**1. INTRODUCTION :**

An operating system (OS) is a software program designed to act as an interface between a user and a computer. It controls the computer hardware, manages system resources and supervise the interaction between the system and its user. The operating system also forms a base on which application software are developed and executed.

**Functions of an Operating System:**

An Operating system is expected to perform various functions

These functions include

1. Command interpretation

2. Peripheral management

3. Memory management

4. Process management

**TYPES OF OPERATING SYSTEM:**

**Single-user operating system**-DOS is the most popular example of a single OS

**Multi-user operating system** -UNIX is the most popular example of a multi-user OS.

* **Unix operating system-** Unix is a multiuser, multiprocessing, portable operating system. It is designed to facilitate Programming, text processing and communication

**COMPONENTS OF UNIX**

The unix operation system consists of three main software components as shown below:

**Applications programs**

|  |
| --- |
| **Shell** |
| **Kernel** |
| **Hardware** |

**Kernel**: It directly interacts with the hardware and controls the computer resources. It starts up a separate, interactive program, called for each user.

**Unix utilities**: Unix utilities or commands are a collection of about 200 program that service the day-today processing requirement. These programs are invoked are through the shell, which is itself another utility.

**Application software**: Unix-based application program, like Database management system, word processors, accounting software and language processors, are available from independent software vendors.

**Features and Benefits of UNIX:**

→Portability

→Background Processing

→Machine-independence

→UNIX Shell

→Multi-user Operations

→Maturity

→Hierarchical File System

→Pipes and Filters

→Utilities

→Software Development Tools

The UNIX system supports a wide variety of languages -C, FORTRAN, BASIC, PASCAL, ADA, COBAL, lisp and prolog.

**Types of UNIX Operating System:**

|  |
| --- |
| **UNIC**  **1969** |
| **FIFTH EDITION**  **1973** |
| **SIXTH EDITION**  **1976** |
| **SEVENTH EDITION**  **1978** |

**Shell**: The shell is a utility program that comes with the UNIX System. However, it plays a very important role. It acts as the command interpreter for the kernel and is the interface between the user and the kernel.

The UNIX system's shell includes the following major features:

→Interactive processing

→Background processing

→Input/output Redirection

→Pipes

→Shell scripts

→Programming Language Constructs

**Types of shells**

1. Bourne shell

2. C Shell

3. Korn Shell

4. Restricted Shell

5. Bash Shell

**2. INTRODUCTION**

Linux was developed by Linux Torvalds in the year 1991. Torvalds, a university student is the father of Linux. He made the entire software available on the Internet. Linux is particularly strong in networking and internet features. The distributions, which are shipped on multiple CR ROMs, include a plethora of software -from C and C++ compilers to java, browsers like Netscape, Web servers, Internet servers, proxy servers and multimedia software.

**LINUX COMMANDS**

There are many commands available in Linux. These can be divided into 10 common categories that are as follows

**File-Handling commands:** cat, cp, rm, my, more, less, lpr, file, wc, split, cmp, comm, diff, chown, chgrp, chmod, touch, in, find, tee.

**Directory-handling commands:** pwd, mkdir, cd, rmdir, ls.

**General Purpose Utilities:** cal, date, tty, who, uname, passwd, echo, bc,time, script.

**Filters:** pr, head, tail, cut, paste, sort, uniq, nl, tr, grep.

**Process-related Commands:** ps, kill, bg, fg, wait & jobs.

**Communication and E-mail:** write,talk, mail, mesg, finger.

**Environment:** set, alias, history.

**Compression and archiving:** gzip, gunzip,tar,zip,unzip.

**Networking:** ftp, telnet.

**System Administration:** shutdown, wall, df, du, dd, whereis, mount, ping.

**Alias -** assign short hand names for commands you may be using quite frequently ex.$ alias 1='is=1

**Bc**- it is a calculator {$ bc}

**Bg:** it is used to push the last job to the background.

**Fg**: Any of the background jobs can be brought to the foreground.

**Kill:** It is used to terminate a process.

**Chown:** change ownership of a process,

**Chgrp**: change the group ownership of a file.

**Ln**: links.

**Exit:** it is used to terminate a program.

**Ps:** used to display the attributes of a process.

**Write:** is lets you have a two-way communication.

**Mesg n:** command used to prevent other users from sending messages to you.

**Mesg y:** command used to allow other users to send messages to you.

**Talk:** differentiate between send &received messages.

**Mail**: enable sending of mail to a user.

**Pine:** program for internet news & e-mail.

**Finger:** details of user. Passwd: changing password.

**Wall:** it is used by admin. It addresses all user simultaneously.

**Shutdown:** used to shut down the machine.

**Df(disk free):** separating free space available in the disk.

**Du:** disk usage.

**Gzip:** compresses files, have the extension .gz

**Gunzip:** unzip or uncompressed.gz file.

**Tar:** creates archive on tapes.

**telnet(terminal network):** used to connect to a remote UNIX system.

**ftp(File Transfer protocol):** used to transfer both binary & text files.

**Filters:** the cmds are pr, head, tail, cut, paste, uniq, nl, tr, sort, grep.

**Pr-cmd:** prepares a file for printing

**Head:** it displays the top of the file.

**Tail:** if displays the end of the file.

**Cut:** used to slice a file vertically

**Paste:** use to concatenation vertically.

**Sort:** sort performs sorting functions.

**Uniq:** eliminate duplicate entries.

**N1**: used for numbering lines.

**Tr(translate):** it manipulates individual characters in a file.

**greap(globally search for regular expression & print out):** It scans afile for the occurrence of a pattern & can display the selected pattern.

**Regular expression:** if an expression uses any of the following characters,it is termed a regular expression. The characters are \*.?^

**File:** determine the type of file.

**More:** allows the user to view a file, one screen at a time.

**Echo:** display its arguments.

**Clear:** used to clear your screen.

**History:** command lists out most recently used commands.

**Pwd:** print working directory.

**Find:** it examines a dir tree to work for files matching some criteria.

**Wait:** checks whether all background processes have been completed.

**Zip:** it compresses multiple files into a single file.

**archive:** a file containing a group of files.

**Unzip:** is used for decompression.

**Comm:** compare two sorted files.

**Cmp:** compare two files.

**Diff:** used to display file differences.

**Who:** displays all the users.

**Who am i:** displays the users login name & terminal name&date.

**Cat:** create a new file.

**Cp:** copy the contents of file1.a into a file2.b.

**Rm:** remove the file or directory.

**Mv:** rename a file.

**Ls:** list the contents of the directory.

**Cal:** displays the calendar

**VIM (VISUAL MODIFIED) EDITOR:**

Vi is a text editor released by Bram Moolenar in 1991 for the Amiga computer. The name "Vi" is an acronym for "Vi Improved" because Vim was created as an extended version of the vi editor, whit many additional features designed to be helpful in editing program source code.

**SHELL PROGRAMMING**

A group of commands that is to be executed together can be better stored in a file called as shell scripts shell programs, or shell procedures.

It is convention to use .sh extension for shell script

$cat script.sh -or vi script.sh

‘vi’ is the editor script filename.

‘sh’ is the extension name stands for the shell program.

To Execute

$ sh script.sh (or) $ bash script.sh (or) $./script.sh

**Exit-**script termination.

**expr-** It is used to perform arithmetic operation on integers.

**What is unix :** A unix is a multiuser and a multi-processor operating system.

**Give few examples of multi-user system** : Unix, Xeneix, VMS – virtual machine system

**Who developed unix :** Unix was developed by ken Thomson of bell laboratories in 1969

**From where unix is evolved :** Unix is evolved from multies because it is well-known time-sharing OS.

**What is multi user :** A multi user system permits several users to use the same computer simultaneously

**What are different types of shells :**

Bourne shell – extension – bash

Kern shell – extension – ksh

Visual – extension – vsh

Restricted shell – extension – rsh

C shell – extension – csh

**What is Telnet:**

Telnet is called terminal network emulator. It works with the help of IP address (Internet protocol)160.111.123.145

**What is date command:** Date is general purpose command to display the current date with time

%n - name of the month

%H - hour display

%m - month

%d - day of month

%y - display last two digits of the Year

%m - minutes

%s - seconds

%a - display day of the week

**What is who am i :** It displays respective username and its terminal number The Who command the information about all the user who are currently working in the terminal/server/network **What is a shell:** The shell act as a command interpreter waits for the command input to be given by the user. it is one layer about the kernel

**What is kernel called as :** It a heart of operating system. It acts as an interface between the shell and CPU

**What is echo:** To display the line of text we use echo

**Mention some wild characters used in Unix :** Wild characters used in Unix “%.?,!, \*”

**List the important features of Unix**

Time sharing

Multi user multitasking

Portability

Modularity

System security

Communication

File security

**Functions of kernel**

Memory management

File management

Input output management

Storage management

Entrepreneur and error handling

Process management

**ls\* -** this command is used to list all the related files present in the directory

**esc-** Make a us come out of the program

**:W -** writing to the disk furtherly early to save the program which displays on the last status line of the program

**:q -** exit to exit out of the program

**.x -** used to delete a character inside the file

**dd -** used to delete a line inside the file

**clear** - to clear the screen

**cat -** display the content of the file cal - display the date (month and year)

**tty -** show the name of the terminal

**stty -** used to reset the terminal when it is not responding properly

**-a & -g -** are the argument format which are human uaclable

**ctrl+c -** use it to kill the data

ctrl+k - used to erase the data

**bc -** basic calculator

**ctrl+cl-** exit from calculator

**i -** insert text before the current character

**I -** insert text at the beginning of current line

**wq&zz -** save the content to the buffer and exit

**:q-** exit without saving

**$ls** -l - long listing of the files

**$ls -** list all the files and directories in a alphabetical order

**$ls -a** - display all the files and directories along with hidden files

**$ls -d** - display Only working directory

**$ls -n -** display user id and group ID

**$ls -r -** display in in reverse order from z to a

**TIME SORTING**

1. Basic time sorting(-lt) .

Shot by stamp with latest file first

1. Second time sorting(-lu) .

Last working access files

1. Start time sorting(-lc) .

List the files by the date change

**Identify directory** : $ls -p

**Rename a file :**mv.file1 file2

**Shell programming :** It is a file that content shell command which performs a useful function which is also known as Shell Script

**Bourne shell :** Introduce by Steve bourne in AT&T laboratory. it is called bourne against sell the latest version of bourne shell is Bash shell

**C shell :** Introduced by bill Joy berkely laboratory it is very compatible shell in Linux

**Kernel shell :** Introduced by David kome in AT&T laboratory which is a a new version and mash powerful

**Two major parts of shells**

**(a).Interrupter** - it reads your command works with kernel to execute them

**(b).Program capability** - allows you to write shell script or Shell programming

**Moving from 1 shell to another shell**: $bash/ksh/csh #move to bash /come /c shell

• Unix define the standard

• That is used by command

• standard input, standard output, standard error

**Command Execution:** we can combine for 4 syntactical commands

1 group commands

2 chained commands

3 conditional commands

4 Sequenced commands

**Group command**

Commands are grouped by placing them in parenthesis( {}, (), [] )Ex- angular brackets $((n%2))

**Chained commands**

The methods of combining commands with the help of pipe line symbol,Ex- C= `expr $a+$b|bc`

**Conditional command**

Combine 2 or more command using conditional relations,Ex- [$Avg -ge70 ]&& [$Avg -lt 99]

**Sequential command**

Sequence of commands on one like its comment must be separated from it is preceded by relation between comments,Ex- $echo “\n welcome to vvism\n”

**Adding text with basic “vi” commands**

a-appends text after the current character

A-adds text at the end of the current line

x- delete the current character

h- moves cursor One character left

l- moves cursor One character right

o- moves cursor to the beginning of current line

k- moves cursor one line up

j- moves Cursed one line down

- - moves cursor to the beginning of previous line

+ - moves cursor to the beginning of the next line

u- undo only last edit

v- undo all the changes on the current line

s- moves cursor to the end of the current line

# LINUX PROGRAMMING ENVIRONMENT

## LINUX & UNIX:

**LINUX** and **UNIX** are known as 2 comparable and robust [operating systems](https://ipwithease.com/what-is-operating-system-and-its-functions/). In fact, Linux is considered a very close cousin of UNIX OS.

* While LINUX and UNIX have quite a few similarities, still there is plenty of difference between both.
* **UNIX** system began in the year 1969 when [AT&T](https://en.wikipedia.org/wiki/AT%26T) developed the version of the UNIX operating system, which was completely written in C language.
* On the other hand, **LINUX** is quite a newbie to the IT market since it came to inception in the year 1991.
* In term of standards, **LINUX** is an Open-Source OS whereas **UNIX** is only limited to its copywriters.
* When LINUX was developed, the intent was to spread the product equally amongst home users, developers etc. and UNIX was developed mainly for servers, workstations and mainframes.
* Linux is a Unix clone, behaves like Unix but doesn't contain its code. Unix contain a completely different coding developed by AT&T Labs. Linux is just the kernel. Unix is a complete package of Operating system.
* The Linux kernel is the main component of a Linux operating system (OS) and is the core interface between a computer's hardware and its processes.

## UBUNTU:

**Ubuntu** is a Linux distribution and composed mostly of free and open-source software.

**Initial release**: Ubuntu 4.10 (Warty Warthog) / 20 October 2004 (17 years ago)

**Latest release**: Ubuntu 22.04 (Jammy Jellyfish) / 21 April 2022

## vi Fiber:

### What is vi?

The vi editor is elaborated as **vi**sual editor. It is installed in every Unix system. In other words, it is available in all Linux distros. It is user-friendly and works same on different distros and platforms. It is a very powerful application. An improved version of vi editor is **vim**.

### The vi editor has two modes:

* **Command Mode:** In command mode, actions are taken on the file. The vi editor starts in command mode. Here, the typed words will act as commands in vi editor. To pass a command, you need to be in command mode.
* **Insert Mode:** In insert mode, entered text will be inserted into the file. The **Esc** key will take you to the command mode from insert mode.

By default, the vi editor starts in command mode. To enter text, you have to be in insert mode, just type **'i'** and you'll be in insert mode. Although, after typing **i** nothing will appear on the screen but you'll be in insert mode. Now you can type anything.

To exit from insert mode press **Esc** key, you'll be directed to command mode. Using vi

The vi-editor tool is an interactive tool as it displays changes made in the file on the screen

while you edit the file.

In vi editor you can insert, edit or remove a word as cursor moves throughout the file. Commands are specified for each function like to delete its x or dd.

The vi editor is case-sensitive. For example, **p** allows you to paste after the current line while **P** allows you to paste before the current line.

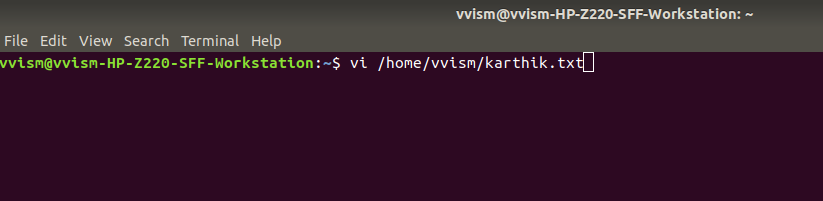
## Linux vi example

To start vi, open your terminal and type vi command followed by file name. If your file is in some other directory, you can specify the file path. And if in case, your file doesn't exist, it will create a new file with the specified name at the given location.

### Example:

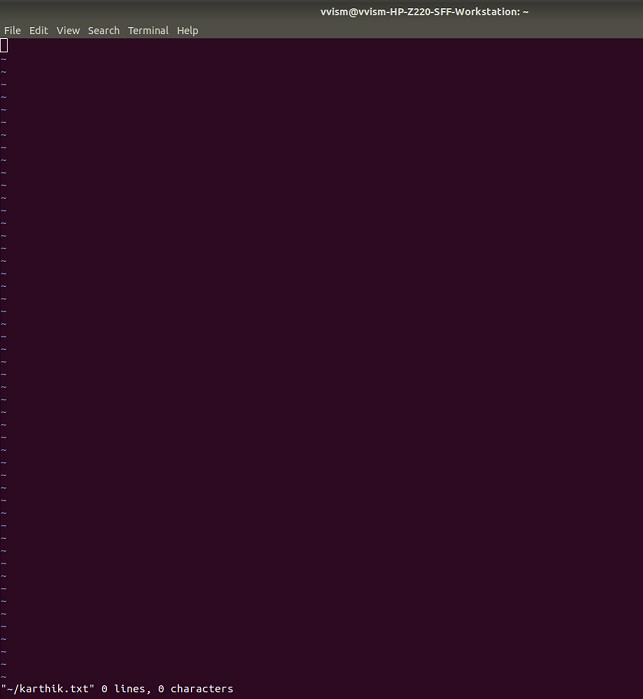
1. vi /home/vvism/karthik.txt

Look at the above snapshot, we are creating a new file **karthik.txt** (as this file doesn't exist) and have entered the full path for the directory **vvism**



## Command mode

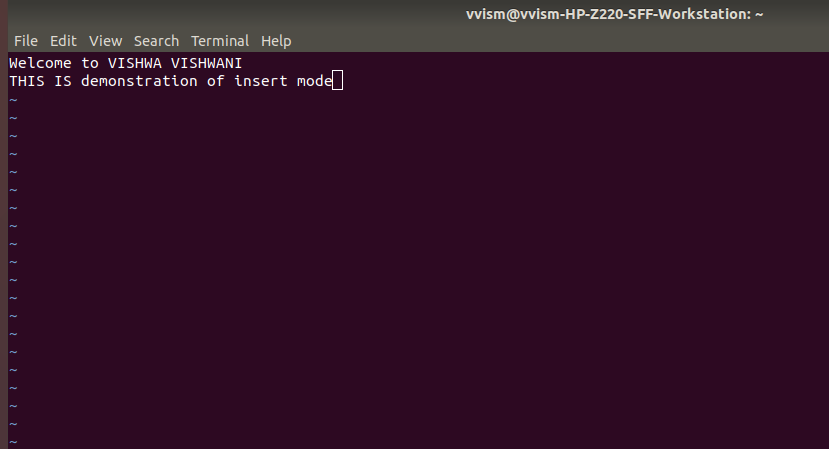
This is what you'll see when you'll press enter after the above command. If you'll start typing, nothing will appear as you are in command mode. By default vi opens in command mode.



Look at the above snapshot, it is blank as it is a new file. To start typing, you have to move to the insert mode. At the end of the terminal window, directory name and file name are displayed.

## Insert mode

To move to the insert mode press **i.** Although, there are other commands also to move to insert mode which we'll study in next page.



Look at the above snapshot, after pressing **i** we have entered into insert mode. Now we can write anything. To move to the next line press, enter.

Once you have done with your typing, press **esc** key to return to the command mode.

## To Save and Quit

You can save and quit vi editor from command mode. Before writing save or quit command you have to press colon **(:).** Colon allows you to give instructions to vi

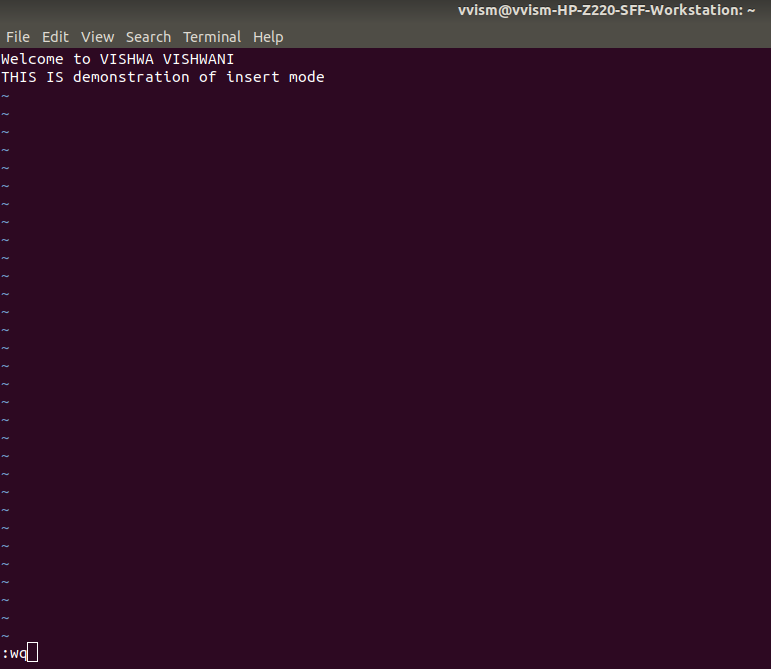
**exit vi table:**

| **Commands** | **Action** |
| --- | --- |
| :wq | Save and quit |
| :w | Save |
| :q | Quit |

| :w fname | Save as fname |
| --- | --- |
| ZZ | Save and quit |
| :q! | Quit discarding changes made |
| :w! | Save (and write to non-writable file) |

To exit from vi, first ensure that you are in command mode. Now, type :wq and press enter. It will save and quit vi.

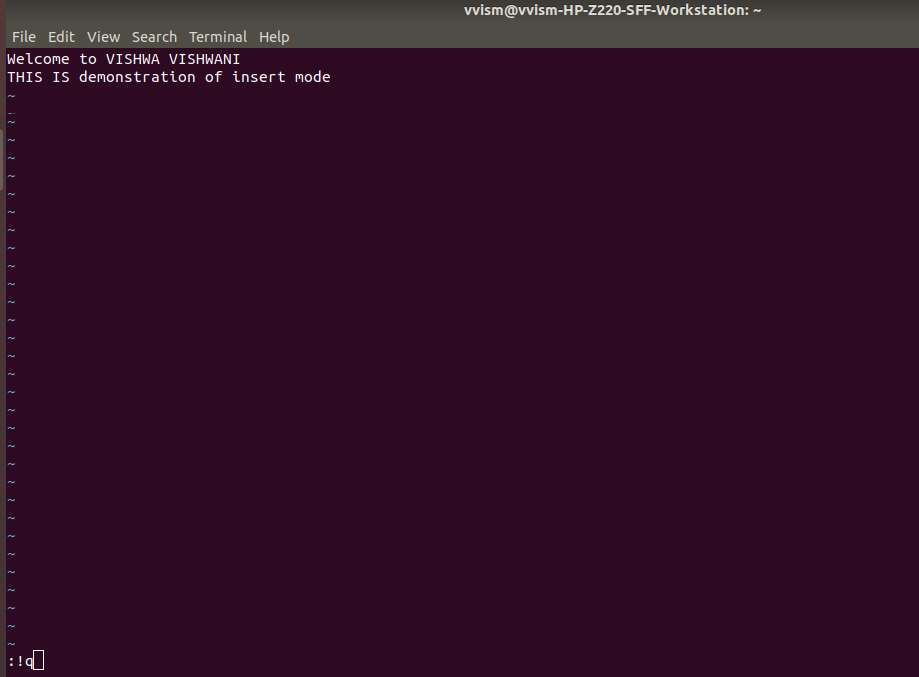
Type: wq to save and exit the file.



Look at the above snapshot, command: wq will save and quit the vi editor. When you'll type it in command mode, it will automatically come at bottom left corner.

If you want to quit without saving the file, use:**q.** This command will only work when you have not made any changes in the file.

The above file can be saved with the command: q. It discards the changes made in the file and save it.



Look at the above snapshot, we have typed**:!q,** it will save our file by discarding the changes made.

# Vi Commands

Linux vi editor is different from other editors. You have to use different keys to use different functions. Although, it's quite easy and interesting to use vi editor.

The vi editor commands are case sensitive.

Have a look at the vi commands in the following table.

## To switch from command to insert mode:

| **Command** | **Action** |
| --- | --- |
| i | Start typing before the current character |
| I | Start typing at the start of current line |
| a | Start typing after the current character |
| A | Start typing at the end of current line |
| o | Start typing on a new line after the current line |
| O | Start typing on a new line before the current line |

**To move around a file:**

| **Commands** | **Action** |
| --- | --- |
| J | To move down |
| K | To move up |
| H | To move left |
| L | To move right |

## To jump lines:

| **Commands** | **Action** |
| --- | --- |
| G | Will direct you at the last line of the file |
| `` | Will direct you to your last position in the file |

**To delete:**

| **Commands** | **Action** |
| --- | --- |
| x | Delete the current character |
| X | Delete the character before the cursor |
| r | Replace the current character |
| xp | Switch two characters |
| dd | Delete the current line |
| D | Delete the current line from current character to the end of the line |
| dG | delete from the current line to the end of the file |

## To repeat and undo:

| **Commands** | **Action** |
| --- | --- |
| u | Undo the last command |
| . | Repeat the last command |

**Command to cut, copy and paste:**

| **Commands** | **Action** |
| --- | --- |
| dd | Delete a line |
| yy | (yank yank) copy a line |
| p | Paste after the current line |
| P | Paste before the current line |

### Shell Script:

A shell script is a computer program designed to be run by the Unix/Linux shell which could be one of the following

* The Bourne Shell
* The C Shell
* The Korn Shell
* The GNU Bourne-Again Shell

### Shell:

A shell is a command-line interpreter and typical operations performed by shell scripts include file manipulation, program execution, and printing text.

### Extended Shell Scripts:

Shell scripts have several required constructs that tell the shell environment what to do and when to do it. Of course, most scripts are more complex than the above one.

The shell is, after all, a real programming language, complete with variables, control structures, and so forth. No matter how complicated a script gets, it is still just a list of commands executed sequentially.

The following script uses the read command which takes the input from the keyboard and assigns it as the value of the variable PERSON and finally prints it on STDOUT(Standard Output).

$ vi test.sh

**Variable**:

A variable is a character string to which we assign a value. The value assigned could be a number, text, filename, device, or any other type of data.

For example, first we set a variable TEST and then we access its value using the echo command

It produces the following result.

Unix Programming

Note that the environment variables are set without using the $ sign but while accessing them we use the $ sign as prefix. These variables retain their values until we come out of the shell.

When you log in to the system, the shell undergoes a phase called initialization to set up the environment.

The shell displays a prompt –

$

This is the prompt where you can enter commands in order to have them executed.

# LINUX BASIC COMMANDS

## pwd command

Use the **pwd** command to find out the path of the current working directory (folder) you’re in. The command will return an absolute (full) path, which is basically a path of all the directories that starts with a forward slash (/). An example of an absolute path is /home/username.

## cd command

To navigate through the Linux files and directories, use the cd command

* + **cd..** (With two dots) to move one directory up
  + cd to go straight to the home folder
  + cd- (with a hyphen) to move to your previous directory

On a side note, Linux’s shell is case sensitive. So, you have to type the name’s directory exactly as it is.

## ls command

The ls command is used to view the contents of a directory. By default, this command will display the contents of your current working directory.

If you want to see the content of other directories, type ls and then the directory’s path. For example, enter **ls /home/username/Documents** to view the content of Documents. There are variations you can use with the ls command:

* + **ls -R** will list all the files in the sub-directories as well
  + **ls -a** will show the hidden files
  + **ls -al** will list the files and directories with detailed information like the permissions, size, owner, etc.

## [cat command](https://www.hostinger.in/tutorials/linux-cat-command-tutorial-and-examples/)

cat (short for concatenate) is one of the most frequently used commands in Linux. It is used to list the contents of a file on the standard output (sdout). To run this command, type cat followed by the file’s name and its extension.

For instance: **cat file.txt**.

Here are other ways to use the cat command:

* + cat > filename creates a new file
  + cat filename1 filename2>filename3 joins two files (1 and 2) and stores the output of them in a new file (3)

## 5.cp command

Use the cp command to copy files from the current directory to a different directory. For instance, the command **cp scenery.jpg /home/username/Pictures** would create a copy of scenery.jpg (from your current directory) into the Pictures directory.

**6.**[**mv command**](https://www.hostinger.in/tutorials/how-to-rename-files-in-linux/)

The primary use of the mv command is to move files, although it can also be used to rename files.The arguments in mv are similar to the cp command. You need to type mv, the file’s name, and the destination’s directory.

For example: **mv file.txt /home/username/Documents**.

To rename files, the Linux command is mv oldname.ext newname.ext

## 7.mkdir command

Use mkdir command to make a new directory — if you type **mkdir Music** it will create a directory called Music.

There are extra mkdir commands as well:

* + To generate a new directory inside another directory, use this Linux basic command mkdir Music/Newfile
  + use the p (parents) option to create a directory in between two existing directories. For example, mkdir -p Music/2020/Newfile will create the new “2020” file.

**8.**[**rmdir command**](https://www.hostinger.in/tutorials/how-to-remove-files-and-folders-using-linux-command-line/)

If you need to delete a directory, use the **rmdir** command. However, rmdir only allows you to delete empty directories.

## 9.rm command

The rm command is used to delete directories and the contents within them. If you only want to delete the directory — as an alternative to rmdir — use rm -r**.**

Note: Be very careful with this command and double-check which directory you are in. This will delete everything and there is no undo.

## 

## 10.[touch command](https://www.hostinger.in/tutorials/linux-touch-command-with-useful-examples/)

The touch command allows you to create a blank new file through the Linux command line. As an example, enter **touch /home/username/Documents/Web.html** to create an HTML file entitled Web under the Documents directory.

**11.locate command**

You can use this command to locate a file, just like the search command in Windows. What’s more, using the -i argument along with this command will make it case- insensitive, so you can search for a file even if you don’t remember its exact name.

To search for a file that contains two or more words, use an asterisk (\*). For

example, **locate -i school\*note** command will search for dany file that contains the word “school” and “note”, whether it is uppercase or lowercase.

## 12.[find command](https://www.hostinger.in/tutorials/how-to-use-find-and-locate-commands-in-linux/)

Similar to the locate command, using find also searches for files and directories. The difference is, you use the find command to locate files within a given directory.

As an example, **find /home/ -name notes.txt** command will search for a file called notes.txt within the home directory and its subdirectories.

Other variations when using the find are:

* + To find files in the current directory use, find . -name notes.txt
  + To look for directories use, / -type d -name notes. txt

## 13.[grep command](https://www.hostinger.in/tutorials/grep-command-in-linux-useful-examples/)

Another basic Linux command that is undoubtedly helpful for everyday use is grep. It lets you search through all the text in a given file.

To illustrate, **grep blue notepad.txt** will search for the word blue in the notepad file. Lines that contain the searched word will be displayed fully.

## 14.df command

Use df command to get a report on the system’s disk space usage, shown in percentage and KBs. If you want to see the report in megabytes, type **df -m**.

## 15.[du command](https://www.hostinger.in/tutorials/vps/how-to-check-and-manage-disk-space-via-terminal)

If you want to check how much space a file or a directory takes, the du (Disk Usage) command is the answer. However, the disk usage summary will show disk block

numbers instead of the usual size format. If you want to see it in bytes, kilobytes, and megabytes, add the -h argument to the command line. Example **du - h**

## 16.head command

The head command is used to view the first lines of any text file. By default, it will show the first ten lines, but you can change this number to your liking. For example, if you only want to show the first five lines, type **head -n 5 filename.ext**.

## 17.[tail command](https://www.hostinger.in/tutorials/how-to-use-tail-command/)

This one has a similar function to the head command, but instead of showing the first lines, the tail command will display the last ten lines of a text file.

For example, **tail -n filename.ext**.

## 18.diff command

Short for difference, the diff command compares the contents of two files line by line. After analyzing the files, it will output the lines that do not match. Programmers often use this command when they need to make program alterations instead of rewriting the entire source code. The simplest form of this command is **diff file1.ext file2.ext**

## 19.[chmod command](https://www.hostinger.in/tutorials/vps/change-linux-permissions-and-owners)

chmod is another Linux command, used to change the read, write, and execute permissions of files and directories

## 20.[chown command](https://www.hostinger.in/tutorials/linux-chown-command/)

In Linux, all files are owned by a specific user. The chown command enables you to change or transfer the ownership of a file to the specified username. For instance, **chown linuxuser2 file.ext** will make linuxuser2 as the owner of the file.ext.

## 21.jobs command

jobs command will display all current jobs along with their statuses. A job is basically a process that is started by the shell.

## 22.[ping command](https://www.hostinger.in/tutorials/linux-ping-command-with-examples/)

Use the ping command to check your connectivity status to a server. For example, by simply entering **ping google.com**, the command will check whether you’re able to connect to Google and also measure the response time.

## 23.wget command

The Linux command line is super useful — you can even download files from the internet with the help of the wget command. To do so, simply type wget followed by the download link. Example: **wget** [**https://youtu.be/DXh2\_CTJW9w**](https://youtu.be/DXh2_CTJW9w)

## 24.uname command

The **uname** command, short for Unix Name, will print detailed information about your Linux system like the machine name, operating system, kernel, and so on.

## 25.[top command](https://www.hostinger.in/tutorials/vps/how-to-manage-processes-in-linux-using-command-line)

As a terminal equivalent to Task Manager in Windows, the **top** command will display a list of running processes and how much CPU each process uses. It’s very useful to monitor system resource usage, especially knowing which process needs to be terminated because it consumes too many resources.

## 26.history command

When you’ve been using Linux for a certain period of time, you’ll quickly notice that you can run hundreds of commands every day. As such, running **history** command is particularly useful if you want to review the commands you’ve entered before.

## 27.man command

Confused about the function of certain Linux commands? Don’t worry, you can easily learn how to use them right from Linux’s shell by using the man command. For instance, entering **man tail** will show the manual instruction of the tail command.

## 28.echo command

This command is used to move some data into a file. For example, if you want to add the text, “Hello, my name is John” into a file called name.txt, you would type

**echo Hello, my name is John >> name.txt**

## 29.hostname command

If you want to know the name of your host/network simply type hostname. Adding a - i to the end will display the IP address of your network. Example: **hostname -i**

## 30.useradd, userdel command

Since Linux is a multi-user system, this means more than one person can interact with the same system at the same time. useradd is used to create a new user, while passwd is adding a password to that user’s account. To add a new person named John type, **useradd John** and then to add his password type, passwd 123456789.

To remove a user is very similar to adding a new user. To delete the users account type, userdel UserName

## Bonus Tips and Tricks

Use the clear command to clean out the terminal if it is getting cluttered with too many past commands.

Try the TAB button to autofill what you are typing. For example, if you need to type Documents, begin to type a command (let’s go with cd Docu, then hit the TAB key) and the terminal will fill in the rest, showing you cd Documents.

Ctrl+C and Ctrl+Z are used to stop any command that is currently working. Ctrl+C will stop and terminate the command, while Ctrl+Z will simply pause the command.

If you accidental freeze your terminal by using Ctrl+S, simply undo this with the unfreeze Ctrl+Q.

Ctrl+A moves you to the beginning of the line while Ctrl+E moves you to the end. You can run multiple commands in one single command by using the “;” to separate them. For example Command1; Command2; Command3. Or use && if you only want the next command to run when the first one is successful

**03.Shell Script to print student name, student number, student class.**

echo enter the stno

read stno

echo enter the stname

read stname

echo enter the stcls

read stcls

echo $stno "-" $stname "-" $stcls

**OUTPUT:**

rashmitha088@SYSL30:~$ vi studentdetails.sh

rashmitha088@SYSL30:~$ bash studentdetails.sh

enter the stno

88

enter the stname

rashmitha

enter the stcls

bscs

88 - rashmitha– bscs