Lecture 1

Link : https://youtu.be/Juo_0lpBMPY

About the lecture :

This Linux tutorial will also give you an introduction to the basic Linux commands so that you can start using the Linux CLI. Do watch the video till the very end to see all the demonstrations. Below are the topics covered in this tutorial:

1) What happens when you turn on a computer?
2) Kernels, Bootloaders, and Linux History
3) Why go for Linux?
4) Various distributions of Linux
5) Basic Linux commands: ls, cd, pwd, clear commands
6) Working with files & directories: cat, vi, gedit, mkdir, rmdir, rm commands

What happens when you start your computer?
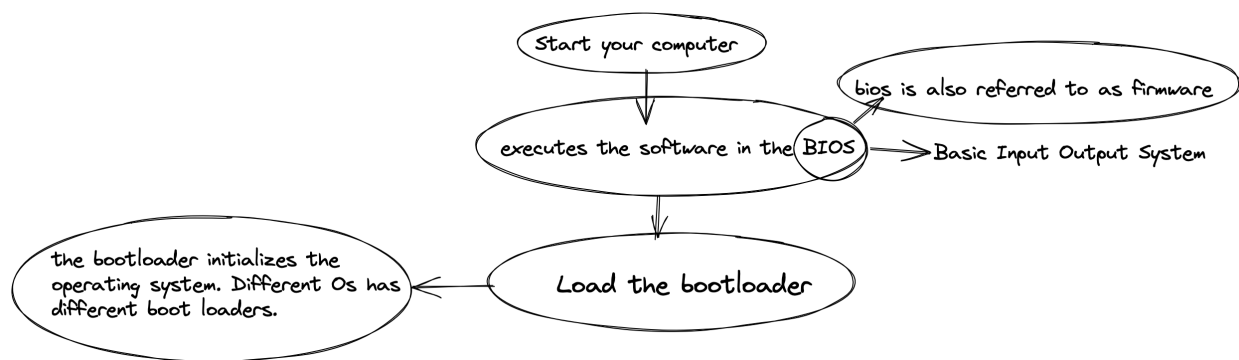
What is an Operating System?

What is an Operating System

Operating System is just a simple software that is responsible for managing and operating a computing device

eg: Laptop, Atm machines, mobile, etc

Things a Software needs to be an Operating System

1) Kernel

2) File System

3) Some sort of User Interface
eg : CLI(Command Line Interface) or GUI(Graphical User Interface)

4) It should be able to take your instructions and it should be able to manipulate the data based on commands.

What is a Kernel?

▼ A kernel is basically the core of any Operating System and it generally has complete control over everything in the system. It can control the memory, cpu time of any software that is there. It basically facilitates the interaction between a software and the hardware.

What is a file system?

▼ File System is basically a method or data structure that the Operating System Uses to store and retrieve data in the memory.

Notes:

▼ MacOs and Linux both are UNIX based Operating Systems.

▼ Windows on the other hand has some components which are UNIX based but majorly it has been derived from other Operating Systems.

Why use Linux?

▼ Linux is Open Source. Developers love open source.

▼ It supports almost all programming languages. It also has vast variety of applications that are developer friendly.

▼ It has Terminal which has better support and superior to Command Line.

▼ It opens scope for Bash Scripting.

▼ It gives you the ability to SSH into any server.

History of Linux :

Wikipedia : https://en.wikipedia.org/wiki/History_of_Linux
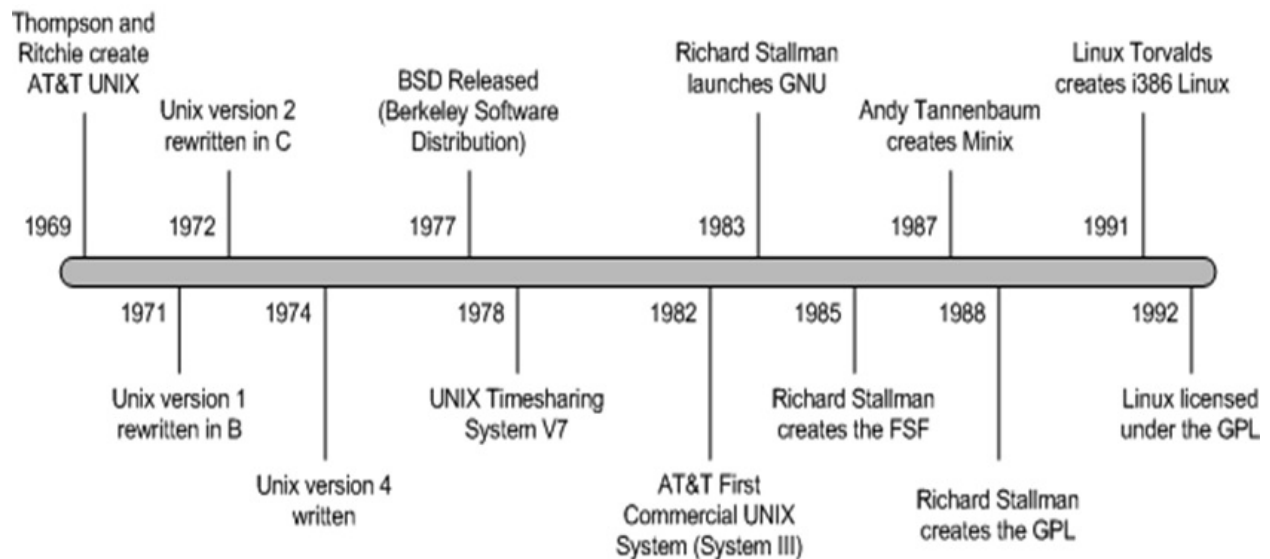
Important Point : Good developers focus more on keyboard than mouse.

What is a terminal?

▼ It is basically a text input and output environment. It is also referred to as console. Its job is to just launch the shell that is there.

What is a shell?

▼ A shell is basically a command line interpreter.

Why it's an interpreter?

▼ Because it executes each command line by line.

Most of the Shell use Bash Programming Language.

Writing Hello World in Bash Programming Language
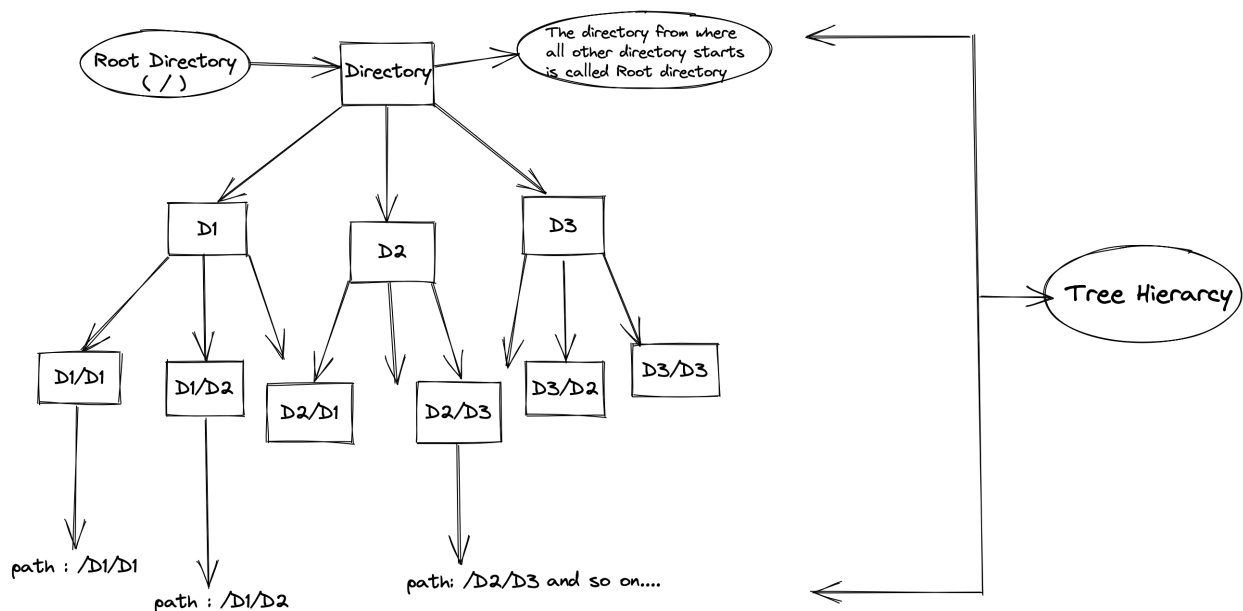
```
> echo Hello, World!
Hello, World!

~
```

echo command basically prints everything after that.

Hello World is the argument here(what do we want to echo).

Few Commands :

▼ pwd(Print Working Directory or present working directory) : It simply tells you where you are in the computer

Tree Hierarchy for directories/folders:

▼ ls(list items) : with the help of this command you can see the folders that are inside the current directory

▼ cd (change directory) : with the help of this command we can change directories

▼ cd - (previous directory) : With the help of this command we can go back to previous directory.

▼ ls -a : With the help of this we can see all the hidden files in the directory.

. : current directory

.. : Parent directory

~ : home directory

— :Previous directory

.

Note : Whatever starts with a . in linux is a hidden by default. We can't access it with the gui.

▼ ls -l : This command gives a description of all the folders that can be access with gui too.

▼ ls -la : This command gives a long description of all the hidden directories which cannot be accessed using gui.

▼ clear : If you feel your terminal is populated you can use this command and everything will get erased out.

▼ #command : Since bash is a programming language we can add comments to it using hash.

▼ mkdir(make directory) : This command helps us to create a new directory

▼ touch : It simply creates a file.

Relative way to create a folder from another directory.

```
❯ touch Documents/bashlanguage/a.txt
❯ ls Documents/bashlanguage
a.txt     bash.txt


 ~
 0.0 kB↓                                    0.0 kB↑    3.6 GB
```

▼ ls / : to check whats in the root directory.

Note (Important vim commands):

▼ vi "file-name" : with the help of this we can enter vim and then write texts into a file.

▼ To save a file in Vim / vi, **press Esc key, type :w and hit Enter key.**

▼ save a file and quit vim / Vi by pressing Esc key, type :x and hit Enter key.

▼ cat (file-name) : Short form of concatenate. It just displays the files contents and it can also combine multiple files content and show the output to us in the console.

▼ history : Gives you the history of all the commands you have executed in the terminal since the beginning of the computer.

▼ Up arrow key : With this key we can see the previous command we executed.

▼ control c : Adds a new line and comes handy and exits from anything in the terminal.

▼ control r(Reverse Search) : With this command we can search the command which we entered previously.

▼ cp(copy file) : This helps to copy file and past it in another directory.

```
> cp class\ HelloWorld.java Documents
> cd Documents
> ls
Apna College          IntelliJ              bashlanguage             project
GitHub                Sublime files         class HelloWorld.java projects
HTML                  Udemy                 linux_master_class

~/Documents
```

▼ mv(move command) : With the help of this command we can move a file from one directory to another.

```
> mv class\ HelloWorld.java folder
> cd folder
> ls
class HelloWorld.java hello.java              input.java              mini.class
> cd ..
> pwd
/Users/soumyadeepdutta
> ls
Applications          Library                Public
Desktop               Movies                 folder
Documents             Music                  java_error_in_idea_25
Downloads             Pictures               powerlevel10k

~
  0.0 kB↓                            0.0 kB↑   ⊞ 3.6 GB                      ⊟ 5%
```

▼ cd / : to go to root directory.

▼ mkdir -p : We can make directory inside directory using this command.

The `-p` flag basically creates a directory and inside the directory points the sub directory.

```
> mkdir -p folder/hanga2
> ls folder
class HelloWorld.java hanga2                  input.java              output.java
hanga                 hello.java              mini.class
```

▼ rm dir : This command is used to remove a directory which has no content inside.

▼ rm -r : This command is used to delete a directory which has content inside.

The `-r` means recursion and it removes all the files inside the directory recursively and then removes the directory.

▼ rm -rv : The command is used to delete a directory which has content inside but it then displays the content which it deleted.

The `v` means verbally and tells you what it deleted.

▼ find : with the help of this command we can find any file .

```
apoorvgoyal@apoorvtwts:~$ mkdir -p bootcamp/testing/testing2
apoorvgoyal@apoorvtwts:~$ touch bootcamp/testing/testing2/a.txt
apoorvgoyal@apoorvtwts:~$ find ~ -name a.txt
/home/apoorvgoyal/bootcamp/testing/testing2/a.txt
apoorvgoyal@apoorvtwts:~$ find ~ -name a.*
/home/apoorvgoyal/bootcamp/testing/testing2/a.txt
apoorvgoyal@apoorvtwts:~$ find ~ -name *.txt
```

▼ man : with the help of this command we can find the guide of a particular command.

▼ whatis : this command tells you about a particular command. Its like asking the computer what is ls , etc.

▼