

1 //Let's see an example to print the classloader name

```

2 public class ClassLoaderExample
3 {
4     public static void main(String[] args)
5     {
6         // Let's print the classloader name of current class.
7         //Application/System classloader will load this class
8         Class c=ClassLoaderExample.class;
9         System.out.println(c.getClassLoader());
10        //If we print the classloader name of String, it will print null
            because it is an
11        //in-built class which is found in rt.jar, so it is loaded by
            Bootstrap classloader
12        System.out.println (String.class.get ClassLoader());
13    }
14 }

```

### आउटपुट:

हे Java द्वारे प्रदान केलेले इंटरनल क्लासलोडर्स आहेत. तुम्हाला तुमचा स्वतःचा क्लासलोडर तयार करायचा असल्यास, तुम्हाला ClassLoader क्लास वाढवावा लागेल.

### Class (मेथड्स) एरिया

क्लास (मेथड्स) एरिया प्रति-क्लास स्ट्रक्चर जसे की रनटाइम कॉन्स्टंट पूल, फील्ड आणि मेथड्स डेटा, पद्धतींसाठी कोड स्टोर करते.

### Heap

हे रनटाइम डेटा एरिया आहे ज्यामध्ये ऑब्जेक्ट्सचे वाटप केले जाते.

### स्टॅक

Java Stack फ्रेम्स साठवतो. यात लोकल व्हेरिएबल्स आणि आंशिक रिझल्ट आहेत आणि मेथड्स आवाहन आणि रिटर्नमध्ये भूमिका बजावते.

प्रत्येक थ्रेडमध्ये प्राइवेट JVM स्टॅक असतो, थ्रेड प्रमाणेच तयार केला जातो.

प्रत्येक वेळी एखादी मेथड्स लागू केल्यावर एक नवीन फ्रेम तयार केली जाते. जेव्हा फ्रेमची मेथड्स पूर्ण होते तेव्हा ती नष्ट होते.

### प्रोग्रॅम काउंटर रजिस्टर

PC (प्रोग्राम काउंटर) रजिस्टरमध्ये सध्या कार्यान्वित होत असलेल्या Java व्हर्च्युअल मशीन सूचनांचा एड्रेस असतो.

### मूळ मेथड्स स्टॅक

यात ऍप्लिकेशनमध्ये वापरल्या जाणाऱ्या सर्व मूळ पद्धतींचा समावेश आहे.

## एक्सक्युशन इंजिन

### त्यात समाविष्ट आहे:

- 1 आभासी प्रोसेसर
- 2 **दुभाषी:** बाइटकोड प्रवाह वाचा नंतर सूचना कार्यान्वित करा.
- 3 **जस्ट-इन-टाइम (जेआयटी) कंपाइलर:** हे कार्यप्रदर्शन सुधारण्यासाठी वापरले जाते. JIT बाइट कोडचे पार्ट संकलित करते ज्यात एकाच वेळी समान कार्यक्षमता असते आणि त्यामुळे संकलनासाठी लागणारा वेळ कमी होतो. येथे, "कंपाइलर" हा शब्द Java व्हर्च्युअल मशीन (JVM) च्या इंस्ट्रक्शन सेटपासून स्पेसिफिक CPU च्या इंस्ट्रक्शन सेटपर्यंत अनुवादकाला सूचित करतो.

**जावा नेटिव्ह इंटरफेस:** Java नेटिव्ह इंटरफेस (JNI) हे एक फ्रेमवर्क आहे जे C, C++, असेंबली इत्यादी सारख्या दुसऱ्या भाषेत लिहिलेल्या दुसऱ्या अनुप्रयोगाशी संवाद साधण्यासाठी इंटरफेस प्रदान करते. Java कन्सोलला आउटपुट पाठवण्यासाठी किंवा OS लायब्ररीशी संवाद साधण्यासाठी JNI फ्रेमवर्क वापरते.

### जावा मध्ये बाइटकोड म्हणजे काय

#### आढावा

Java मधील बाइटकोड हा Java व्हर्च्युअल मशीनसाठी सूचनांचा संच आहे जो त्यांचा अर्थ लावण्यासाठी जबाबदार आहे. Java प्रोग्राम संकलित केल्यावर, Java bytecode व्युत्पन्न केला जातो जो JVM वापरून कोणत्याही प्लॅटफॉर्मवर (प्लॅटफॉर्मवर अवलंबून) कार्यान्वित केला जाऊ शकतो. ते .class फाईलमध्ये असते.

#### व्याप्ती

या लेखाचा उद्देश आहे:

- जावा मधील बाइटकोडची संकल्पना स्पष्ट करा.
- चे फायदे आणि तोटे समजण्यास मदत करा Java मध्ये bytecode.

### Java मध्ये Bytecode म्हणजे काय?

तुम्ही अशा जगाची कल्पना करू शकता जिथे आम्हाला ऑपरेटिंग सिस्टम, प्रोसेसर आर्किटेक्चर इ. सारख्या वेगवेगळ्या वैशिष्ट्यांसह प्रत्येक डिव्हाइससाठी अनेक वेळा कोड लिहावा लागेल? हे होईल

एक त्रासदायक प्रक्रिया व्हा. या गोष्टी लक्षात घेऊन, जावा डेव्हलपर्सनी राईट वन्स रीड एनीव्हेअर ही संकल्पना आणली, जी जावा भाषेत एक वैशिष्ट्य म्हणून समाविष्ट झाली. हे साध्य करण्यासाठी, आमचा जावा सोर्स कोड संकलित केल्यावर बाइटकोड म्हणून ओळखला जाणारा कोड तयार होतो. आता आपण त्यावर चर्चा करू.

Java मधील Bytecode हा Java Virtual Machine साठी सूचनांचा संच आहे. Java व्हर्च्युअल मशीन, JVM म्हणून संक्षिप्त, संगणकाला Java मध्ये लिहिलेला कोड चालविण्यास सक्षम करते. जेव्हा जावा प्रोग्राम संकलित केला जातो तेव्हा बायकोड तयार होतो. हे C++ मधील असेंबलरच्या समतुल्य आहे.



बाइटकोड हा प्लॅटफॉर्म-स्वतंत्र सूचनांचा संच आहे कारण त्याचा अर्थ JVM द्वारे केला जातो आणि अंमलात आणला जातो. कोणत्याही बदलाशिवाय मल्टिपल संगणक आर्किटेक्चरवर चालणाऱ्या कोडला मशीन किंवा प्लॅटफॉर्म स्वतंत्र कोड म्हणतात.

बाइटकोड निम्न-स्तरीय आणि उच्च-स्तरीय सूचनांच्या संचामध्ये आहे. आम्ही आमचा कोड उच्च-स्तरीय भाषेत लिहितो, तो बायटेकोडमध्ये संकलित केला जातो आणि नंतर, JVM त्याचा मशीन कोडमध्ये अर्थ लावतो, एक निम्न-स्तरीय सूचनांचा संच अंमलात आणण्यासाठी तयार असतो. म्हणून, आम्ही बाइटकोडला निम्न-स्तरीय आणि उच्च-स्तरीय भाषेमधील कोड म्हणतो.

### वस्तुस्थिती:

जावा ही एक संकलित आणि व्याख्या केलेली भाषा आहे, इतर भाषांपेक्षा वेगळी, जी एकतर संकलित किंवा व्याख्या केली जाते.

### हे कस काम करत?

वरील प्रक्रिया स्पष्टीकरणासह समजून घेऊया:

- आपण Java मध्ये जो कोड लिहितो त्याला सोर्स कोड म्हणतात, जो उच्च-स्तरीय भाषेत लिहिला जातो. उच्च-स्तरीय भाषा ही एक प्रोग्रामर-अनुकूल भाषा आहे ज्यामध्ये स्टेटमेंट इंग्रजीमध्ये लिहिलेली आहेत आणि ती मानवी भाषांच्या जवळ आहे. Java फाइलचा एक्सपान्शन ".java" आहे.
- जेव्हा आपण प्रोग्राम संकलित करतो, तेव्हा कंपायलर ".java" फाइल संकलित करतो आणि ".class" फाइल तयार करतो. त्यात बाइटकोड आहे.
- बायटेकोड आम्हाला इतर कोणत्याही प्लॅटफॉर्मवर ".class" फाइल चालवण्याची परवानगी देतो.
- परंतु या बायकोडला ते कार्यान्वित करण्यासाठी दुभाष्याची आवश्यकता आहे. येथे JVM चित्रात येते. JVM कडे दुभाषी आहे. ते कोड तुकडा तुकड्याने कार्यान्वित करते, म्हणजे, एका वेळी एक स्टेटमेंट, जोपर्यंत त्याला त्रुटी सापडत नाही किंवा कोडच्या शेवटी कार्यान्वित केले जात नाही.

म्हणून, बायटेकोड निम्न-स्तरीय भाषा आणि उच्च-स्तरीय भाषा यांच्यामध्ये असल्याचे म्हटले जाते. एक उदाहरण पाहू

समजा तुम्ही कॅल्क्युलेटर ॲपसाठी Java मध्ये कोड लिहिला आहे. अशा प्रकारे आपण त्यावर प्रक्रिया कशी होते हे सहज समजू शकतो.

### फायदे

#### प्लॅटफॉर्म स्वातंत्र्य

जावा प्रकल्प कोणत्याही बदलाशिवाय कोणत्याही डिव्हाइसवर समान कोड चालवण्याच्या उद्देशाने सुरू करण्यात आला. Java मधील बाइटकोडची ही संकल्पना हे उद्दिष्ट साध्य करण्यात मदत करते. बाइटकोड वेगवेगळ्या प्रणालींसाठी भिन्न असू शकतात, परंतु ते सर्व प्रणालींवर JVM द्वारे सारखेच अर्थ लावले जातात आणि कार्यान्वित केले जातात.

**बाइटकोड हा मूलतः** जावा वर्च्युअल मशीनवर चालणारा मशीन-स्तरीय कोड आहे. जेव्हा जेव्हा एखादा क्लास लोड केला जातो तेव्हा त्याला प्रति क्लास मेथड्स नुसार बाइटकोडचा प्रवाह मिळतो. जेव्हा जेव्हा प्रोग्रामच्या अंमलबजावणीदरम्यान ती मेथड्स कॉल केली जाते तेव्हा त्या मेथड्स साठी बायकोड मागवला जातो. Javac केवळ प्रोग्रॅम संकलित करत नाही तर प्रोग्रॅमसाठी बायकोड तयार करते. अशाप्रकारे, आमच्या लक्षात आले आहे की बायकोड अंमलबजावणी Java ला प्लॅटफॉर्म-स्वतंत्र भाषा बनवते.

### पोर्टेबिलिटी

हे पोर्टेबिलिटी जोडण्यास मदत करते जे सुनिश्चित करते की कोड विविध उपकरणांवर चालू शकतो. हे Java द्वारे अंमलात आणलेल्या "एकदा लिहा, कुठेही धावा" या उक्तीशी सुसंगत आहे. आम्हाला इतर कोणत्याही डिव्हाइसवर कोड पुन्हा लिहिण्याची गरज नाही.

तोटे

- बायटेकोड इंटरप्रीटर किंवा JVM शिवाय चालू शकत नाही. कोणत्याही डिव्हाइसमध्ये JVM नसल्यास, त्या डिव्हाइसवर बायटेकोड चालणार नाही.
- बायकोडचे विश्लेषण करणे कठीण आहे कारण ते बायनरी स्वरूपात आहे आणि मानवांना समजू शकत नाही.

### निष्कर्ष

- Java मधील बाइटकोड हा Java आभासी मशीनसाठी सूचनांचा संच आहे.
- बाइटकोड हा प्लॅटफॉर्म-स्वतंत्र कोड आहे.
- बाइटकोड हा एक कोड आहे जो निम्न-स्तरीय भाषा आणि उच्च-स्तरीय भाषा यांच्यामध्ये असतो.
- Java कोड संकलित केल्यानंतर, बायटेकोड तयार होतो, जो JVM वापरून कोणत्याही मशीनवर कार्यान्वित केला जाऊ शकतो.
- जेव्हा आपण प्रोग्राम संकलित करतो, तेव्हा कंपायलर ".java" फाइल संकलित करतो आणि ".class" फाइल तयार करतो.
- बायकोडला चालवण्यासाठी दुभाष्याची आवश्यकता आहे; म्हणून, JVM दुभाषी म्हणून काम करते.
- Java संकलित आणि व्याख्या दोन्ही आहे.

### Java मध्ये क्लासपाथ कसा सेट करायचा?

- वाचा
- चर्चा करा
- सराव
- व्हिडिओ
- अभ्यासक्रम

CLASSPATH त्या स्थानाचे डिस्क्रिप्शन करते जेथे सर्व आवश्यक फाईल्स सबलब्ध आहेत ज्या ऍप्लिकेशनमध्ये वापरल्या जातात. Java Compiler आणि JVM (Java Virtual Machine) आवश्यक फाईल्स सर्च प्यासाठी

CLASSPATH वापरतात. CLASSPATH सेट न केल्यास, Java Compiler आवश्यक फाईल्स शोधू शकणार नाही आणि त्यामुळे खालील त्रुटी टाकेल.

त्रुटी: मुख्य क्लास <क्लास नाव> शोधू किंवा लोड करू शकलो नाही (उदा. GFG) CLASSPATH सेट केल्यावर वरील त्रुटीचे निराकरण केले जाते.

## जावा

```
// If the following code is run when the CLASSPATH is not
// set, it will throw the above error.
// If it is set, we get the desired result
import java.io.*;
class GFG {
public static void main(String[] args)
{
// prints GeeksForGeeks to the console
System.out.println("GeekForGeeks!");
}
}
```

आउटपुट

GeekForGeeks!

Windows मध्ये JAVA मध्ये CLASSPATH सेट करा

कमांड प्रॉम्प्ट:

PATH=.;C:\Program Files\Java\JDK1.6.20\bin सेट करा

**नोंद:** सेमी-कोलन (;) हे विभाजक म्हणून वापरले जाते आणि वरील कमांडमध्ये डॉट (.) हे CLASSPATH चे डिफॉल्ट व्हॅल्यू आहे.

## GUI:

1 स्टार्ट निवडा

2 कंट्रोल पॅनेलवर जा

पिक्चर

3 सिस्टिम आणि सुरक्षा निवडा

पिक्चर

4 एडवॉन्स सिस्टिम सेटिंग्ज निवडा

पिक्चर

5 Environment Variables वर क्लिक करा

पिक्चर

6 सिस्टिम व्हेरिएबल्स इंटरनल नवीन वर क्लिक करा

पिक्चर

7 व्हेरिएबल नाव म्हणून CLASSPATH आणि व्हेरिएबल व्हॅल्यू म्हणून फाइल्सचा पाथ अड करा. पिक्चर

ओके निवडा.

## लिनक्सवर CLASSPATH सेट करा

### कमांड लाइन:

तुम्ही Java कुठे इन्स्टॉल केले आहे ते शोधा, मुळात, ते /usr/lib/jvm पाथ मध्ये आहे. वापरून /etc/environment मध्ये CLASSPATH सेट करा

```
sudo <एडिटर नाव> /etc/environment
```

खालील ओळी अड करा,

```
JAVA_HOME = "/usr/lib/jvm/<java फोल्डर (उदा. java-1.8.0-openjdk-amd64>)/bin" JAVA_HOME निर्यात करा
```

```
CLASSPATH=".:usr/lib/jvm/<java folder>/lib:/home/name/Desktop" CLASSPATH निर्यात
```

**नोंद:** Colon (:) ही वेगळी डिरेक्टरी म्हणून वापरली जाते आणि वरील कमांडमध्ये डॉट (.) हे CLASSPATH चे डिफॉल्ट व्हॅल्यू आहे.

करंट CLASSPATH तपासण्यासाठी, चालवा

```
echo ${CLASSPATH}
```

## कीवर्ड

### Java Keywords

जावा कीवर्ड हे आरक्षित शब्द म्हणूनही ओळखले जातात. कीवर्ड हे स्पेसिफिक शब्द आहेत जे कोडची किल्ली म्हणून काम करतात. हे जावासोबतही पूर्वपरिभाषित शब्द बदलता येणारे किंवा ऑब्जेक्टनाम किंवा वर्गाचे नाव म्हणून वापरले जाऊ शकत नाहीत.

### Java कीवर्डची यादी

खाली दिलेले जावाकीवर्ड किंवा आरक्षित शब्दांची यादी:

- गोपवारा:** ऑब्स्ट्रक्ट क्लास घोषित करण्यासाठी Java abstract कीवर्ड वापरला जातो. एक ऑब्स्ट्रक्ट क्लास इंटरफेसची अंमलबजावणी प्रदान करू शकतो. त्यात ऑब्स्ट्रक्ट आणि ऑब्स्ट्रक्ट मेथड्स असू शकतात.
- बुलियन:** जावा बुलियन कीवर्डचा वापर बुलियन टाइप म्हणून व्हेरिएबल घोषित करण्यासाठी केला जातो. ते खरे आणि असत्य व्हॅल्यू धरू शकतात.
- खंडित:** जावा ब्रेक कीवर्ड लूप किंवा स्विच स्टेटमेंट तोडण्यासाठी वापरला जातो. हे स्पेसिफाइड परिस्थितीत प्रोग्रामचा करंट प्रवाह खंडित करते.
- बाइट:** Java byte कीवर्ड व्हेरिएबल that can hold 8-bit data values घोषित करण्यासाठी वापरला जातो.
- केस:** जावा केस कीवर्ड टेक्स्ट चे ब्लॉक चिन्हांकित करण्यासाठी स्विच स्टेटमेंटसह वापरले जाते.

- 6 **झेल:** जावा कॅच कीवर्ड ट्राय स्टेटमेंटद्वारे व्युत्पन्न केलेले अपवाद पकडण्यासाठी वापरला जातो. ते फक्त ब्लॉक केल्यानंतर वापरणे आवश्यक आहे.
- 7 **चार :** Java char कीवर्ड अॅक्सेसिबल घोषित करण्यासाठी वापरला जातो ज्यात unsigned 16-bit Unicode कॅरक्टर असू शकतात
- 8 **क्लास :** क्लास घोषित करण्यासाठी Java class keyword वापरला जातो.
- 9 **सुरू:** लूप चालू ठेवण्यासाठी Java continue कीवर्ड वापरला जातो. तो प्रोग्रामचा करंट प्रवाह चालू ठेवतो आणि स्पेसिफाइड स्थितीत उर्वरित कोड वगळतो.
- 10 **डीफॉल्ट:** स्विक स्टेटमेंटमध्ये कोडचा डीफॉल्ट ब्लॉक स्पेसिफाइड करण्यासाठी Java default कीवर्ड वापरला जातो.
- 11 **करा:** लूप घोषित करण्यासाठी कंट्रोल स्टेटमेंटमध्ये Java do कीवर्ड वापरला जातो. हे प्रोग्रामचा एक पार्ट अनेक वेळा पुनरावृत्ती करू शकते.
- 12 **दुप्पट:** Java दुहेरी कीवर्ड 64-bit floating-point number धारण करू शकणारे व्हेरिबल घोषित करण्यासाठी वापरले जाते.
- 13 **इतर:** Java else keyword चा वापर पर्यायी शाखांना सूचित करण्यासाठी केला जातो.
- 14 **enum:** Java enum कीवर्ड स्थिरांकांचा निश्चित संच परिभाषित करण्यासाठी वापरला जातो. एनम कन्स्ट्रक्टर नेहमी प्राइवेट किंवा डीफॉल्ट असतात.
- 15 **एक्सपेंडेड करते:** Java extends कीवर्ड हा क्लास दुसऱ्या क्लास किंवा इंटरफेसमधून काढला आहे हे दर्शविण्यासाठी वापरला जातो.
- 16 **अंतिम:** Java अंतिम कीवर्ड हे सूचित करण्यासाठी वापरले जाते की व्हेरिबलमध्ये स्थिर व्हॅल्यू असते. हे व्हेरिबलसह वापरले जाते. यूजर ला व्हेरिबलचे व्हॅल्यू अपडेट करण्यापासून प्रतिबंधित करण्यासाठी वापरले जाते.
- 17 **शेवटी:** शेवटी जावा कीवर्ड ट्राय-कॅच स्ट्रक्चरमध्ये कोडचा ब्लॉक दर्शवतो. अपवाद हाताळला किंवा नसला तरीही हा ब्लॉक नेहमी कार्यान्वित केला जातो.
- 18 **फ्लोट:** Java float key शब्द a32-bit floating-point number धारण करू शकणारे available घोषित करण्यासाठी वापरला जातो.
- 19 **च्या साठी:** जावा फॉर कीवर्डचा वापर फॉर लूप सुरू करण्यासाठी केला जातो. जेव्हा काही कंडीशन सत्य होते तेव्हा वारंवार सूचना/फंक्शनसचा संच अंमलात आणण्यासाठी याचा वापर केला जातो. पुनरावृत्तीची नंबर निश्चित असल्यास, लूपसाठी वापरण्याची शिफारस केली जाते.
- 20 **जर:** Java if keywords the condition. It executes if block if the condition is true.
- 21 **इम्प्लिमेंट्स:** इंटरफेस अंमलात आणण्यासाठी Java लागू करतो कीवर्ड वापरला जातो.
- 22 **आयात:** Java इंपोर्ट कीवर्ड क्लासेस आणि इंटरफेस सबलब्ध करून देतो आणि सध्याच्या सोर्स कोडमध्ये एक्सेस करण्यायोग्य बनवतो.
- 23 **याचे उदाहरण:** Java instance of keyword वापरला जातो की ऑब्जेक्ट स्पेसिफाइड क्लासर इंटरफेस लागू करतो याचे उदाहरण आहे की नाही हे तपासण्यासाठी.
- 24 **int:** Javaint कीवर्डचा वापर व्हेरिबल घोषित करण्यासाठी केला जातो ज्यामध्ये 32-बिट-साइन केलेले पूर्णांक असू शकतात.
- 25 **इंटरफेस:** Java interface कीवर्ड इंटरफेस घोषित करण्यासाठी वापरला जातो. यात फक्त ऑब्स्ट्रक्ट मेथड्स असू शकतात.
- 26 **लांब:** Java long keyword चा वापर केला जातो घोषित व्हेरिबल ज्यामध्ये 64-bit integer असू शकते.
- 27 **Netive:** JNI (JavaNativeInterface) वापरून मूळ कोडमध्ये मेथड्स लागू केली आहे हे स्पेसिफाइड करण्यासाठी Java मूळ कीवर्ड वापरला जातो.
- 28 **नवीन:** Javaneew कीवर्ड नवीन ऑब्जेक्ट्स तयार करण्यासाठी वापरला जातो.
- 29 **null :** Java null keyword हा रेफरेन्स कशाचाही रेफरेन्स देत नाही हे दर्शविण्यासाठी वापरला जातो. हे कचरा व्हॅल्यू काढून टाकते.
- 30 **पॅकेज:** Java पॅकेज कीवर्ड जावा पॅकेज घोषित करण्यासाठी वापरला जातो ज्यामध्ये क्लासेसचा समावेश होतो.
- 31 **प्राइवेट :** Java प्रायव्हेट कीवर्ड हा ऍक्सेस मॉडिफायर आहे. हे सूचित करण्यासाठी वापरले जाते की एखादी मेथड्स किंवा व्हेरिबल केवळ ज्या वर्गात घोषित केले जाते त्यात एक्सेस केला जाऊ शकतो.
- 32 **संरक्षित:** Java संरक्षित कीवर्ड हा ऍक्सेस मॉडिफायर आहे. हे पॅकेजमध्ये आणि पॅकेजच्या बाहेर एक्सेस योग्य असू शकते परंतु केवळ हेरिटेन्सद्वारे. ते वर्गासह लागू केले जाऊ शकत नाही.
- 33 **सार्वजनिक:** जावा पब्लिक कीवर्ड हा ऍक्सेस मॉडिफायर आहे. हे सूचित करण्यासाठी वापरले जाते की एखादी वस्तू कुठेही एक्सेस योग्य आहे. इतर सर्व सुधारकांमध्ये सर्वात विस्तृत व्याप्ती आहे.
- 34 **परत:** Java रिटर्न कीवर्डचा वापर मेथडमधून लिंगनिदान पूर्ण झाल्यावर परत येण्यासाठी केला जातो.
- 35 **short :** Java शॉर्टकीवर्ड हे व्हेरिबल घोषित करण्यासाठी वापरले जातात ज्यामध्ये 16-बिट इंटीजर असू शकतात.
- 36 **static :** व्हेरिबल किंवा मेथड्स ही क्लास मेथड्स आहे हे दर्शविण्यासाठी Java स्थिर कीवर्ड वापरला जातो. Java मधील स्थिर कीवर्ड मुख्यतः मेमरी मॅनेजमेंट साठी वापरला जातो.
- 37 **strictfp:** जावास्ट्रिक्ट एफपीआय जेव्हा पोर्टेबिलिटीची खात्री असेल तेव्हा फ्लोटिंग पॉइंट गणना प्रतिबंधित करण्यासाठी वापरली जाते.

- ३८ super :** Java सुपर कीवर्ड हा एक रेफरेन्स व्हेरिएबल आहे जो पालक वर्गाच्या वस्तूचा रेफरेन्स देण्यासाठी वापरला जातो. तात्काळ पॅरेन्टक्लास मेथड्स लागू करण्यासाठी याचा वापर केला जाऊ शकतो.
- 39 स्विच:** Java switch कीवर्डमध्ये एक स्विच स्टेटमेंट आहे जे टेस्टिंग मूल्यावर आधारित कोड कार्यान्वित करते. स्विच स्टेटमेंट मल्टिपल व्हॅल्यू विरुद्ध व्हेरिएबलची समानता तपासते.
- 40 समक्रमित:** Java सिंक्रोनाइझ केलेला कीवर्ड मल्टी थ्रेडेड कोडमधील गंभीर विपार्ट किंवा मेथड्स स्पेसिफाइड करण्यासाठी वापरला जातो.
- ४१ this :** Java हा कीवर्ड मेथड किंवा कन्स्ट्रक्टरमध्ये करंट ऑब्जेक्टचा रेफरेन्स देण्यासाठी वापरला जाऊ शकतो.
- 42 फेकणे:** Java थ्रो कीवर्ड स्पष्टपणे अपवाद टाकण्यासाठी वापरला जातो. थ्रो कीवर्ड प्रामुख्याने कस्टम अपवाद फेकण्यासाठी वापरला जातो. Itis नंतर एक उदाहरण.
- ४३ फेकतो:** Java throws कीवर्ड अपवाद घोषित करण्यासाठी वापरला जातो. तपासलेले अपवाद थ्रो सह प्रसारित केले जाऊ शकतात.

- ४४ transient :**जावा ट्रान्झिएंट कीवर्ड सीरियलायझेशनमध्ये वापरला जातो. तुम्ही कोणत्याही डेटा सदस्याला क्षणिक म्हणून परिभाषित केल्यास, ते अनुक्रमित केले जाणार नाही.
- ४५ प्रयत्न:** जावा ट्राय कीवर्ड कोडचा ब्लॉक सुरू करण्यासाठी वापरला जातो ज्याची अपवादांसाठी टेस्टिंग केली जाईल. ट्राय ब्लॉक नंतर एकतर पकडणे किंवा शेवटी ब्लॉक करणे आवश्यक आहे.
- ४६ void :** जावावॉइडकीवर्ड हे स्पेसिफाइड करण्यासाठी वापरला जातो या मेथड्स ला र्टर्न व्हॅल्यू नसते.
- ४७ volatile:** Java volatile keyword हे सूचित करण्यासाठी वापरले जाते की avariable सहज कालानुरूप बदलू शकते.
- ४८ while :** Java while कीवर्डचा वापर while loop सुरू करण्यासाठी केला जातो. हा लूप प्रोग्रामचा एक पार्ट अनेक वेळा पुनरावृत्ती करतो. जर पुनरावृत्तीची नंबर निश्चित केली गेली नाही, तर व्हेललूप वापरण्याची शिफारस केली जाते.

## Java मधील डेटाटाइप (Datatypes in Java)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- डेटाटाइप टाइप
- javaPrimitiveDataTypes
- non-Primitive डेटा टाइप

### DataTypesinJava

डेटा टाइप भिन्न आकार आणि व्हॅल्यू स्पेसिफाइड करतात जे व्हेरिएबलमध्ये स्टोर केले जाऊ शकतात. जावामध्ये दोन टाइप चे डेटाटाइप आहेत:

**Primitive डेटा टाइप :** Primitive डेटा प्रकारांमध्ये बुलियन, चार, बाइट, शॉर्ट, इंट, लॉंग, फ्लोट आणि डबल यांचा समावेश होतो.

**नॉन-प्रिमिटिव डेटाटाइप:** त्यानंतरच्या-Primitive डेटा प्रकारांमध्ये क्लास , इंटरफेस आणि अॅरे यांचा समावेश होतो.

### JavaPrimitiveDataTypes

जावा भाषेत, Primitive डेटा टाइप हे डेटा हाताळणीचे बिल्डिंग ब्लॉक्स आहेत. हे जावा भाषेत सबलब्ध असलेले सर्वात बेसिक डेटा टाइप आहेत.

**टीप:** जावा ही स्टॅटिकली टाइप केलेली प्रोग्रामिंग भाषा आहे. याचा अर्थ, सर्व व्हेरिएबल्स वापरण्यापूर्वी घोषित करणे आवश्यक आहे. व्हेरिएबलचे टाइप आणि नाव घोषित करणे आवश्यक आहे.

Primitive डेटा प्रकारांचे 8 टाइप आहेत:

- 1 booleandatatype
- 2 bytedatatype
- 3 chardatatype
- 4 shortdatatype

- 5 intdatatype
- 6 longdatatype
- 7 floatdatatype
- 8 doubledatatype

### JavaDataTypes

डेटा टाइप	डीफॉल्ट व्हॅल्यू	डीफॉल्ट आकार
बुलियन	खोटे	1 बिट
चार	'\u0000'	2 बाइट
बाइट	0	1 बाइट
लहान	0	2 बाइट
int	0	4 बाइट
लॉंब	0L	8 बाइट
फ्लोट	0.0f	4 बाइट
दुप्पट	०.०दि	8 बाइट



- शिफ्ट ऑपरेटर,
- रिलेशनल ऑपरेटर,
- BitwiseOperator,
- लॉजिकल ऑपरेटर,
- TernaryOperator आणि
- असाइनमेंट ऑपरेटर.

### JavaOperatorPrecedence

ऑपरेटर टाइप	रेन्ज	अग्रक्रम
युनरी	पोस्टफिक्स	expr++expr--
सबसर्ग	++expr--expr+expr-expr ~!	
अर्थमेटिक	गुणाकार	* /%
additive	+ -	
शिफ्ट	शिफ्ट	<<>>>>
संबंधीत	कम्प्यारीजन	<><=>= उदाहरण
उदाहरण	== !=	
बिटवाइज	bitwiseAND	&
bitwiseexclusiveOR	^	
bitwise inclusiveOR		
लॉजिकल	लॉजिकल आणि	&&
लॉजिकल	किंवा	
टर्नरी	तिरंगी	? :
असाइनमेंट	असाइनमेंट	= += -= *= /= %= &= ^=  = << >> >>> >>>=

### JavaUnaryOperator

Java unary ऑपरेटरना फक्त एक ऑपरेंड आवश्यक आहे. युनरी ऑपरेटरचा वापर विविध ऑपरेशन्स करण्यासाठी केला जातो.

- incrementing/decrementing a value by one
- negating an expression
- inverting the value of a boolean

JavaUnaryOperator उदाहरण: ++ आणि --

```
public class OperatorExample {
    public static void main(String args[]) {
        int x = 10;
        System.out.println(x++); // 10 (11)
        System.out.println(++x); // 12
        System.out.println(x--); // 12 (11)
        System.out.println(--x); // 10
    }
}
```

### आउटपुट:

10  
12  
12  
10

JavaUnaryOperator Example 2: ++ आणि --

```
public class OperatorExample {
    public static void main(String args[]) {
        int a = 10;
        int b = 10;
        System.out.println(a++ + ++a); // 10+12=22
        System.out.println(b+++b++); // 10+11=21
    }
}
```

## आउटपुट:

22

२१

JavaUnaryOperator उदाहरण: ~ आणि!

```
publicclassOperatorExample{
    public static void main(String args[]){inta=10;
    int b=-10;
    boolean c=true;
    booleand=false;
    System.out.println(~a);//-11 (minus of total positive
    value which starts from 0)
    System.out.println(~b);//9 (positive of total minus,
    positive starts from 0)
    System.out.println(!c);//false (opposite of boolean
    value)
    System.out.println(!d);//true
    }}
```

## आउटपुट:

-11

९

खोटे

खरे

## Java Arithmetic Operators

जावा अर्थमेटिक ऑपरेटर बेरीज, वजाबाकी, गुणाकार आणि भागाकार करण्यासाठी वापरले जातात. ते बेसिक गणितीय क्रिया म्हणून कार्य करतात.

## Java अर्थमेटिक ऑपरेटर उदाहरण

```
publicclassOperatorExample{
    public static void main(String args[]){
    inta=10;
    int b=5;
    System.out.println(a+b);//15
    System.out.println(a-b);//5
    System.out.println(a*b);//50
    System.out.println(a/b);//2
    System.out.println(a%b);//0
    }}
```

## आउटपुट:

१५

५

50

2

0

## JavaArithmeticOperatorExample:Expression

```
publicclassOperatorExample{
    publicstaticvoidmain(Stringargs[]){
    System.out.println(10*10/5+3-1*4/2);
    }}
```

## आउटपुट:

21

## JavaLeftShiftOperator

Java लेफ्ट शिफ्ट ऑपरेटर << हे मूल्यातील सर्व बिट्स ठराविक संख्येच्या डावीकडे अनेक वेळा शिफ्ट करण्यासाठी वापरले जाते.

जावा लेफ्ट शिफ्ट ऑपरेटर

```
ExamplepublicclassOperatorExample{
    publicstaticvoidmain(Stringargs[]){
    System.out.println(10<<2);//10*2^2=10*4=40
    `System.out.println(10<<3);//10*2^3=10*8=80
    System.out.println(20<<2);//20*2^2=20*4=80
    System.out.println(15<<4);//15*2^4=15*16=240
    }}
```

## आउटपुट:

40

80

80

240

## JavaRightShiftOperator

Java राईट शिफ्ट ऑपरेटर >> उजव्या ऑपरेंडने स्पेसिफाइड केलेल्या बिट्सच्या संख्येनुसार डाव्या ऑपरेंडचे व्हॅल्यू उजवीकडे हलविण्यासाठी वापरले जाते.

जावा राईट शिफ्ट ऑपरेटरचे उदाहरण

```
publicOperatorExample{
    public static void main(String args[]){
    System.out.println(10>>2);//10/2^2=10/4=2
    System.out.println(20>>2);//20/2^2=20/4=5
    System.out.println(20>>3);//20/2^3=20/8=2
    }}
```

## आउटपुट:

2

५

2

JavaShiftOperatorExample:>>vs>>>

```

publicclassOperatorExample{
publicstaticvoidmain(Stringargs[]){
//For positive number, >> and >>> works same
System.out.println(20>>2);
System.out.println(20>>>2);
//For negative number, >>> changes parity bit (MSB)
to 0
System.out.println(-20>>2);
System.out.println(-20>>>2);
}}

```

आउटपुट:

५

५

-5

१०७३७४१८१९

#### JavaANDOperatorउदाहरण:लॉजिकल आणि आणि बिटवाइज&

पहिली कंडीशन असत्य असल्यास लॉजिकल && ऑपरेटर दुसरी कंडीशन तपासत नाही. पहिली कंडीशन खरी असेल तरच ती दुसरी कंडीशन तपासते.

बिटवाइज आणि ऑपरेटर नेहमी दोन्ही अटी तपासतो की पहिली कंडीशन खरी आहे की खोटी.

```

publicclass OperatorExample{
publicstaticvoidmain(Stringargs[]){
int a=10;int b=5;intc=20;
System.out.println(a<b&&a<c);//false && true = false
System.out.println(a<b&a<c);//false&true=false
}}

```

आउटपुट:

खोटे

खोटे

#### JavaANDOperatorउदाहरण:लॉजिकल&&vsBitwise&

```

publicclassOperatorExample{
publicstaticvoidmain(Stringargs[]){
inta=10;
int b=5;
intc=20;
System.out.println(a<b&&a++<c);//false && true =
false
System.out.println(a);//10 because second condition
is not checked

```

```

System.out.println(a<b&a++<c);//false && true =
false
System.out.println(a);//11 because second condition
is checked
}}

```

आउटपुट:

असत्य10

असत्य11

#### JavaOROperatorExample:Logical||andBitwise |

लॉजिकल || पहिली कंडीशन सत्य असल्यास ऑपरेटर दुसरी कंडीशन तपासत नाही. पहिली कंडीशन खोटी असेल तरच ती दुसरी कंडीशन तपासते.

बिटवाइज | ऑपरेटर नेहमी दोन्ही अटी तपासतो की पहिली कंडीशन खरी आहे की खोटी.

```

publicclass OperatorExample{
publicstaticvoidmain(Stringargs[])
{
inta=10;
intb=5;
intc=20;
System.out.println(a>b||a<c);//true || true = true
System.out.println(a>b|a<c);//true|true=true
//|| vs|
System.out.println(a>b||a++<c);//true || true = true
System.out.println(a);//10 because second condition
is not checked
System.out.println(a>b|a++<c);//true | true = true
System.out.println(a);//11because second condition
is checked
}}

```

आउटपुट:

खरे

खरे

खरे

10

खरे

11

#### JavaTernaryOperator

Java Ternary operator चा वापर if-then-else स्टेटमेंटसाठी एक लाईन रिप्लेसमेंट म्हणून केला जातो आणि Javaprogramming मध्ये भरपूर



वापरला जातो. हा एकमेव कंडिशनल ऑपरेटर आहे जो तीन ऑपरेंड घेतो.

Java Ternary ऑपरेटर उदाहरण

```
Java Ternary Operator Example
public class OperatorExample {
    public static void main (String args[]){inta=2;
    intb=5;
    int min=(a<b)?a:b;
    System.out.println(min);
    }
```

आउटपुट:

2

दुसरे उदाहरण:

```
public class OperatorExample{
    public static void main(String args[]){
    inta=10;
    intb=5;
    int min=(a<b)?a:b;System.out.println(min);
    }
```

आउटपुट:

५

### JavaAssignmentOperator

Java असाइनमेंट ऑपरेटर सर्वात सामान्य ऑपरेटरपैकी एक आहे. हे त्याच्या उजवीकडील व्हॅल्यू त्याच्या डावीकडील ऑपरेंडला नियुक्त करण्यासाठी वापरले जाते.

Java असाइनमेंट ऑपरेटरचे उदाहरण

```
public class OperatorExample{
    public static void main(String args[]){
    inta=10;
    intb=20;
    a+=4;//a=a+4 (a=10+4)
    b-=4;//b=b-4 (b=20-4)
    System.out.println(a);
    System.out.println(b);
    }
```

आउटपुट:

14

16

Java असाइनमेंट ऑपरेटरचे उदाहरण

```
public class OperatorExample{
    public static void main(String[] args){
    inta=10;
    a+=3;//10+3
    System.out.println(a);
    a-=4;//13-4
    System.out.println(a);
    a*=2;//9*2
    System.out.println(a);
    a/=2;//18/2
    System.out.println(a);
    }
```

आउटपुट:

13

९

१८

९

### JavaAssignmentOperatorExample:Addingshort

```
public class OperatorExample{
    public static void main(String args[]){
    shorta=10;
    shortb=10;
    //a+=b;//a=a+binternallysofine
    a=a+b;//Compile time error because 10+10=20 now
    int
    System.out.println(a);
    }
```

आउटपुट:

टाइमर संकलित करा

टाइपकास्ट केल्यानंतर:

```
public class OperatorExample
{
    public static void main(String args[])
    {
    shorta=10;
    shortb=10;
    a=(short)(a+b);//20 which is int now converted to
    shortSystem.out.println(a);
    }
```

```
}
```

आउटपुट:

20

### Java I/O streams

या ट्यूटोरियलमध्ये आपण Java इनपुट/आउटपुट प्रवाह आणि त्यांचे टाइप जाणून घेऊ.

Java मध्ये, प्रवाह हा डेटाचा क्रम आहे जो स्त्रोताकडून वाचला जातो आणि डेस्टिनेशन स्थानावर लिहिला जातो.

```
class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
Run Code
```

स्त्रोताकडील डेटा वाचण्यासाठी इनपुट प्रवाह वापरला जातो. आणि, डेस्टिनेशन स्थानावर डेटा लिहिण्यासाठी आउटपुट प्रवाह वापरला जातो.

उदाहरणार्थ, आमच्या पहिल्या Hello World उदाहरणामध्ये, आम्ही स्ट्रिंग प्रिंट करण्यासाठी System.out चा वापर केला आहे. येथे, System.out हा एक टाइप चा आउटपुट प्रवाह आहे.

त्याचप्रमाणे, इनपुट घेण्यासाठी इनपुट प्रवाह आहेत.

इनपुट स्ट्रीम्स आणि आउटपुट स्ट्रीम्सबद्दल आपण नंतरच्या ट्यूटोरियल्समध्ये तपशीलवार शिकू.

### Streams(प्रवाहांचे) टाइप

प्रवाहात असलेल्या डेटावर अवलंबून, त्याचे वर्गीकरण केले जाऊ शकते:

- बाइट प्रवाह
- कॅरक्टर प्रवाह

#### बाइट प्रवाह (Streams)

सिंगल बाइट (8 बिट्स) डेटा वाचण्यासाठी आणि लिहिण्यासाठी बाइट प्रवाह वापरला जातो.

सर्व बाइट स्ट्रीम क्लासेस इनपुटस्ट्रीम आणि आउटपुटस्ट्रीम नावाच्या बेस अॅबस्ट्रॅक्ट क्लासेसमधून घेतले जातात.

#### कॅरक्टर प्रवाह (Streams)

कॅरक्टर स्ट्रीमचा वापर डेटाचा एकच कॅरक्टर वाचण्यासाठी आणि लिहिण्यासाठी केला जातो.

सर्व कॅरक्टर स्ट्रीम क्लासेस रीडर आणि रायटर या बेस अॅबस्ट्रॅक्ट क्लासेसमधून घेतले आहेत.

#### Java युजर इनपुट (स्कॅनर)

##### Java युजर इनपुट

स्कॅनर क्लास युजर इनपुट मिळविण्यासाठी वापरला जातो, आणि तो java.util पैकेजमध्ये आढळतो.

स्कॅनर क्लास वापरण्यासाठी, क्लासचा एक ऑब्जेक्ट तयार करा आणि स्कॅनर क्लास डॉक्युमेंटेशनमध्ये सबलब्ध असलेल्या कोणत्याही मेथड्स वापरा. आमच्या उदाहरणात, आम्ही nextLine() मेथड्स वापरू, जी स्ट्रिंग

```
import java.util.Scanner; // Import the Scanner class

class Main {
    public static void main(String[] args) {
        Scanner myObj = new Scanner(System.in); // Create a Scanner object
        System.out.println("Enter username");

        String userName = myObj.nextLine(); // Read user input
        System.out.println("Username is: " + userName); // Output user input
    }
}
```

वाचण्यासाठी वापरली जाते:

## उदाहरण

### रन एक्साम्पल

जर तुम्हाला पॅकेज काय आहे हे माहित नसेल तर आमचे Java Packages ट्यूटोरियल वाचा.

### इनपुट टाइप

मेथड्स	डिस्क्रिप्शन
NextBoolean()	यूजर कडून बुलियन व्हॅल्यू वाचते
NextByte()	यूजर कडून बाइट व्हॅल्यू वाचते
NextDouble()	यूजर कडून दुहेरी व्हॅल्यू वाचते
नेक्स्ट फ्लोट()	यूजर कडून फ्लोट व्हॅल्यू वाचते
NextInt()	यूजर कडून इंट व्हॅल्यू वाचते
पुढील ओळ()	यूजर कडून स्ट्रिंग व्हॅल्यू वाचते
NextLong()	यूजर कडून एक लांब व्हॅल्यू वाचतो
पुढील लहान()	यूजर कडून एक लहान व्हॅल्यू वाचतो

वरील उदाहरणात, आम्ही nextLine() मेथड्स वापरली, जी Strings वाचण्यासाठी वापरली जाते. इतर टाइप वाचण्यासाठी खालील तक्त्याकडे लक्ष द्या:

खालील उदाहरणामध्ये, आम्ही विविध प्रकारांचा डेटा वाचण्यासाठी वेगवेगळ्या मेथड्स वापरतो:

उदाहरण

```
import java.util.Scanner;
class Main{
```

Java.io.Console वर्गासाठी घोषणा पाहू:

मेथड्स	डिस्क्रिप्शन
Reader reader()	कन्सोलशी संबंधित रीडर ऑब्जेक्ट पुनर्प्राप्त करण्यासाठी याचा वापर केला जातो
String readLine()	कन्सोलमधील टेक्स्ट ची एक ओळ वाचण्यासाठी याचा वापर केला जातो.
String readLine(String fmt, Object... args)	हे एक फॉर्मॅटिंग प्रॉम्प्ट प्रदान करते त्यानंतर कन्सोलमधील टेक्स्ट ची सिंगल ओळ वाचते.
char[] readPassword()	कन्सोलवर प्रदर्शित होत नसलेला पासवर्ड वाचण्यासाठी याचा वापर केला जातो.
char[] readPassword(String fmt, Object... args)	ते एक फॉर्मॅटिंग प्रॉम्प्ट प्रदान करते आणि नंतर कन्सोलवर प्रदर्शित होत नसलेला पासवर्ड वाचतो.
Console format(String fmt, Object... args)	कन्सोल आउटपुटवर फॉर्मेटेड स्ट्रिंग लिहिण्यासाठी याचा वापर केला जातो प्रवाह
Console printf(String format Object... args)	कन्सोल आउटपुट प्रवाहावर स्ट्रिंग लिहिण्यासाठी याचा वापर केला जातो.
PrintWriter writer()	कन्सोलशी संबंधित प्रिंटर ऑब्जेक्ट पुनर्प्राप्त करण्यासाठी याचा वापर केला जातो.
void flush()	हे कन्सोल फ्लश करण्यासाठी वापरले जाते.

```
public static void main(String[] args){
Scanner myObj = new Scanner(System.in);
System.out.println("Enter name, age and salary:");
// String input
String name = myObj.nextLine();
// Numerical input
int age = myObj.nextInt();
double salary = myObj.nextDouble();
// Output input by user
System.out.println("Name: " + name);
System.out.println("Age: " + age);
System.out.println("Salary: " + salary);
}
}
```

### रन एक्साम्पल

**टीप:** तुम्ही चुकीचे इनपुट (उदा. नंबर त्मक इनपुटमधील टेक्स्ट ) एंटर केल्यास, तुम्हाला अपवाद/त्रुटी संदेश मिळेल (जसे की "InputMismatchException").

तुम्ही अपवादांबद्दल आणि अपवाद प्रकरणातील त्रुटी कशा हाताळायच्या याबद्दल अधिक वाचू शकता.

### जावा कन्सोल क्लास

जावा कन्सोल क्लास कन्सोलमधून इनपुट मिळविण्यासाठी वापरला जातो. हे टेक्स्ट आणि संकेतशब्द वाचण्यासाठी मेथड्स प्रदान करते.

तुम्ही कन्सोल क्लास वापरून पासवर्ड वाचल्यास, तो यूजर ला प्रदर्शित केला जाणार नाही.

java.io.Console क्लास आंतरिकरित्या सिस्टम कन्सोलसह संलग्न आहे. 1.5 पासून कन्सोल क्लास सुरू केला आहे.

कन्सोलमधून टेक्स्ट वाचण्यासाठी एक साथे उदाहरण पाहू.

- 1 स्ट्रिंग टेक्स्ट = System.console().readLine();
- 2 System.out.println("टेक्स्ट आहे: " + text);

### जावा कन्सोल क्लास डीक्लेरेशन

1 पब्लिक फायनल क्लास कन्सोल ऑब्जेक्ट इम्प्लमेंट्सचा एक्सपान्शन करतो फ्लश करण्यायोग्य

### जावा कन्सोल क्लास मेथड्स

#### कन्सोलचा ऑब्जेक्ट कसा मिळवायचा

सिस्टीम क्लास एक स्टॅटिक मेथड कन्सोल () प्रदान करते जे कन्सोल क्लासचे सिंगलटन उदाहरण देते.

```
1 सार्वजनिक स्थिर कन्सोल कन्सोल(){
```

कन्सोल क्लासचे उदाहरण मिळवण्यासाठी कोड पाहू.

```
1 कन्सोल c=System.console();
```

#### जावा कन्सोल उदाहरण

```
1 import java.io.Console;
```

```
2 class ReadStringTest{
```

```
3 public static void main(String args[]){
```

```
4 Console c=System.console();
```

```
5 System.out.println("Enter your name: ");
```

```
6 String n=c.readLine();
```

```
7 System.out.println("Welcome "+n);
```

```
Enter your name: Nakul Jain
Welcome Nakul Jain
```

```
8 }
```

```
9 }
```

#### Output

पासवर्ड वाचण्यासाठी Java कन्सोलचे उदाहरण1

```
1 import java.io.Console;
```

```
2 class ReadPasswordTest{
```

```
3 public static void main(String args[]){
```

```
4 Console c=System.console();
```

```
5 System.out.println("Enter password: ");
```

```
6 char[] ch=c.readPassword();
```

```
7 String ass=String.valueOf(ch);//converting char array into
string
```

```
8 System.out.println("Password is: "+pass);
```

```
9 }
```

```
10 }
```

#### Output

```
Enter password:
Password is: 123
```

## JAVA प्रोग्राम फ्लो कंट्रोल (JAVA Program Flow Control)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- कंडीशनल स्टेटमेंट
- लूप
- अरे.

### जावा कंडिशन आणि if स्टेटमेंट

तुम्हाला आधीच माहित आहे की जावा गणितातील नेहमीच्या लॉजिकल परिस्थितीचे सपोर्ट करते:

- Less than:  $a < b$
- Less than or equal to:  $a \leq b$
- Greater than:  $a > b$
- Greater than or equal to:  $a \geq b$
- Equal to  $a == b$
- Not Equal to:  $a != b$

तुम्ही या अटी वेगवेगळ्या निर्णयांसाठी वेगवेगळ्या क्रिया करण्यासाठी वापरू शकता.

### Java मध्ये खालील कंडीशनल स्टेटमेंट आहेत:

- कार्यान्वित करण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी वापरा, जर स्पेसिफाइड कंडीशन सत्य असेल
- कार्यान्वित करण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी इतर वापरा, जर तीच कंडीशन खोटी असेल तर
- टेस्टिंग साठी नवीन कंडीशन नमूद करायची असेल तर वापरा, जर पहिली कंडीशन असत्य असेल तर
- कार्यान्वित करण्यासाठी कोडचे अनेक पर्यायी ब्लॉक स्पेसिफाइड करण्यासाठी स्विक वापरा

The if Statement

कंडीशन सत्य असल्यास अंमलात आणण्यासाठी जावा कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी if स्टेटमेंट वापरा.

सिन्टेक्स

```
if(condition){
    // block of code to be executed if the condition is true
}
```

लक्षात ठेवा की जर लोअरकेस अक्षरात असेल. अप्परकेस अक्षरे (if किंवा IF) एरर निर्माण करेल : खालील उदाहरणामध्ये, 20 18 पेक्षा मोठे आहे की नाही हे सर्व प्यासाठी आम्ही दोन व्हॅल्यू ची टेस्टिंग करतो. जर स्थिती सत्य असेल, तर काही टेक्स्ट प्रिंटेड करा:

उदाहरण

```
जर(20>18){
    System.out.println("20 18 पेक्षा जास्त आहे");
}
```

ते स्वतः वापरून पहा

आम्ही व्हेरिबल्स देखील तपासू शकतो:

उदाहरण

```
int x =20;
int y =18;
जर(x > y){
    System.out.println("x y पेक्षा मोठा आहे");
}
```

स्वतः करून पहा"

उदाहरण स्पष्ट केले

वरील उदाहरणामध्ये x हा y पेक्षा मोठा आहे की नाही हे तपासण्यासाठी x आणि y ही दोन चल वापरतो (> ऑपरेटर वापरून). जसे x 20 आहे, आणि y 18 आहे, आणि आम्हाला माहित आहे की 20 18 पेक्षा मोठे आहे, आम्ही स्क्रीनवर "x y पेक्षा मोठा आहे" असे प्रिंटेड करतो.

दुसरे स्टेटमेंट

कंडिशन असत्य असल्यास अंमलात आणण्यासाठी कोडचा ब्लॉक स्पेसिफाइड करण्यासाठी else स्टेटमेंट वापरा.

सिन्टेक्स

```
if(condition){
    // block of code to be executed if the condition is true
}else{
    // block of code to be executed if the condition is false
}
```

उदाहरण

```
int time =20;
```

```

if(time < 18){
System.out.println("Good day.");
}else{
System.out.println("Good evening.");
}
// Outputs "Good evening."

```

**ते स्वतः वापरून पहा**

**उदाहरण स्पष्ट केले**

वरील उदाहरणामध्ये, वेळ (20) 18 पेक्षा जास्त आहे, त्यामुळे कंडीशन चुकीची आहे. यामुळे, आम्ही इतर स्थितीकडे जातो आणि स्क्रीनवर "शुभ संध्याकाळ" प्रिंट करतो. जर 18 पेक्षा कमी वेळ असेल तर प्रोग्रॅम "अच्छे दिन" छापेल.

**बाकी else स्टेटमेंट**

पहिली कंडीशन असत्य असल्यास नवीन कंडीशन स्पेसिफाइड करण्यासाठी else if स्टेटमेंट वापरा.

सिन्टेक्स

```

if(condition1){
// block of code to be executed if condition1 is true
}elseif(condition2){
// block of code to be executed if the condition1 is
false and condition2 is true
}else{
// block of code to be executed if the condition1 is
false and condition2 is false
}

```

उदाहरण

```

int time =22;
if(time < 10){
System.out.println("Good morning.");
}elseif(time < 18){
System.out.println("Good day.");
}else{
System.out.println("Good evening.");
}
// Outputs "Good evening."

```

**ते स्वतः वापरून पहा**

**उदाहरण स्पष्ट केले**

वरील उदाहरणामध्ये, वेळ (22) 10 पेक्षा जास्त आहे, म्हणून पहिली कंडीशन चुकीची आहे. पुढील कंडीशन, else if स्टेटमेंटमध्ये, देखील असत्य आहे, म्हणून आपण else कंडिशनवर जाऊ कारण कंडिशन1 आणि कंडिशन2

दोन्ही खोटे आहेत - आणि स्क्रीनवर "शुभ संध्याकाळ" प्रिंट करा.

तथापि, जर वेळ 14 असेल, तर आमचा प्रोग्रॅम "अच्छे दिन" छापेल.

जावा स्विच

जावा स्विच स्टेटमेंट

अनेक if..else स्टेटमेंट लिहिण्याऐवजी, तुम्ही स्विच स्टेटमेंट वापरू शकता. स्विच स्टेटमेंट अंमलात आणण्यासाठी अनेक कोड ब्लॉक्सपैकी एक निवडते:

सिन्टेक्स

```

switch(expression){
case x:
// code block
break;
case y:
// code block
break;
default:
// code block
}

```

**हे कसे कार्य करते:**

- स्विच अभिव्यक्तीचे एकदा व्हॅल्यू मापन केले जाते.
- अभिव्यक्तीच्या मूल्याची कम्प्यारीजन प्रत्येक केसच्या व्हॅल्यू शी केली जाते.
- जुळत असल्यास, कोडचा संबंधित ब्लॉक कार्यान्वित केला जातो.
- ब्रेक आणि डीफॉल्ट कीवर्ड ऐच्छिक आहेत, आणि या प्रकरणात नंतर डिस्क्रिप्शन केले जाईल

खालील उदाहरण आठवड्याच्या दिवसाच्या नावाची गणना करण्यासाठी आठवड्याच्या दिवसाची नंबर वापरते:

**उदाहरण**

```

int day =4;
switch(day){
case1:
System.out.println("Monday");
break;
case2:
System.out.println("Tuesday");
break;
case3:
System.out.println("Wednesday");
}

```

```

break;
case4:
    System.out.println("Thursday");
    break;
case5:
    System.out.println("Friday");
    break;
case6:
    System.out.println("Saturday");
    break;
case7:
    System.out.println("Sunday");
    break;
}
// Outputs "Thursday" (day 4)

```

### स्वतः करून पहा”

#### ब्रेक कीवर्ड

जेव्हा Java ब्रेक कीवर्डपर्यंत पोहोचतो, तेव्हा तो स्विच ब्लॉकमधून बाहेर पडतो. हे ब्लॉकमध्ये अधिक कोड आणि केस टेस्टिंग ची अंमलबजावणी थांबवेल.

जेव्हा एखादा सामना सापडतो आणि काम पूर्ण होते, तेव्हा विश्रांतीची वेळ असते. अधिक चाचण्या करण्याची गरज नाही.

ब्रेकमुळे अंमलबजावणीचा बराच वेळ वाचू शकतो कारण ते स्विच ब्लॉकमधील उर्वरित सर्व कोडच्या अंमलबजावणीकडे “दुर्लक्ष” करते.

#### डीफॉल्ट कीवर्ड

डीफॉल्ट कीवर्ड केस जुळत नसल्यास चालविण्यासाठी काही कोड स्पेसिफाइड करतो:

#### Example

```

int day =4;
switch(day){
case6:
    System.out.println("Today is Saturday");
    break;
case7:
    System.out.println("Today is Sunday");
    break;
default:
    System.out.println("Looking forward to the Week-
end");
}
// Outputs "Looking forward to the Weekend"

```

### स्वतः करून पहा”

लक्षात ठेवा की डीफॉल्ट स्टेटमेंट शेवटचे म्हणून वापरले असल्यास स्विच ब्लॉकमधील स्टेटमेंट, त्याला ब्रेकची आवश्यकता नाही.

### Loops

जोपर्यंत स्पेसिफाइड स्थिती गाठली जाते तोपर्यंत लूप कोडचा ब्लॉक कार्यान्वित करू शकतात.

लूप सुलभ आहेत कारण ते वेळेची बचत करतात, त्रुटी कमी करतात आणि ते कोड अधिक वाचनीय (रिडेबल) बनवतात.

### Java While Loop

स्पेसिफाइड कंडीशन सत्य असल्यापर्यंत कोडच्या ब्लॉकमधून लूप लूप होतो:सिन्टेक्स

```

while(condition){
// code block to be executed
}

```

खाली दिलेल्या उदाहरणात, जोपर्यंत व्हेरिएबल (i) 5 पेक्षा कमी असेल तोपर्यंत लूपमधील कोड पुन्हा पुन्हा चालू होईल:

#### उदाहरण

```

int i =0;
while(i <5){
    System.out.println(i);
    i++;
}

```

### स्वतः करून पहा”

टीप:कंडिशनमध्ये वापरलेले व्हेरिएबल वाढविण्यास विसरू नका, अन्यथा लूप कधीही संपणार नाही!

### डू/व्हाइल लूप (Do/While Loop)

do/while loop हा while loop चा एक टाईप आहे. हा लूप कोड ब्लॉक एकदा कार्यान्वित करेल, कंडिशन सत्य आहे की नाही हे तपासण्यापूर्वी, कंडिशन सत्य आहे तोपर्यंत तो लूपची पुनरावृत्ती करेल.

#### सिन्टेक्स

```

do{
// code block to be executed
}
while(condition);

```

खालील उदाहरण do/while लूप वापरते. लूप नेहमी किमान एकदा अंमलात आणला जाईल, जरी कंडीशन चुकीची असली तरीही, कारण कंडीशन तपासण्यापूर्वी कोड ब्लॉक कार्यान्वित केला जातो:

## उदाहरण

```
int i =0;
do{
System.out.println(i);
i++;
}
while(i <5);
```

### स्वतः करून पहा”

कंडिशनमध्ये वापरलेले व्हेरिबल वाढविण्यास विसरू नका, अन्यथा लूप कधीही संपणार नाही!

### जावा फॉर लूप

कोडच्या ब्लॉकमधून तुम्हाला किती वेळा लूप करायचा आहे हे जेव्हा तुम्हाला माहित असेल, तेव्हा लूपऐवजी फॉर लूप वापरा:

### सिन्टेक्स

```
for(statement 1; statement 2; statement 3){
// code block to be executed
}
```

**स्टेटमेंट 1** कोड ब्लॉकच्या अंमलबजावणीपूर्वी (एक वेळ) कार्यान्वित केले जाते.

**स्टेटमेंट 2** कोड ब्लॉक कार्यान्वित करण्यासाठी स्थिती परिभाषित करते. कोड ब्लॉक कार्यान्वित झाल्यानंतर

**स्टेटमेंट 3** (प्रत्येक वेळी) कार्यान्वित केले जाते. खालील उदाहरण 0 ते 4 अंक प्रिंटेड करेल:

### उदाहरण

```
for(int i =0; i <5; i++){
System.out.println(i);
}
```

### स्वतः करून पहा”

### उदाहरण स्पष्ट केले

लूप सुरू होण्यापूर्वी स्टेटमेंट 1 व्हेरिबल सेट करते (int i = 0).

स्टेटमेंट 2 लूप चालवण्याची स्थिती परिभाषित करते (मी 5 पेक्षा कमी असणे आवश्यक आहे). कंडीशन सत्य असल्यास, लूप पुन्हा सुरू होईल, जर ते खोटे असेल, तर लूप संपेल.

प्रत्येक वेळी लूपमधील कोड ब्लॉक कार्यान्वित केल्यावर स्टेटमेंट 3 व्हॅल्यू (i++) वाढवते.

### दुसरे उदाहरण

हे उदाहरण फक्त 0 आणि 10 मधील सम व्हॅल्यू प्रिंटेड करेल:

## उदाहरण

```
for(int i =0; i <=10; i = i +2){
System.out.println(i);
}
```

### ते स्वतः वापरून पहा

### नेस्टेड लूप

दुसऱ्या लूपच्या आत लूप ठेवणे देखील शक्य आहे. याला नेस्टेड लूप म्हणतात. “आतील लूप” “एक्सटर्नल लूप” च्या प्रत्येक पुनरावृत्तीसाठी एकदाच कार्यान्वित केले जाईल:

### उदाहरण

```
// Outer loop
for(int i =1; i <=2; i++){
System.out.println(“Outer: ” + i);// Executes 2 times
// Inner loop
for(int j =1; j <=3; j++){
System.out.println(“ Inner: ” + j);// Executes 6 times (2 * 3)
}
}
```

### जावा ब्रेक

तुम्ही या ट्यूटोरियलच्या आधीच्या अध्यायात वापरलेले ब्रेक स्टेटमेंट पाहिले आहे. हे स्विक स्टेटमेंटमधून “जंप आउट” करण्यासाठी वापरले होते.

ब्रेक स्टेटमेंटचा वापर लूपमधून बाहेर पडण्यासाठी देखील केला जाऊ शकतो.

जेव्हा मी 4 च्या बरोबरीचे असते तेव्हा हे उदाहरण लूप थांबवते:

### उदाहरण

```
for(int i =0; i <10; i++){
if(i ==4){
break;
}
System.out.println(i);
}
```

### ते स्वतः वापरून पहा

### Java Continue

कंटिन्यू स्टेटमेंट एक पुनरावृत्ती (लूपमध्ये) खंडित करते, जर स्पेसिफाइड स्थिती उद्भवते, आणि लूपमधील पुढील पुनरावृत्तीसह चालू राहते.

हे उदाहरण 4 चे व्हॅल्यू वगळते:

### उदाहरण



```

for(int i =0; i <10; i++){
if(i ==4){
continue;
}
System.out.println(i);
}

```

**ते स्वतः वापरून पहा**

### Break and Continue in While Loop

तुम्ही लूप असताना ब्रेक आणि सुरू ठेवू शकता:

#### ब्रेक उदाहरण

```

int i =0;
while(i <10){
System.out.println(i);
i++;
if(i ==4){
break;
}
}

```

#### स्वतः करून पहा”

#### उदाहरण सुरू ठेवा

```

int i =0;
while(i <10){
if(i ==4){
i++;
continue;
}
System.out.println(i);
i++;
}

```

#### जावा अॅरे

प्रत्येक व्हॅल्यूसाठी स्वतंत्र व्हेरिअबल्स घोषित करण्याऐवजी एकाच व्हेरिअबलमध्ये अनेक व्हॅल्यू स्टोअर करण्यासाठी अॅरेचा वापर केला जातो.

अॅरे घोषित करण्यासाठी, व्हेरिअबल टाइप स्केअर ब्रॅकेटसह परिभाषित करा:

#### स्ट्रिंग [ ] कार;

आम्ही आता एक व्हेरिअबल घोषित केले आहे ज्यामध्ये स्ट्रिंगचा अॅरे आहे. त्यात व्हॅल्यू घालण्यासाठी, तुम्ही स्वल्पविरामाने विभक्त केलेल्या लिस्ट मध्ये, कर्ली ब्रेसिसमध्ये व्हॅल्यू ठेवू शकता:

```
String[ ] cars ={"Volvo","BMW","Ford","Mazda"};
```

पूर्णांकांची अॅरे तयार करण्यासाठी, तुम्ही लिहू शकता:

```
int[ ] myNum ={10,20,30,40};
```

#### अॅरेच्या एलिमेंटमध्ये एक्सेस करा

तुम्ही इंडेक्स नंबरचा रेफरन्स देऊन अॅरे एलिमेंटत एक्सेस करू शकता. हे स्टेटमेंट कारमधील पहिल्या एलिमेंटच्या मूल्यामध्ये एक्सेस करते:

```
String[ ] cars ={"Volvo","BMW","Ford","Mazda"};
System.out.println(cars[0]);
```

// Outputs Volvo

#### स्वतः करून पहा”

**टीप:** अॅरे इंडेक्स 0 ने सुरू होतो: [0] हा पहिला एलिमेंट आहे. [१] दुसरा एलिमेंट आहे, इ.

#### अॅरे एलिमेंट बदला

स्पेसिफिक एलिमेंटचे व्हॅल्यू बदलण्यासाठी, पहा निर्देशांक क्रमांक:

#### उदाहरण

```
कार[0]="ओपल";
```

#### उदाहरण

ते स्वतः वापरून पहा

```
String[] cars ={"Volvo","BMW","Ford","Mazda"};
cars[0]="Opel";
System.out.println(cars[0]);
```

// Now outputs Opel instead of Volvo

#### अॅरे लेन्य

अॅरेमध्ये किती एलिमेंट आहेत हे सर्च प्यासाठी, लांबी प्रॉपर्टीज वापरा:

#### उदाहरण

```
String[] cars ={"Volvo","BMW","Ford","Mazda"};
System.out.println(cars.length);
// Outputs 4
```

## JAVA क्लास , ओव्हरलोडिंग आणि इन्हेरिटन्स (JAVA Classes, Overloading and Inheritance)

उद्दिष्टे: या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- java OOPs संकल्पना
- ऑब्जेक्ट क्लोनिंग
- मेथड्स चे टाइप .

### जावा ऑब्जेक्ट क्लास

#### Java OOPs संकल्पना

आपण OOP च्या बेसिक गोष्टींबद्दल जाणून घेऊ. ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग हा एक नमुना (थिअरी ) आहे जो अनेक संकल्पना (कल्पना) प्रदान करतो, जसे की हेरिटन्स, डेटा बाइंडिंग, पॉलीमॉर्फिझम इ.

सिमुला ही पहिली ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग भाषा मानली जाते. प्रोग्रामिंग पॅराडाइम जेथे सर्व काही असेनॉब्जेक्ट म्हणून दर्शविले जाते ज्याला अट्टलीऑब्जेक्टओरिएंटेड प्रोग्रामिंग भाषा म्हणून ओळखले जाते.

Smalltalk ही खऱ्या अर्थाने पहिली ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग भाषा मानली जाते. जावा, C#, PHP, Python, C++, इत्यादी लोकप्रिय ऑब्जेक्ट ओरिएंटेड भाषा आहेत.

ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंगचे मुख्य उद्दीष्ट वास्तविक-जगातील एलिमेंटची अमलबजावणी करणे आहे, उदाहरणार्थ, ऑब्जेक्ट, क्लासेस, अॅबस्ट्रॅक्शन, इनहेरिटन्स, पॉलीमॉर्फिझम, अॅबस्ट्रॅक्शन, एन्कॅप्सुलेशन इ.

#### OOPs (ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग सिस्टम)

ऑब्जेक्ट म्हणजे पेन, खुर्ची, टेबल, कॉम्प्युटर, घड्याळ इ. सारखी वास्तविक-ग्लोबल अस्तित्व. ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग ही क्लास आणि ऑब्जेक्ट्स वापरून प्रोग्राम डिझाइन करण्यासाठी एक मेथड्स किंवा नमुना आहे. हे काही संकल्पना प्रदान करून सॉफ्टवेअर विकास आणि देखभाल सुलभ करते:

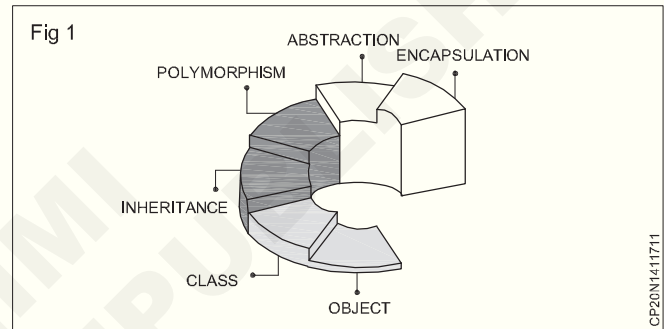
- Object
- Class
- Inheritance
- Polymorphism
- Abstraction
- Encapsulation

या संकल्पनांच्या व्यतिरिक्त, इतर काही संज्ञा आहेत जे डिनऑब्जेक्ट-ओरिएंटेड डिझाइन वापरतात:

- Coupling
- Cohesion

- Association
- Aggregation
- Composition

#### JavaOOPs संकल्पना



#### ऑब्जेक्ट

राज्य आणि वर्तन असलेली कोणतीही संस्था ऑब्जेक्ट म्हणून ओळखली जाते. उदाहरणार्थ, खुर्ची, पेन, टेबल, कीबोर्ड, बाईक, इ. हे फिजिकल किंवा लॉजिकल असू शकते.

ऑब्जेक्टला वर्गाचे उदाहरण म्हणून परिभाषित केले जाऊ शकते. ऑब्जेक्टमध्ये एड्रेस असतो आणि काही स्पेस मेमरी घेते. ऑब्जेक्ट्स एकमेकांच्या डेटा किंवा कोडचे तपशील जाणून घेतल्याशिवाय संवाद साधू शकतात. फक्त आवश्यक गोष्ट म्हणजे स्वीकारल्या गेलेल्या संदेशाचा टाइप आणि वस्तूद्वारे प्रतिसादाचा टाइप .

**उदाहरण:** कुत्रा ही एक वस्तू आहे कारण त्यात रंग, नाव, जात इ. तसेच शेपूट वाकवणे, भुंकणे, खाणे इ. यांसारखी वर्तणूक असते.

**क्लास :** कॉलड क्लासच्या वस्तूंचा संग्रह. इतिसालॉजिकल अस्तित्व.

क्लासला ब्लूप्रिंट म्हणून देखील परिभाषित केले जाऊ शकते ज्यामधून तुम्ही वैयक्तिक ऑब्जेक्ट तयार करू शकता. क्लास कोणतीही जागा वापरत नाही.

#### इन्हेरिटन्स (Inheritance)

जेव्हा एखादी वस्तू मूळ वस्तूचे सर्व प्रॉपर्टीज आणि वर्तन मिळवते तेव्हा त्याला इन्हेरिटन्स म्हणून ओळखले जाते. हे कोड पुन्हा वापरण्यायोग्यता प्रदान करते. रनटाइम पॉलीमॉर्फिझम साध्य करण्यासाठी वापरला जातो.

## बहुरूपता (Polymorphism)

जर एक कार्य वेगवेगळ्या प्रकारे केले जाते, तर त्याला बहुरूपता म्हणतात. उदाहरणार्थ: ग्राहकाला वेगळ्या मेथड्स ने पटवून देण्यासाठी, काहीतरी काढण्यासाठी, उदाहरणार्थ, आकार, त्रिकोण, आयत इ.

InJava, आम्ही बहुरूपता प्राप्त करण्यासाठी मेथड ओव्हरलोडिंग आणि मेथडओव्हरराइडिंग वापरतो.

दुसरे उदाहरण म्हणजे काहीतरी बोलणे; उदाहरणार्थ, cat म्याऊ बोलते, कुत्रा भुंकतो इ.

## ऑब्स्ट्रक्ट (Abstraction)

इंटरनल तपशील लपवणे आणि कार्यक्षमता दर्शविणे याला ऑब्स्ट्रक्ट ता म्हणून ओळखले जाते. उदाहरणार्थ, फोन कॉल, आम्हाला इंटरनल प्रक्रिया माहित नाही.

InJava, आम्ही ऑब्स्ट्रक्ट क्लास आणि इंटरफेस वापरतो.

## एन्कॅप्सुलेशन

बंधनकारक (किंवा रॅपिंग) कोड आणि डेटा एकत्रितपणे एकाच युनिटमध्ये एन्कॅप्सुलेशन म्हणून ओळखला जातो. उदाहरणार्थ, ऍकॅप्सूल, आयटीस वेगवेगळ्या औषधांनी गुंडाळलेले.

जावा क्लास हे एन्कॅप्सुलेशनचे उदाहरण आहे. जावा बीन हा पूर्णपणे एन्कॅप्सुलेट केलेला क्लास आहे कारण येथे सर्व डेटा सदस्य प्राइवेट आहेत.

## कपलिंग

कपलिंग म्हणजे ज्ञान किंवा इनफार्मेशन किंवा दुसऱ्या वर्गाचे अवलंबित्व. जेव्हा क्लास एकमेकांबद्दल इनफार्मेशन घेतात तेव्हा हे उद्भवते. एखाद्या वर्गाकडे दुसऱ्या वर्गाची तपशीलवार इनफार्मेशन असल्यास, मजबूत जोडणी असते. Java मध्ये, आपण क्लास , मेथड्स आणि फील्डची व्ह मानता पातळी प्रदर्शित करण्यासाठी प्राइवेट , संरक्षित आणि सार्वजनिक सुधारक वापरतो. तुम्ही कमकुवत कपलिंगसाठी इंटरफेस वापरू शकता कारण कोणतीही ठोस अंमलबजावणी नाही.

## एकसंधता

सुसंगतता एका एलिमेंटच्या पातळीचा रेफरन्स देते जे सिंगल सु-परिभाषित कार्य करते. एकच सु-परिभाषित कार्य अत्यंत सुसंगत मेथड्स ने केले जाते. कमकुवतपणे एकसंध मेथड्स मुळे कार्य वेगळ्या भागांमध्ये स्प्लिट केले जाईल. java.io पॅकेज हे एक अत्यंत एकत्रित पॅकेज आहे कारण त्यात I/O संबंधित क्लास आणि इंटरफेस आहे. तथापि, thejava.utilpackageisa कमकुवतपणे एकसंध असलेले पॅकेज आहे कारण ते क्लास आणि इंटरफेसशी संबंधित नाहीत.

## असोसिएशन

असोसिएशन वस्तूंमधील संबंध दर्शवते. येथे, एक ऑब्जेक्ट एका ऑब्जेक्टशी अनेक ऑब्जेक्ट्सशी संबंधित असू शकतो. वस्तूंमध्ये चार टाइप चे संबंध असू शकतात:

- OnetoOne
- OnetoMany
- ManytoOne, आणि
- ManytoMany

रिअल-टाइम उदाहरणांसह संबंध समजून घेऊ. उदाहरणार्थ, एका देशाला एक पंतप्रधान (एक ते एक) असू शकतो आणि पंतप्रधानांना अनेक मंत्री (एक ते अनेक) असू शकतात. तसेच, अनेक खासदारांकडे एक पंतप्रधान (अनेक ते एक) असू शकतो आणि अनेक मंत्र्यांकडे अनेक विपार्ट असू शकतात (अनेक ते अनेक).

असोसिएशन दिशाहीन किंवा द्विदिशात्मक असू शकते.

## एकत्रीकरण (Aggregation)

एकत्रीकरण हा असोसिएशन मिळवण्याचा एक पाथ आहे. एकत्रीकरण हे नाते दर्शवते जेथे एका ऑब्जेक्टमध्ये त्याच्या स्थितीचा पार्ट म्हणून इतर वस्तू असतात. हे वस्तूंमधील कमकुवत संबंध दर्शवते. त्याला अहास असेही म्हणतात a relationship in Java.Like, inheritance is represent the is a relationship. It is another way to use objects.

## रचना (Composition)

रचना देखील असोसिएशन प्राप्त करण्याचा एक पाथ आहे. रचना त्या संबंधाचे प्रतिनिधित्व करते जेथे एक ऑब्जेक्टमध्ये त्याच्या स्थितीचा एक पार्ट म्हणून इतर वस्तू असतात. अंतर्भूत वस्तू आणि अवलंबित्व वस्तू यांच्यात मजबूत संबंध आहे. ही अशी अवस्था आहे जिथे वस्तूंना स्वतंत्र अस्तित्व नसते. तुम्ही पॅरेंटऑब्जेक्ट हटवल्यास, सर्व चाइल्डऑब्जेक्ट ऑटोमैटिक हटवले जातील.

## OOPsover प्रक्रिया-देणारं प्रोग्रामिंग भाषेचा फायदा

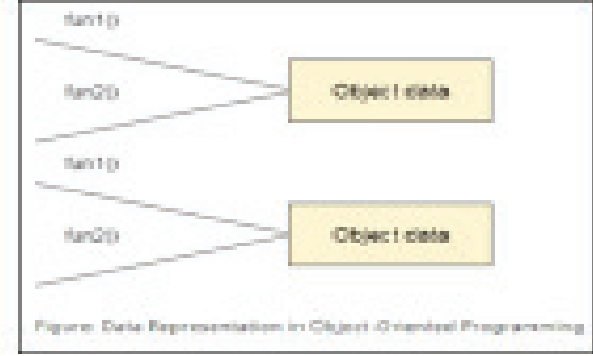
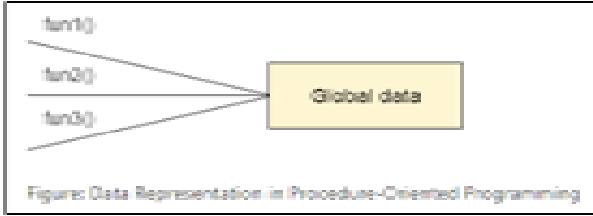
- 1 OOPs विकास आणि देखभाल सुलभ करते, तर, प्रक्रिया देणार्या प्रोग्रामिंग भाषेत, प्रकल्प आकार वाढला की कोड वाढल्यास मॅनेज करणे सोपे नसते.
- 2 OOPs डेटा लपविण्याची सुविधा देतात, तर, प्रक्रियाभिमुख प्रोग्रामिंग भाषेत, ग्लोबल डेटा कोठूनही ऍक्सेस केला जाऊ शकतो.
- 3 OOPs वास्तविक-जगातील इव्हेंट्स अधिक प्रभावीपणे अनुकरण करण्याची क्षमता प्रदान करतात. जर आपण ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग भाषा वापरत असाल तर आपण वास्तविक जगाच्या समस्येचे निराकरण करू शकतो.

## ग्लोबलडेटा

**आकृती:** DataRepresentationin Procedure-oriented Programming (Fig 2)

**ऑब्जेक्टडेटा :** ऑब्जेक्ट-ओरिएंटेड प्रोग्रामिंग भाषा आणि भाषेमध्ये काय फरक आहे? ऑब्जेक्ट-आधारित प्रोग्रामिंग भाषा इनहेरिटन्स वगळता OOPs च्या सर्व वैशिष्ट्यांचे अनुसरण करते. JavaScript आणि VB Script ही ऑब्जेक्ट-आधारित प्रोग्रामिंग भाषांची उदाहरणे आहेत.

Fig 2



### तुम्हाला माहित आहे का?

- मुख्य मेथड्स ओव्हरलोड करू शकता?
- AJavaConstructorreturns असेसमेंट करतात पण, काय?
- आपण मुख्य मेथड्स शिवाय प्रोग्रॅम तयार करू शकतो?
- हा कीवर्ड वापरण्याचे सहा पाथ काय आहेत?
- जावा मध्ये एकापेक्षा जास्त इन्हेरिटन्स का समर्थित नाही?
- एकत्रीकरण का वापरावे?
- आपण स्टॅटिक मेथड्स ओव्हरराइड करू शकतो?
- कोव्हेरियन रिटर्नटाइप काय आहे?
- जावासुपर कीवर्डचे तीन यूजेस काय आहेत?
- इन्स्टन्स इनिशिएलायझर ब्लॉक का वापरावे?
- रिक्त अंतिम व्हेरिएबलचा वापर काय आहे?
- अमरकर किंवा टॅग केलेले इंटरफेस म्हणजे काय?
- रन टाइम पॉलीमॉर्फिज्मर डायनॅमिक मेथड म्हणजे पॅच म्हणजे काय?
- स्टॅटिक आणि डायनॅमिक बाइंडिंगमध्ये काय फरक आहे?
- Howdowncastingispossible inJava?
- प्राइवेट कन्स्ट्रक्टरचा हेतू काय आहे?
- ऑब्जेक्टक्लॉनिंग म्हणजे काय?

### OOPs संकल्पना काय असतील?

- OOPs चा फायदा
- आमचे अधिवेशन
- ऑब्जेक्ट आणि क्लास

- मेथड्स ओव्हरलोडिंग
- कन्स्ट्रक्टर
- स्टॅटिक कीवर्ड
- सहा वापरासह हा कीवर्ड
- इन्हेरिटन्स
- एकत्रीकरण
- मेथड्स ओव्हरराइडिंग
- CovariantReturnType
- सुपरकीवर्ड
- InstanceInitializerblock
- अंतिम कीवर्ड
- ऑब्स्ट्रक्ट क्लास
- इंटरफेस
- रनटाइम पॉलीमॉर्फिज्म
- स्टॅटिक आणि डायनॅमिक बाइंडिंग
- ऑपरेटरच्या बरोबरीने डाऊनकास्ट करणे
- पॅकेज
- ऍक्सेस मॉडिफायर्स
- एन्कॅप्सुलेशन

### ऑब्जेक्ट क्लॉनिंग

#### Method in Java

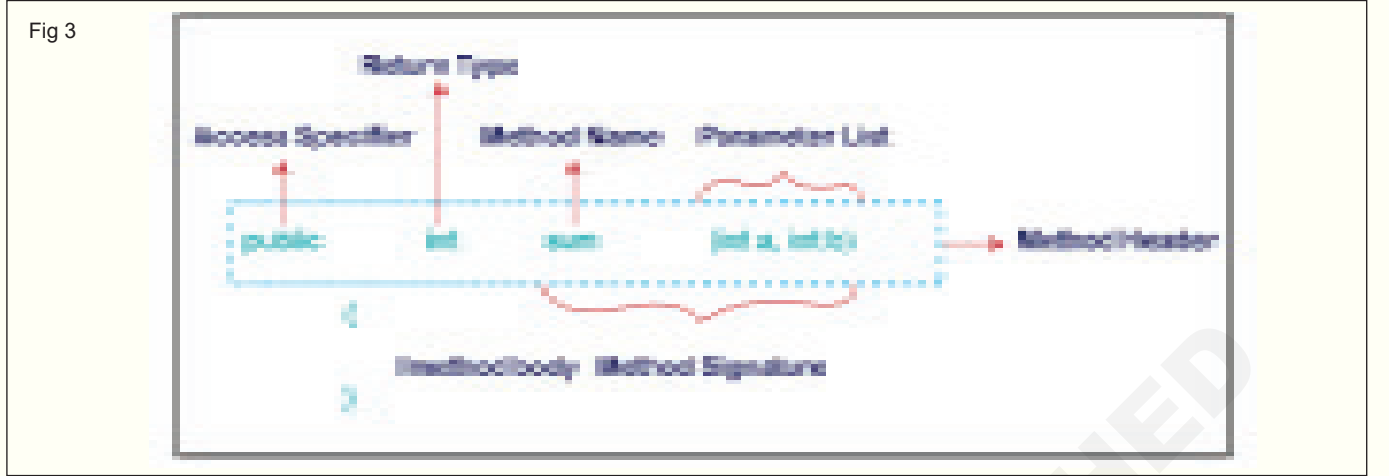
सर्वसाधारणपणे, मेथड्स म्हणजे काही कार्य करण्याचा एक पाथ . त्याचप्रमाणे, Java मधील मेथड्स ही सूचनांचा संग्रह आहे जी स्पेसिफिक कार्य करते. हे कोडची पुनः संयोजिता प्रदान करते. आपण मेथड्स वापरून कोड देखील सहज बदलू शकतो. या विभागात, आपण जावा मधील मेथड्स काय आहे, पद्धतीचे टाइप , मेथड डिक्लोरेशन आणि ऍमेथोडिन जावा कसे कॉल करावे हे शिकू.

#### जावा काय?

मेथड्स म्हणजे कोडचा एक ब्लॉक किंवा विधानांचा संग्रह किंवा स्पेसिफिक कार्य किंवा ऑपरेशन करण्यासाठी एकत्रित केलेल्या कोडचा संच. हे कोडची पुनः संयोजिता प्राप्त करण्यासाठी वापरले जाते. आम्ही एक मेथड्स एकदा लिहितो आणि ती अनेक वेळा वापरतो. आम्हाला पुन्हा पुन्हा कोड लिहिण्याची आवश्यकता नाही. हे कोडचे सोपे बदल आणि वाचनीय (रिडेबल)ता देखील प्रदान करते, फक्त जोडणे किंवा कोडचा एक पार्ट काढून टाकणे. जेव्हा आपण vokeit मध्ये कॉल करतो तेव्हाच ही मेथड्स अंमलात आणली जाते.

Java मधील सर्वात महत्वाची मेथड्स म्हणजे main() मेथड्स . तुम्हाला main() मेथड्स बदल अधिक वाचायचे असल्यास, <https://www.javatpoint.com/java mainmethod> या लिंकवर जा.

**मेथड डिक्लेरेशन (पिक्चर 3) :** मेथड डिक्लेरेशन मेथड अॅट्रिब्यूट्सबद्दल इनफार्मेशन देते, जसे की व्ह मानता, रिटर्न-टाइप, नाव आणि आर्ग्यूमेंट्स . यात सहा एलिमेंट आहेत जे मेथड हेडर म्हणून ओळखले जातात, जसे आपण खालील चित्रात दाखवले आहे.



**मेथड स्वाक्षरी:** प्रत्येक मेथड्स ची एक मेथड्स स्वाक्षरी असते. तो मेथड्स घोषणेचा एक पार्ट आहे. त्यामध्ये थीमचे नाव आणि पॅरामीटर लिस्ट समाविष्ट आहे.

**एक्सेस स्पेसिफाइड :** एक्सेस स्पेसिफायर किंवा मॉडिफायर हा मेथड्स चा एक्सेस टाईप आहे. ते थीमची व्ह मानता स्पेसिफाइड करते. जावा चार टाईप चे एक्सेस स्पेसिफाइड कर्ता प्रदान करते:

- **पब्लिक:** जेव्हा आपण पब्लिक स्पेसिफायरइनोअर अॅप्लिकेशन वापरतो तेव्हा ही मेथड्स सर्व वर्गासाठी एक्सेस योग्य असते.
- **प्राइवेट :** जेव्हा आम्ही प्राइवेट एक्सेस स्पेसिफायर वापरतो, तेव्हा ही मेथड्स फक्त ज्या वर्गामध्ये परिभाषित केली आहे त्या वर्गामध्येच एक्सेस करता येते.
- **संरक्षित:** जेव्हा आम्ही संरक्षित एक्सेस स्पेसिफायर वापरतो, तेव्हा मेथड्स समान पॅकेजमध्ये किंवा सबवर्गामध्ये भिन्न नसलेल्या पॅकेजमध्ये एक्सेस योग्य असते.
- **डीफॉल्ट:** जेव्हा आपण मेथड डिक्लेरेशनमध्ये कोणतेही एक्सेस स्पेसिफायर वापरत नाही, तेव्हा Java डीफॉल्टनुसार डिफॉल्ट एक्सेस स्पेसिफायर वापरते. हे फक्त त्याच पॅकेजमधून दिसते.
- **रिटर्न टाईप :** रिटर्न टाईप हा डेटा टाईप आहे जो मेथड्स परत करतो. त्यात Primitive डेटा टाईप , ऑब्जेक्ट, संग्रह, शून्य इ. असू शकतो. जर मेथड्स काहीही परत करत नसेल, तर आम्ही शून्य कीवर्ड वापरतो.
- **मेथड्स चे नाव:** हे एक अद्वितीय नाव आहे जे एखाद्या मेथड्स चे नाव परिभाषित करण्यासाठी वापरले जाते. ते मेथड्स च्या कार्यक्षमतेशी संबंधित असले पाहिजे. समजा, जर आपण दोन नंबर च्या वजाबाकीसाठी एक मेथड्स तयार करत आहोत, तर themethodname आवश्यक आहे वजाबाकी().एक मेथड्स त्याच्या नावाने चालविली जाते.

- **पॅरामीटर लिस्ट :** ही स्वल्पविरामाने विभक्त केलेल्या आणि कंसाच्या जोडीमध्ये बंद केलेल्या पॅरामीटर्सची लिस्ट आहे. त्यात डेटाटाईप आणि व्हेरिएबल नाव आहे. जर मेथड्स मध्ये कोणतेही पॅरामीटर नसेल तर कंस रिक्त सोडा.
- **मेथड्स बॉडी:** तो मेथड्स घोषणेचा एक पार्ट आहे. त्यात करावयाच्या सर्व क्रियांचा समावेश आहे. हे कर्लीब्रिसेसच्या जोडीमध्ये बंद आहे.

### मेथड्स चे नाव देणे

मेथड्स परिभाषित करताना, लक्षात ठेवा की मेथड्स चे नाव क्रियापद असले पाहिजे आणि लहान अक्षराने सुरू झाले पाहिजे. मेथड्स च्या नावात दोनपेक्षा जास्त शब्द असल्यास, पहिले नाव क्रियापद आणि त्यानंतर स्पेशल ण किंवा संज्ञा असणे आवश्यक आहे. मल्टीवर्ड मेथड्स च्या नावामध्ये, पहिला शब्द वगळता प्रत्येक शब्दाचे पहिले अक्षर मोठ्या अक्षरात असले पाहिजे. उदाहरणार्थ:

सिंगल -शब्द मेथड्स चे नाव: बेरीज(), एरिया ()

बहु-शब्द मेथड्स चे नाव: areaOfCircle(), stringComparision()

हे देखील शक्य आहे की एखाद्या मेथड्स चे नाव त्याच वर्गातील दुसऱ्या मेथड्स च्या नावासारखे आहे, ते मेथोडोव्हरलोडिंग म्हणून ओळखले जाते.

### मेथड्स चे टाईप

जावा मध्ये दोन टाईप ची मेथड्स :

- पूर्वनिर्धारित मेथड्स
- युजर -परिभाषित मेथड्स

### पूर्वनिर्धारित मेथड्स

Java मध्ये, पूर्वनिर्धारित मेथड्स ही मेथड्स आहे जी Java क्लास लायब्ररीमध्ये आधीच परिभाषित केलेली आहे ती पूर्वनिर्धारित मेथड्स म्हणून ओळखली जाते. हे स्टॅण्डर्ड लायब्ररी मेथड्स किंवा अंगभूत मेथड्स म्हणून देखील ओळखले जाते. आम्ही या मेथड्स थेट कोणत्याही

वेळी प्रोग्राममध्ये कॉल करून वापरू शकतो. काही पूर्वनिर्धारित मेथड्स आहेत length(), equals(), compareTo(), sqrt(), इ. जेव्हा आम्ही आमच्या प्रोग्राममधील कोणत्याही पूर्वनिर्धारित पद्धतींना कॉल करतो, तेव्हा लायब्ररीमध्ये आधीपासून स्टोर केलेल्या बॅकग्राऊंड तील संबंधित पद्धतींशी संबंधित कोडची सीरीज तयार होते.

प्रत्येक पूर्वनिर्धारित मेथड्स वर्गामध्ये परिभाषित केली जाते. जसे की print() मेथड्स thejava.io.PrintStream वर्गात परिभाषित केली आहे. ते स्टेटमेंट प्रिंटेड करते जे आपण मेथड्स मध्ये लिहितो. उदाहरणार्थ, print("Java"), itprintsJava ontheconsole.

चला पूर्वनिर्धारित मेथड्स चे उदाहरण पाहू.

## डेमो.जावा

```
publicclassDemo
{
publicstaticvoidmain(String[]args)
{
//using themax()methodofMathclass
System.out.print ("The maximum numberis: " +Math.
max(9,7));
}
}
```

## आउटपुट:

कमाल नंबर :9

वरील उदाहरणात, आम्ही तीन पूर्वनिर्धारित मेथड्स main(), print(), आणि max() वापरल्या आहेत. आम्ही या मेथड्स थेट घोषणेशिवाय वापरल्या आहेत कारण त्या पूर्वनिर्धारित आहेत. प्रिंट() मेथड्स ही प्रिंटस्ट्रीम क्लासची एक मेथड्स आहे जी कन्सोलवर निकाल प्रिंटेड करते.

**max**

```
public static int max(int a,
int b)
```

Returns the greater of two int values. same value.

**Parameters:**

a - an argument.

b - another argument.

**Returns:**

the larger of a and b.

max() मेथड्स ही गणित वर्गाची एक मेथड्स आहे जी दोन नंबर पेक्षा जास्त परत करते.

लिंक वापरून आपण कोणत्याही पूर्वनिर्धारित मेथड्स ची मेथड्स स्वाक्षरी देखील पाहू शकतो <https://docs.oracle.com/> जेव्हा आम्ही लिंकद्वारे आणि कमाल () मेथड्स ची स्वाक्षरी पाहतो तेव्हा आम्हाला खालील गोष्टी आढळतात:

वरील मेथड्स च्या स्वाक्षरीमध्ये, आपण पाहतो की मेथड सिग्नेचरमध्ये ऍक्सेस स्पेसिफायर पब्लिक, नॉन-ऍक्सेसमॉडिफायर स्टॅटिक, रिटर्न टाईप int, मेथड नेम max(), पॅरामीटर लिस्ट (int a, int b) आहे. वरील उदाहरणात, मेथड्स परिभाषित करण्याऐवजी, आम्ही फक्त मेथड्स चालविली आहे. हा पूर्वनिर्धारित मेथड्स चा फायदा आहे. हे प्रोग्रामिंग कमी क्लिष्ट करते. त्याचप्रमाणे, आपण print() मेथड्स ची मेथड्स स्वाक्षरी देखील पाहू शकतो.

## युजर -परिभाषित मेथड्स

युजर किंवा प्रोग्रामरने लिहिलेली मेथड्स युजर -परिभाषित मेथड्स म्हणून ओळखली जाते. या मेथड्स मध्ये आवश्यकतेनुसार बदल केले जातात.

## युजर -परिभाषित मेथड्स कशी तयार करावी

चला auserdefinedmethodthatchecksthe number सम किंवा विषम बनवू. प्रथम, आपण मेथड्स परिभाषित करू.

```
//userdefinedmethod
publicstaticvoidfindEvenOdd(intnum)
{
//method bodyif(num%2==0)
System.out.println(num+" is even");else
System.out.println(num+"isodd");
}
```

आम्ही findevenodd() नावाची वरील मेथड्स परिभाषित केली आहे. त्यात टाईप int ची पॅरामीटर नंबर आहे. मेथड्स कोणतेही व्हॅल्यू परत करत नाही म्हणून आम्ही void वापरले आहे. मेथड बॉडीमध्ये नंबर सम किंवा विषम आहे हे तपासण्यासाठी पायर्च्या असतात. जर नंबर सम असेल, तर ती नंबर सम आहे, अन्यथा नंबर विषम छापते.

## InvokeUser-defined Method कसे कॉल करावे

एकदा आपण एक मेथड्स परिभाषित केली की ती कॉल केली पाहिजे. प्रोग्राममधील मेथड्स कॉल करणे सोपे आहे. जेव्हा wecallorinvokeauser-defined method, theprogram कंट्रोल म्हणतात मेथडमध्ये स्थानांतरित होते.

```
import
java.util.Scanner;public
classEvenOdd
{
publicstaticvoidmain(String args[])
```

```

{
//creatingScannerclassobjectScannersc
an=newScanner(System.in);
System.out.print("Enterthnumber:");
//reading value from the
userintnum=scan.nextInt();
//method
callingfindEvenOdd
(num);
}

```

वरील कोड स्निपेटमध्ये, कंपायलर रेषेवर पोहोचताच EvenOdd (num) शोधा, मेथड्स वर कंट्रोल हस्तांतरित करते आणि त्यानुसार आउटपुट देते. एका प्रोग्राममध्ये कोडचे स्निपेट्स एकत्र करून ते कार्यान्वित करू.

#### EvenOdd.java

```

import
java.util.Scanner;public
classEvenOdd
{
publicstaticvoidmain(String args[])
{
//creatingScannerclassobjectScannersc
an=newScanner(System.in);
System.out.print("Enterthnumber:");
//reading value from
userintnum=scan.nextInt();
//method
callingfindEvenOdd
(num);
}
//userdefinedmethod
publicstaticvoidfindEvenOdd(intnum)
{
//method
bodyif(num%2
==0)
System.out.println(num+" is
even");else
System.out.println(num+"isodd");
}
}

```

#### आउटपुट 1:

क्रमांक प्रविष्ट करा:

1212 सम

#### आउटपुट 2:

क्रमांक प्रविष्ट करा: 99

#### 99 clogs

कॉलिंग मेथड्स ला व्हॅल्यू परत करणारा दुसरा प्रोग्राम पाहू.

पुढील प्रोग्राममध्ये, आम्ही दोन नंबर ची बेरीज करणारी add() नावाची मेथड्स परिभाषित केली आहे. यात पूर्णांक प्रकाराचे दोन पॅरामीटर्स n1 आणि n2 आहेत. n1 आणि n2 ची व्हॅल्यू अनुक्रमे a आणि b च्या मूल्याशी संबंधित आहेत. म्हणून, मेथड्स a आणि b चे व्हॅल्यू जोडते आणि व्हेरिअबल्समध्ये स्टोर करते आणि बेरीज मिळवते.

#### Addition.java

```

publicclassAddition
{
publicstaticvoidmain(String[]args)
{
int a =
19;intb
=5;
//methodcalling
int c = add(a, b);//a and b are actual
parametersSystem.out.println("Thesum of
aandbis="+c);
}
//userdefinedmethod
publicstaticintadd(intn1,intn2)//n1andn2are formal
parameters
{
int
s;s=n1+n
2;
returns; //returningthesum
}
}

```

#### आउटपुट:

The sum of a and b is=24

**स्टॅटिक मेथड्स** : स्टॅटिक कीवर्ड असलेली मेथड्स स्थिर मेथड्स म्हणून ओळखली जाते. दुसऱ्या शब्दांत, वर्गाच्या उदाहरणाऐवजी वर्गाशी संबंधित

असलेली मेथड्स स्थिर मेथड्स म्हणून ओळखली जाते. आपण मेथडच्या नावापूर्वी स्टॅटिक हा कीवर्ड वापरून स्थिर मेथड्स देखील तयार करू शकतो.

स्थिर मेथड्स चा मुख्य फायदा हा आहे की आपण त्याला ऑब्जेक्ट न बनवता कॉल करू शकतो. हे स्टॅटिक डेटामेम्बर्समध्ये एक्सेस करू शकते आणि त्याचे व्हॅल्यू देखील बदलू शकते. हे उदाहरण मेथड्स तयार करण्यासाठी वापरले जाते. हे क्लासचे नाव वापरून चालवले जाते. स्टॅटिक मेथड्स चे सर्वोत्तम उदाहरण themain () मेथड्स आहे.

### स्टॅटिकचे उदाहरण

#### methodDisplay.java

```
public class Display
{
    public static void main(String[] args)
    {
        show();
    }
    static void show()
    {
        System.out.println("It is an example of static method.");
    }
}
```

### आउटपुट:

हे स्टॅटिक मेथड्स चे उदाहरण आहे.

### इंस्टेंस मेथड्स

वर्गाची मेथड्स इंस्टेंस मेथड्स म्हणून ओळखली जाते. ही वर्गात परिभाषित केलेली नॉन-स्टॅटिक मेथड्स आहे. उदाहरण मेथड्स कॉल करण्यापूर्वी किंवा इनव्हॉईंग करण्यापूर्वी, त्याच्या वर्गाचा एक ऑब्जेक्ट तयार करणे आवश्यक आहे. उदाहरण मेथड्स चे उदाहरण पाहू.

### इंस्टेंस मेथड्स Example.java

```
public class InstanceMethodExample
{
    public static void main(String[] args)
    {
        //Creating an object of the class
        InstanceMethodExample obj = new InstanceMethodExample();
        //invoking
        instanceMethodSystem.out.println("The sum is:" + obj.add(12,13));
    }
    ints;
    //user-defined method because we have not used static keyword
    public int add(inta, intb)
    {
        s = a+b;
        //returning the sum returns;
    }
}
```

### आउटपुट:

बेरीज आहे: 25

इंस्टेंस मेथड्स चे दोन टाइप आहेत:

- एक्सेसर मेथड्स
- मुतटर मेथड्स

**एक्सेसर मेथड्स :** इन्स्टन्स व्हेरिएबल वाचणारी मेथड्स एक्सेसर मेथड्स म्हणून ओळखली जाते. आम्ही ते सहजपणे ओळखू शकतो कारण या मेथड्स चा सबसर्ग get या शब्दाबरोबर आहे. हे गेटर्स म्हणून देखील ओळखले जाते. हे प्राइवेट क्षेत्राचे व्हॅल्यू परत करते. आयटीआय प्राइवेट क्षेत्राचे व्हॅल्यू मिळवण्यासाठी वापरले जाते.

### उदाहरण

```
public int getld()
{
    रिटर्न आयडी;
}
```

**मुतटर मेथड्स :** मेथड्स इन्स्टन्स व्हेरिएबल वाचते आणि व्हॅल्यूज देखील बदलते. आपण ते सहजपणे ओळखू शकतो कारण मेथड्स सेट या शब्दाबरोबर प्रीफिक्स केलेली आहे. हे सेटर किंवा मॉडिफायर म्हणून देखील ओळखले जाते. ते काहीही परत करत नाही. हे फील्डवर अवलंबून असलेल्या समान डेटा प्रकाराचे पॅरामीटर स्वीकारते. हे प्राइवेट क्षेत्राचे व्हॅल्यू सेट करण्यासाठी वापरले जाते.

एक्सेसर आणि म्युटेटर मेथड्स चे उदाहरण विद्यार्थी. java

```
public class Student
{
    private int roll; private String name;
    public int getRoll() //accessor method
    {
        return roll;
    }
    public void setRoll(int roll) //mutator method
    {
        this.roll = roll;
    }
    public String getName()
    {
        return name;
    }
    public void setName(String name)
    {
        this.name = name;
    }
    public void display()
    {
        System.out.println("Roll no.:" + roll);
        System.out.println("Student name:" + name);
    }
}
```



## JAVA मधील ऑब्स्ट्रक्ट क्लास आणि इंटरफेस (Abstract Classes and Interfaces in JAVA)

**उद्दिष्टे:** या धड्याच्या शेवटी तुम्ही सक्षम व्हाल

- मेथड्स
- इन्हेरिटन्स चे टाइप
- ओवरराइडिंग करण्याचे नियम.

### ऑब्स्ट्रक्ट मेथड्स

ज्या मेथड्स मध्ये मेथड बॉडी नसते ती ऑब्स्ट्रक्ट मेथड्स म्हणून ओळखली जाते. दुस-या शब्दात, अंमलबजावणीशिवाय ऑब्स्ट्रक्ट मेथड्स म्हणून ओळखले जाते. हे नेहमी ऑब्स्ट्रक्ट वर्गात घोषित करते. याचा अर्थ ऑब्स्ट्रक्ट मेथड्स असल्यास क्लास स्वतःच ऑब्स्ट्रक्ट असणे आवश्यक आहे. ऑब्स्ट्रक्ट मेथड तयार करण्यासाठी, आम्ही ऑब्स्ट्रक्ट कीवर्ड वापरतो.

### सिन्टेक्स

```
abstractvoidmethod_name();
Example of abstract
method Demo.java
abstractclass Demo//abstractclass
{
```

### आउटपुट:

ऑब्स्ट्रक्ट मेथड्स ...

### फॅक्टरी मेथड्स

ही एक मेथड्स आहे जी ऑब्जेक्ट ज्या वर्गाशी संबंधित आहे त्याला परत करते. सर्व स्थिर मेथड्स फॅक्टरी मेथड्स आहेत. उदाहरणार्थ, NumberFormatobj = NumberFormat.getNumberInstance();

**इन्हेरिटन्स :** Java मधील इन्हेरिटन्स ही एक यंत्रणा आहे ज्यामध्ये एक ऑब्जेक्ट पॅरेंटऑब्जेक्टचे सर्व प्रॉपर्टीज आणि वर्तन प्राप्त करतो. तो OOPS (ऑब्जेक्ट ओरिएंटेड प्रोग्रामिंग सिस्टम) चा एक महत्त्वाचा पार्ट आहे.

Java मधील इन्हेरिटन्स मागील कल्पना अशी आहे की तुम्ही नवीन क्लास तयार करू शकता जे विद्यमान वर्गावर तयार केले आहेत. जेव्हा तुम्ही अस्तित्वात असलेल्या वर्गाकडून इन्हेरिटन्स घेत असाल, तेव्हा तुम्ही मूळ वर्गाच्या मेथड्स आणि फील्डचा पुनर्वापर करू शकता. शिवाय, तुम्ही तुमच्या सध्याच्या वर्गात नवीन मेथड्स आणि फील्ड देखील जोडू शकता.

इन्हेरिटन्स हे IS-Arelation जहाजाचे प्रतिनिधित्व करते ज्याला पालक-मुलाचे नाते असेही म्हटले जाते.

### जावा मध्ये इन्हेरिटन्स का वापरा

- मेथड ओव्हरराइडिंगसाठी (म्हणून रन टाइम पॉलिमॉर्फिझम मिळवता येईल).

- कोड पुन्हा वापरण्यायोग्यतेसाठी.

### इन्हेरिटन्स मध्ये वापरलेल्या अटी

- **क्लास :** क्लास म्हणजे वस्तूचा समूह ज्यामध्ये सामान्य प्रॉपर्टीज असतात. हे एक टेम्पलेट किंवा ब्लूप्रिंट आहे ज्यातून वस्तू तयार केल्या जातात.
- **सबक्लास /बाल क्लास :** सबक्लास हा एक क्लास आहे जो इतर वर्गाचा इन्हेरिटन्स घेतो. त्याला व्युत्पन्न क्लास , एक्सटेंडेड क्लास किंवा बालक्लास असेही म्हणतात.
- **सुपर क्लास/पालक क्लास :** सुपरक्लास हा क्लास आहे जिथून सबवर्गाला वैशिष्ट्ये इन्हेरिटन्स मिळतात. त्याला बेसक्लास एर एक पालकक्लास असेही म्हणतात.
- **पुन्हा वापरण्यायोग्यता:** नावात नमूद केल्याप्रमाणे, पुनर्वापरयोग्यता ही एक यंत्रणा आहे जी तुम्हाला नवीन क्लास तयार करताना विद्यमान वर्गाची फील्ड आणि मेथड्स पुन्हा वापरण्यास सुलभ करते. तुम्ही मागील वर्गात आधीच परिभाषित केलेल्या समान फील्ड आणि मेथड्स वापरू शकता.

### जावा इन्हेरिटन्सचा सिंटॅक्स

```
classSubclass-nameextends Superclass-name
{
//methodsandfields
}
```

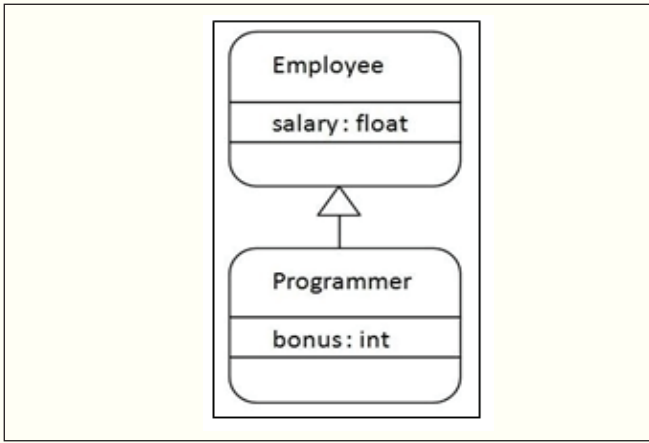
एक्सटेंडेड कीवर्ड सूचित करतो की तुम्ही एक नवीन क्लास बनवत आहात जो विद्यमान वर्गातून प्राप्त होतो. "एक्सपान्शन " चा अर्थ कार्यक्षमता वाढवणे आहे.

जावाच्या परिभाषेत, वारशाने मिळालेल्या वर्गाला पालक किंवा सुपरक्लास म्हणतात आणि नवीन वर्गाला चाइल्डसबक्लास म्हणतात.

### जावा इन्हेरिटन्स

#### उदाहरण

वरील आकृतीमध्ये दाखवल्याप्रमाणे, प्रोग्रामर हा सबक्लास आहे आणि कर्मचारी हा सुपरक्लास आहे. दोन वर्गीकरणप्रोग्रामरISAEmployee मधील संबंध. याचा अर्थ प्रोग्रामर हा एक टाइप चा कर्मचारी असतो.



```

classEmployee
{
floatsalary=40000;
}
class Programmer extends
Employee(intbonus=10000;
publicstaticvoidmain(Stringargs[])
{
Programmer p=new
Programmer();System.out.println("Programmer salary
is:" + p.salary);System.out.print In ("Bonus of Programmeris:
"+p.bonus);
}
}
  
```

**आउटपुट**

प्रोग्रामर पगार

आहे:40000.0Bonusofprogramme

त्याला: 10000

वरील उदाहरणामध्ये, प्रोग्रामर ऑब्जेक्ट स्वतःच्या वर्गाच्या तसेच कर्मचारी वर्गाच्या फील्डमध्ये एक्सेस करू शकतो म्हणजेच कोड पुन्हा वापरता येतो.

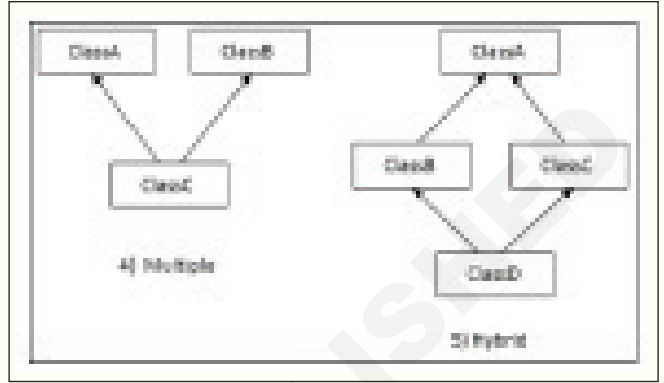
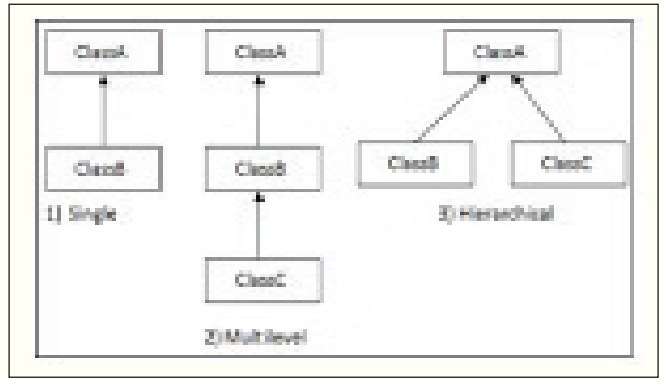
**जावा मधील इन्हेरिटन्स चे टाइप**

वर्गाच्या आधारावर,तीन टाइप असू शकतात इन्हेरिटन्स इंजावा:सिंगल ,बहु-लेव्हलॅड हियरर्किकल.

जावा प्रोग्रामिंगमध्ये, मल्टिपल आणि संकरित इन्हेरिटन्स केवळ इंटरफेसद्वारे समर्थित आहे. इंटरफेसलेटरबद्दल आपण नंतर शिकू.

**टीप:** मल्टिपल इन्हेरिटन्स वर्गाद्वारे जावामध्ये समर्थित नाही.

जेव्हा एखाद्याला अनेक क्लास इन्हेरिटन्स म्हणून ओळखले जातात.



**सिंगल इन्हेरिटन्स उदाहरण**

जेव्हा एखादा क्लास दुसऱ्या वर्गाचा इन्हेरिटन्स घेतो तेव्हा त्याला सिंगल इन्हेरिटन्स म्हणून ओळखले जाते. खाली दिलेल्या उदाहरणात, कुत्रा हा प्राणीवर्गाचा क्लास आहे, सोथेरेस हा सिंगल इन्हेरिटन्स आहे.

File:TestInheritance.java

```

classAnimal{
voideat(){System.out.println("eating...");}
}
class DogextendsAnimal{
voidbark(){System.out.println("barking...");}
}
classTestInheritance{
public static void main(String
args[]){Dog d=newDog();
d.bark();
d.eat();
}}
  
```

**आउटपुट:**

barking...  
eating...

## मल्टिलेव्हल इन्हेरिटन्स उदाहरण

जेव्हा इन्हेरिटन्स ची साखळी असते तेव्हा त्याला मल्टिलेव्हल इन्हेरिटन्स म्हणून ओळखले जाते. तुम्ही खाली दिलेल्या उदाहरणात बघू शकता, बेबीडॉग क्लासला डॉग क्लासचा इन्हेरिटन्स मिळतो जो पुन्हा ॲनिमल क्लासचा इन्हेरिटन्स घेतो, त्यामुळे एक मल्टिलेव्हल इन्हेरिटन्स आहे.

### फाइल: TestInheritance2.java

```
classAnimal{
    voideat(){System.out.println("eating...");
}
class DogextendsAnimal{
    voidbark(){System.out.println("barking...");
}
classBabyDog extendsDog{
    voidweep(){System.out.println("weeping...");
}
classTestInheritance2{
    public static void main(String
    args[]){BabyDog d=new
    BabyDog();d.weep();
    d.bark();
    d.eat();
}}
```

आउटपुट: रडणे... भुंकणे... खाणे...

## रेन्ज बद्ध (हायरार्कीकल) इन्हेरिटन्स उदाहरण

जेव्हा दोन किंवा अधिक क्लास एकच क्लास इन्हेरिटन्स घेतात तेव्हा त्याला रेन्ज बद्ध (हायरार्कीकल) इन्हेरिटन्स म्हणून ओळखले जाते. खाली दिलेल्या उदाहरणात, डोगांड cat क्लास प्राणी वर्गाचा इन्हेरिटन्स घेतात, म्हणून रेन्ज बद्ध (हायरार्कीकल) इन्हेरिटन्स आहे.

### फाइल:TestInheritance3.java

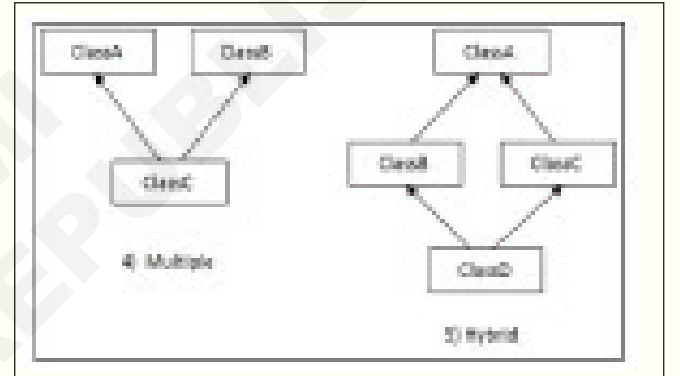
```
classAnimal{
    voideat(){System.out.println("eating...");
}
class DogextendsAnimal{
    voidbark(){System.out.println("barking...");
}
}
```

```
classCatextendsAnimal{
    voidmeow(){System.out.println("meowing...");
}
classTestInheritance3{
    public static void main(String
    args[]){Catc=newCat();
    c.meow();
    c.eat();
    //c.bark();//C.T.Error
}}
```

### आउटपुट:

meowing...

eating...



### जावा मध्ये मल्टिपल इन्हेरिटन्स का समर्थित नाही?

जटिलता कमी करा आणि भाषा सोपी करा, मल्टिपल इन्हेरिटन्स जावामध्ये समर्थित नाही.

A, B, आणि C हे तीन क्लास आहेत अशा परिस्थितीचा विचार करा. C वर्गाला A आणि B वर्गांचा इन्हेरिटन्स मिळतो. जर A आणि B क्लास एकच मेथड्स असेल आणि तुम्ही ती चाइल्ड क्लास ऑब्जेक्टवरून कॉल केली असेल, तर A or B क्लासची मेथड्स म्हणण्यात संदिग्धता असेल.

कंपाइल-टाइम एरर रनटाइम एररसंपेक्षा चांगल्या असल्याने, जर तुम्हाला 2 क्लासेस इनहेरिट केले तर Java कंपाइल-टाइम एरर रेंडर करते. तुमच्याकडे समान मेथड्स वेगळी असली तरीही, कंपाइल टाइम एरर असेल.

```
classA{
    voidmsg(){System.out.println("Hello");
}
class B{
    voidmsg(){System.out.println("Welcome");
}
```

```

}
classCextendsA,B{//suppose ifitwere
public static void main(String args[]){Cobj=newC();
obj.msg();//Nowwhichmsg()methodwouldbeinvoked?
}
}

```

### कंपाइल टाइम एरर

#### जावा मध्ये ओव्हर राइडिंगची मेथड्स

जर सबक्लास (चाइल्ड क्लास) मध्ये पॅरेंट क्लासमध्ये घोषित केलेली मेथड्स समान असेल, तर ती मेथड ओव्हरराइडिंग इन जावा म्हणून ओळखली जाते.

दुसऱ्या शब्दात, जर सबक्लास त्याच्या मूळ वर्गापैकी एकाने घोषित केलेल्या मेथड्स ची स्पेसिफिक अंमलबजावणी प्रदान करत असेल, तर ती मेथड्स ओव्हरराइडिंग म्हणून ओळखली जाते.

#### जावा मेथड ओव्हरराइडिंगचा वापर

- मेथड ओव्हरराइडिंगचा वापर एखाद्या मेथड्स ची स्पेसिफिक अंमलबजावणी प्रदान करण्यासाठी केला जातो जी त्याच्या सुपर क्लासने आधीच प्रदान केली आहे.
- रनटाइम पॉलिमॉर्फिझमसाठी ओव्हर राइडिंग मेथड्स वापरली जाते

#### जावा मेथड ओव्हरराइडिंगसाठी नियम

- 1 थीमथमथवेच नाव पालकवर्गात
- 2 थीममध्ये पॅरेंट क्लास प्रमाणेच पॅरामीटर असणे आवश्यक आहे.
- 3 There must be an IS-संबंध (इन्हेरिटन्स).

मेथड्स ओव्हरराइडिंगशिवाय समस्या समजून घेणे

जर आपण मेथड ओव्हरराइडिंगचा वापर केला नाही तर प्रोग्राममध्ये आपल्याला कोणत्या समस्या येऊ शकतात ते समजून घेऊया.

```

//JavaProgramtodemonstratwhyweneedmethodoverriding
//Here,weare calling themethodof parentclasswithchild
//classobject.
//Creating a parent classclassVehicle{
voidrun(){System.out.println("Vehicleisrunning");
}
//Creating a child classclassBikeextendsVehicle{
publicstaticvoidmain(Stringargs[]){
//creating an instance of child classBike obj =newBike();
//calling the method with child class instanceobj.run();
}
}

```

```

}
}
}

```

### आउटपुट:

#### वाहने धावत आहेत

समस्या अशी आहे की मला सबक्लासमध्ये रन() मेथड्स ची स्पेसिफिक अंमलबजावणी प्रदान करावी लागेल म्हणून आम्ही मेथड ओव्हरराइडिंग वापरतो.

#### ओव्हरराइडिंग मेथड्स चे उदाहरण

या उदाहरणात, आम्ही मूळ वर्गात परिभाषित केल्याप्रमाणे सबक्लासमध्ये रन मेथड्स परिभाषित केली आहे परंतु त्याची काही स्पेसिफिक अंमलबजावणी आहे. मेथड्स चे नाव आणि पॅरामीटर समान आहेत, आणि वर्गामध्ये IS-A संबंध आहे, sothereismethodoverriding.

जावा मेथड ओव्हरराइडिंगचा वापर स्पष्ट करण्यासाठी // JavaProgram

#### जावा मध्ये मेथड ओव्हरराइडिंग

##### MethodOverriding inJava

```

//Creating a parent class.
classVehicle{
//definingamethod
voidrun(){System.out.println("Vehicleisrunning");
}
//CreatingachildclassclassBike2extendsVehicle{
//definingthesamemethodas inthe parentclass
voidrun(){System.out.println("Bikeisrunningsafely");}
public static void main(String
args[]){Bike2 obj = new
Bike2();//creating
objectobj.run();//callingmethod
}
}
}

```

### आउटपुट:

दुचाकी सुरक्षितपणे धावत आहे

JavaMethodOverriding चे एरिया उदाहरण

अशा परिस्थितीचा विचार करा जिथे बँक हा एक क्लास आहे जो व्याजदर मिळविण्यासाठी कार्यक्षमता प्रदान करतो. तथापि, व्याजदर बँकांनुसार बदलतात. उदाहरणार्थ, SBI, ICICI आणि AXIS बँका ८%, ७% आणि ९% व्याज देऊ शकतात.

**टीप:** रनटाइम पॉलिमॉर्फिझममध्ये जावामेथडओव्हरराइडिंगचा वापर केला जातो जो आपण पुढील पेजवर शिकू.

**max**

```
public static int max(int a,  
                      int b)
```

//Java Program to demonstrate the real scenario of Java  
Method Overriding

//where three classes are overriding the method of a parent  
class.

//Creating a parent

```
class Bank{  
    int getRateOfInterest(){return 0;}  
}
```

//Creating child

```
class SBI extends Bank{  
    int getRateOfInterest(){return 8;}  
}
```

```
class ICICI extends Bank{  
    int getRateOfInterest(){return 7;}  
}
```

```
class AXIS extends Bank{  
    int getRateOfInterest(){return 9;}  
}
```

//Test class to create objects and call the

```
methods class Test2{
```

```
    public static void main(String
```

```
    args[]){SBI s=new SBI();
```

```
    ICICI i=new
```

```
    ICICI();AXIS a=new AXIS();
```

```
    System.out.println("SBI Rate of Interest:
```

```
    "+s.getRateOfInterest());System.out.println("ICICI Rate of  
    Interest:
```

```
    "+i.getRateOfInterest()); System.out.println ("AXISRate of  
    Interest:"+a.getRateOfInterest());
```

आउटपुट:

SBI व्याज दर: 8 ICICI व्याज दर: 7 AXISRate व्याज: 9

**आपण ओवरराइडस्टॅटिक मेथड्स करू शकता?**

नाही, स्टॅटिक मेथड्स ओवरराइडिंग करता येत नाही. हे रनटाइम पॉलिमॉर्फिझमद्वारे सिद्ध केले जाऊ शकते, म्हणून आपण ते नंतर शिकू.

**ओवरराइडस्टॅटिक मेथड्स का करू शकता?**

कारण स्टॅटिक मेथड्स वर्गाशी बांधलेली असते तर उदाहरण मेथड्स ऑब्जेक्टशी बांधलेली असते. स्टॅटिक क्लास क्षेत्राशी संबंधित आहे, आणि एक उदाहरण हीप क्षेत्राशी संबंधित आहे.

**आपण जावा मुख्य मेथड्स ओवरराइड करू शकतो का?**

नाही, कारण मुख्य ही एक स्थिर मेथड्स आहे.

**जावा मधील मेथड ओवरलोडिंग आणि मेथड ओवर राइडिंग मधील फरक**

मेथड ओवरलोडिंग आणि ओवरराइडिंग मधील फरकासाठी मला क्लिक करा More topics on Method Overriding (Not For Beginners)

ऍक्सेस मॉडिफायरसह मेथडओवरराइडिंग

ऍक्सेस मॉडिफायरसह मेथड ओवरराइडिंगची संकल्पना पाहू.

Exception Handling with Method Overriding

अपवाद हाताळणीसह मेथड्स ओवरराइडिंगची संकल्पना पाहू.