

Master Executive di II Livello
**BIG DATA ANALYSIS AND
BUSINESS INTELLIGENCE**

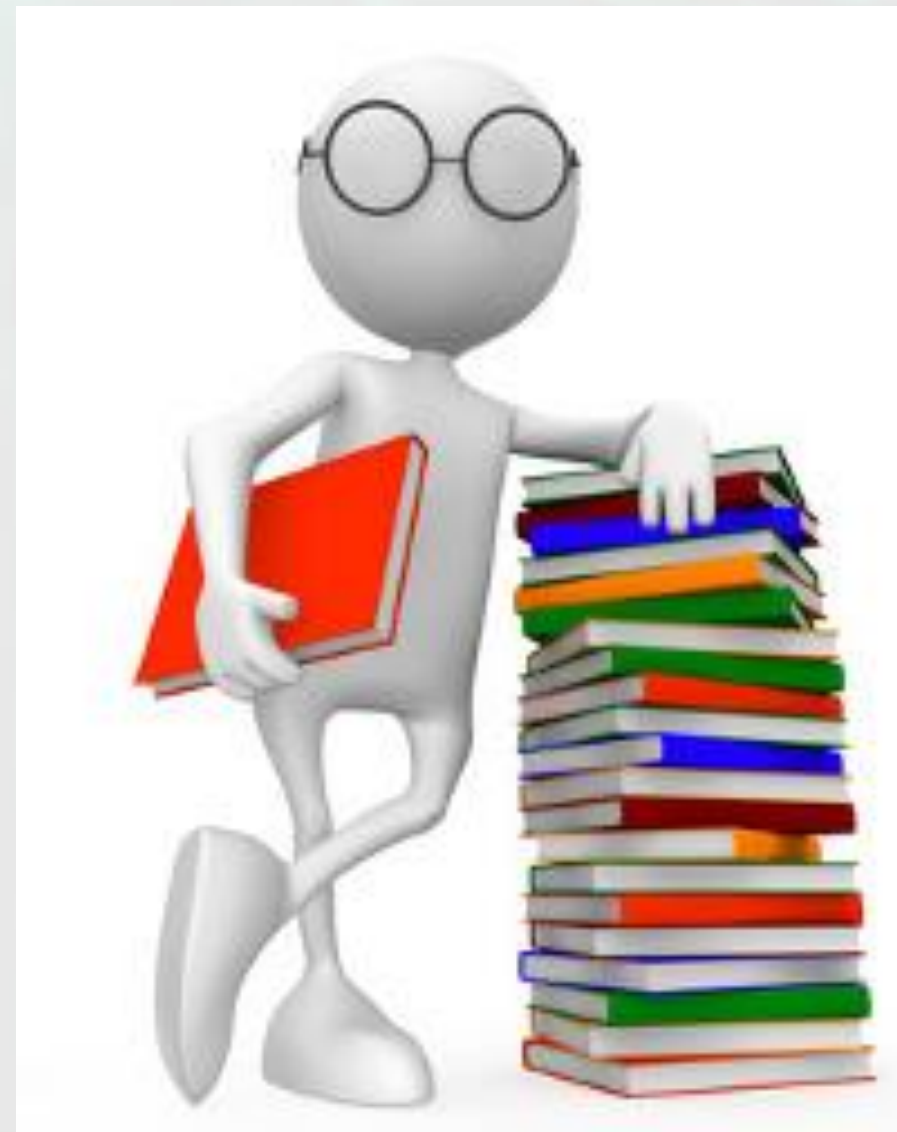
Vamsi Krishna Varma Gunturi
Data science intern at ISTAT
vamsivarmagunturi@gmail.com

Introduction to Linux

fondazione

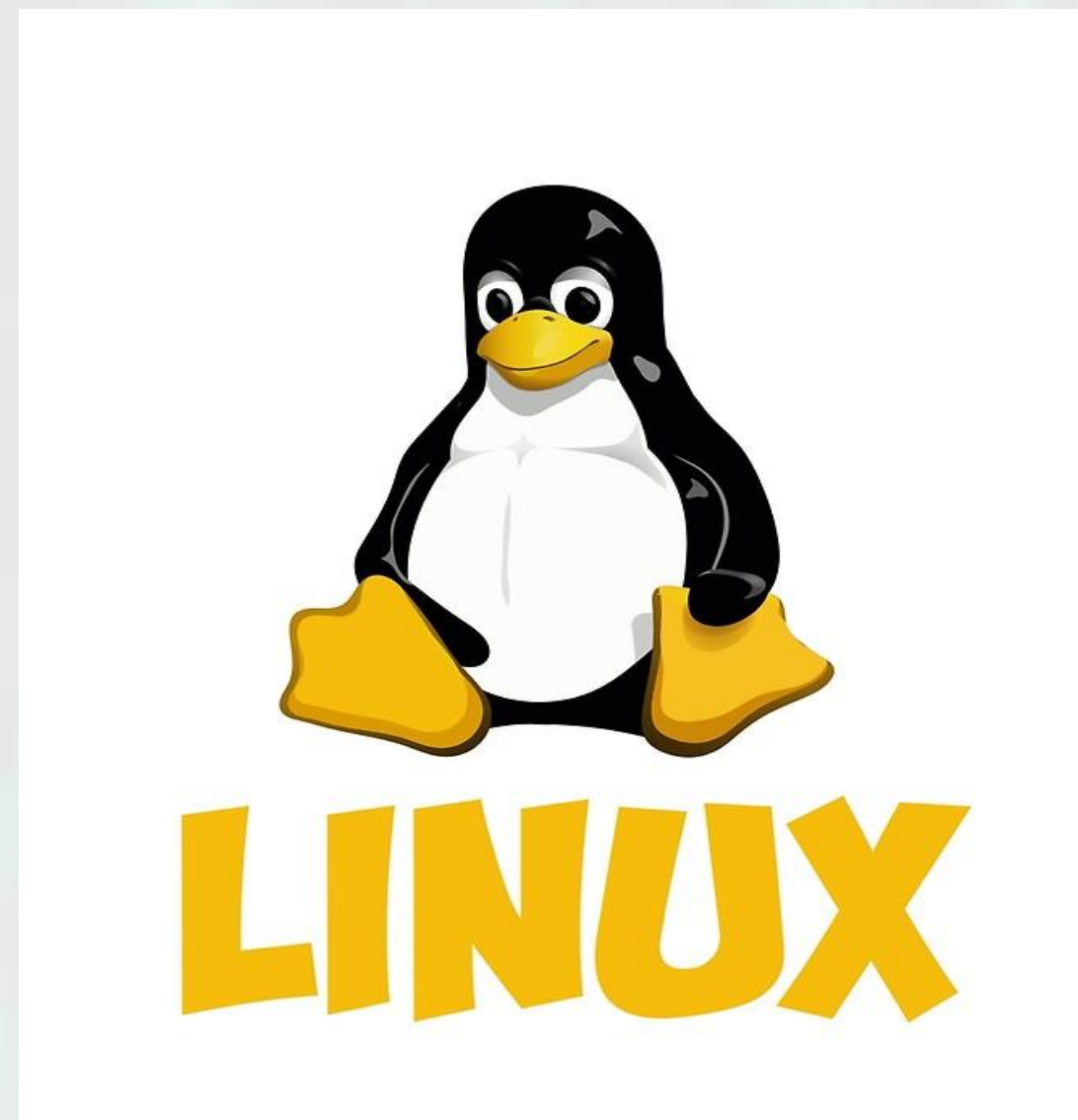
INOIT
TORVERGATA

Topics



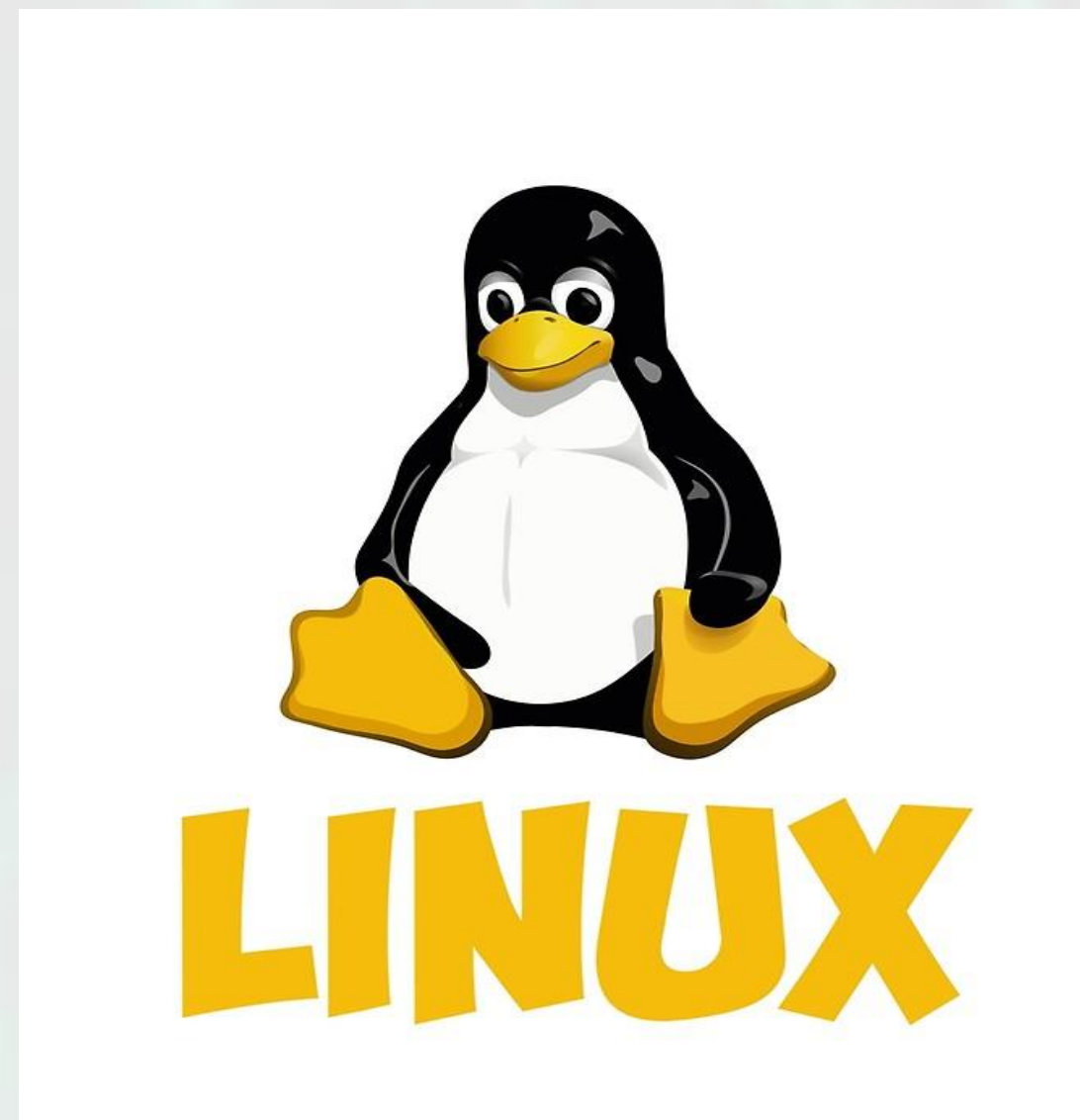
- Introduction to Unix
- History of Unix
- What is Linux ?
- Why Linux ?
- Linux distributions
- Linux OS structure
- Linux file system
- Linux directories
- Users, Groups and permissions
- Some important commands in Linux
- vi editor
- Environment variables

Introduction to Unix



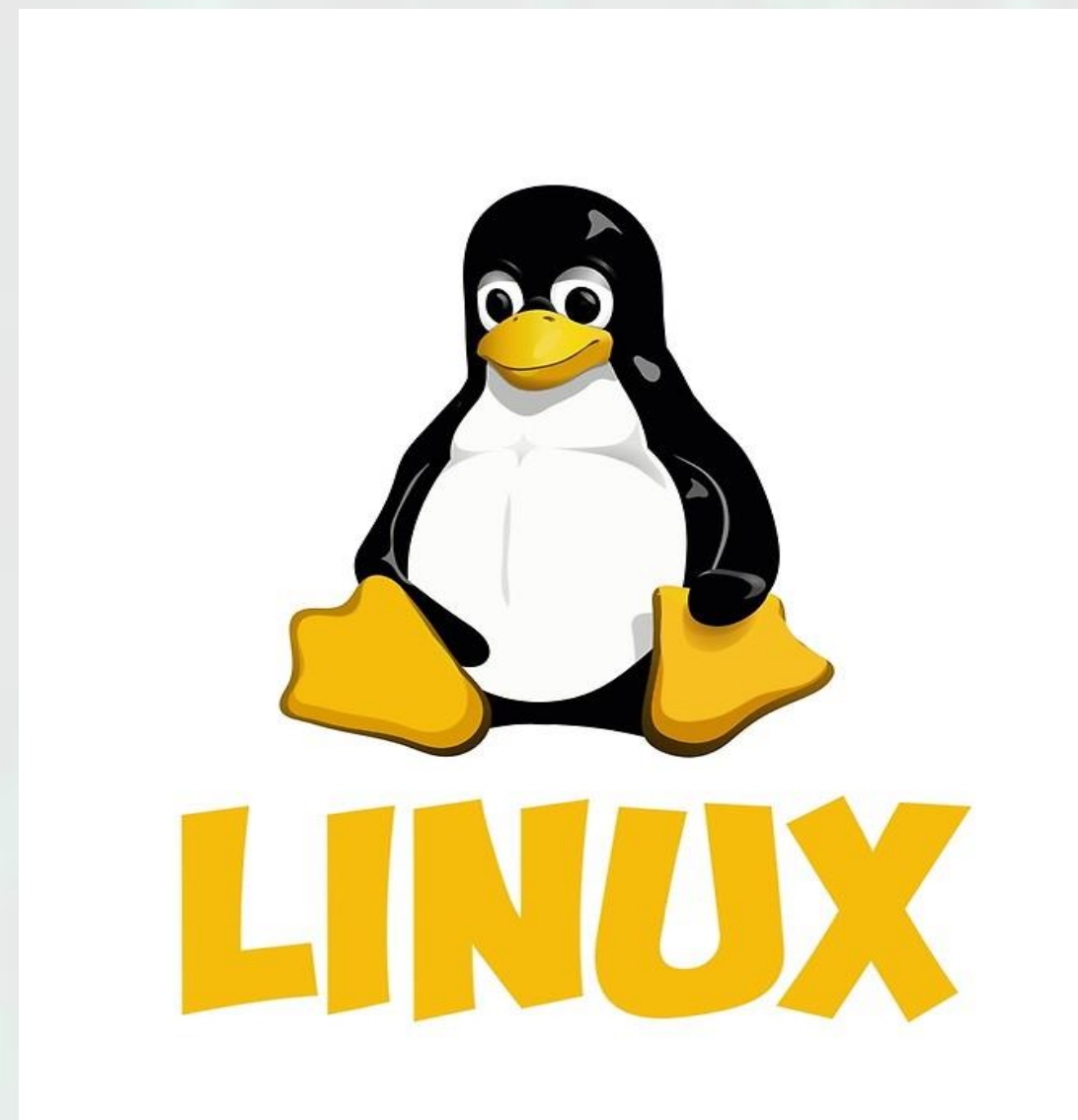
- An operating system
- Developed at AT&T Bell Labs in the 1960's
- Command Line Interpreter
- GUIs (Window systems) are now available.
- Unix is a multi-user, multi-tasking operating system. You can have many users logged into a system simultaneously, each running many programs.
- It's the kernel's job to keep each process and user separate and to regulate access to system hardware , including CPU, memory, disk and other I/O devices.

History of Unix



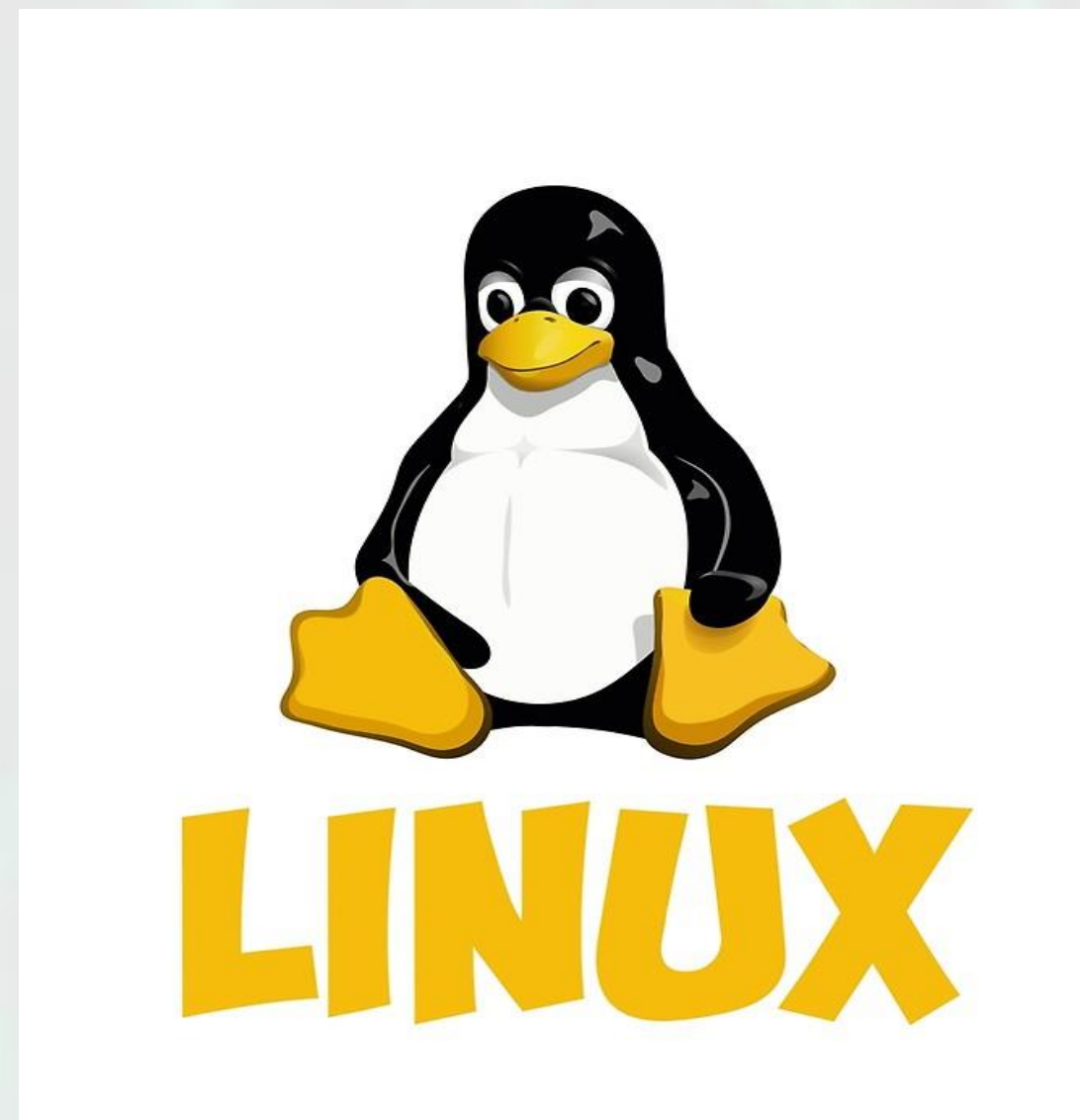
- First Version was created in Bell Labs in 1969.
- 00:00:00 Hours, Jan 1, 1970 is time zero for UNIX. It is also called as epoch.
- 1973 Unix is re-written mostly in C, a new language developed by Dennis Ritchie. Being written in this high-level language greatly decreased the effort needed to port it to new machines.
- 1991 Linux was originated.

What is Linux ?



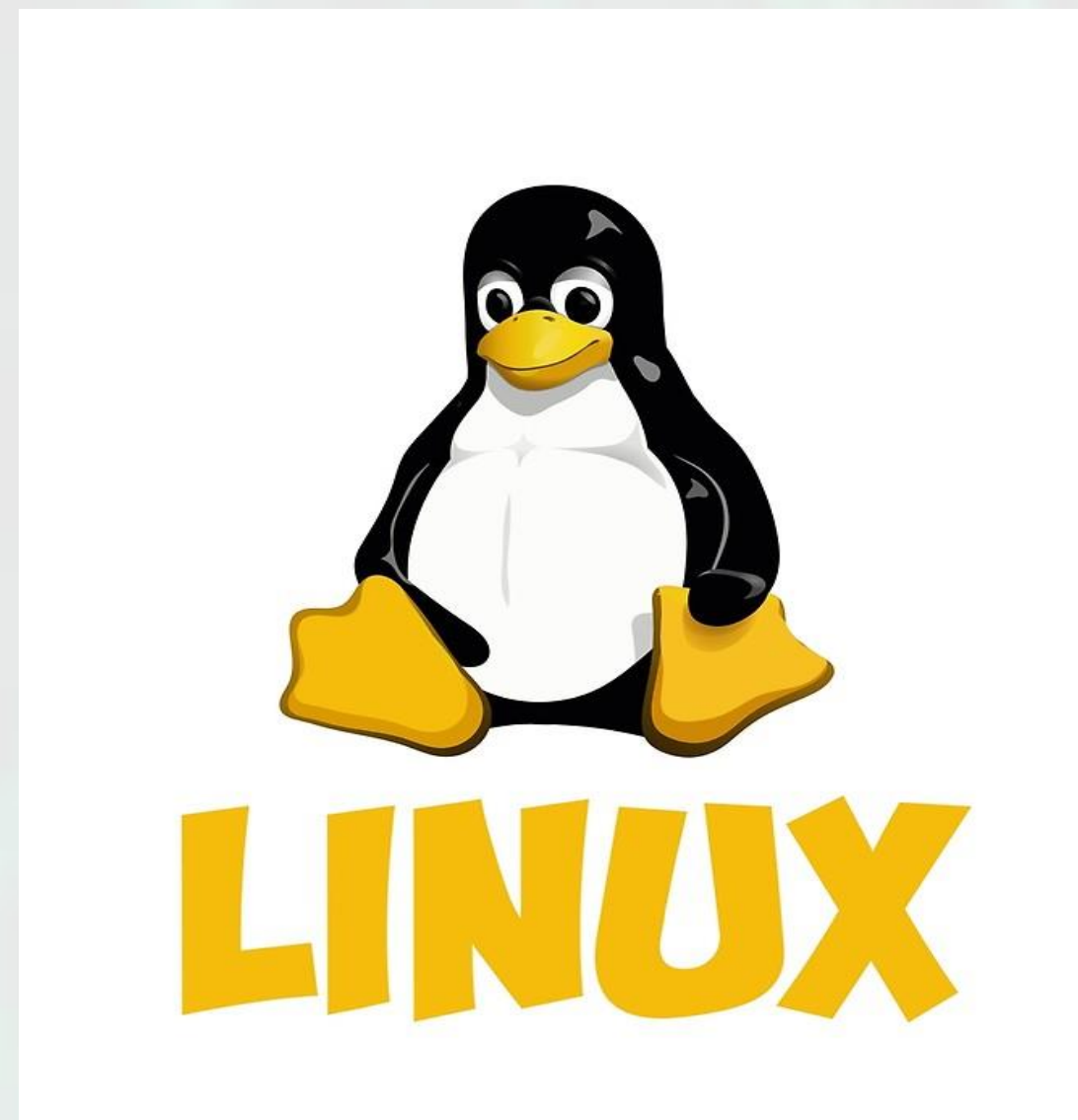
- Linux is a free Unix-type operating system originally created by Linus Torvalds with the assistance of developers around the world. Now it runs on a variety of platforms including PowerPC, Macintosh, Amiga, DEC Alpha, Sun Sparc, ARM, and many others.
- It originated in 1991 as a personal project of Linus Torvalds, a Finnish graduate student.
- Developed under the GNU General Public License , the source code for Linux is freely available to everyone.
- Now it has evolved into a very functional, powerful and usable clone of Unix which has at least 10 million users worldwide.

Why Linux ?



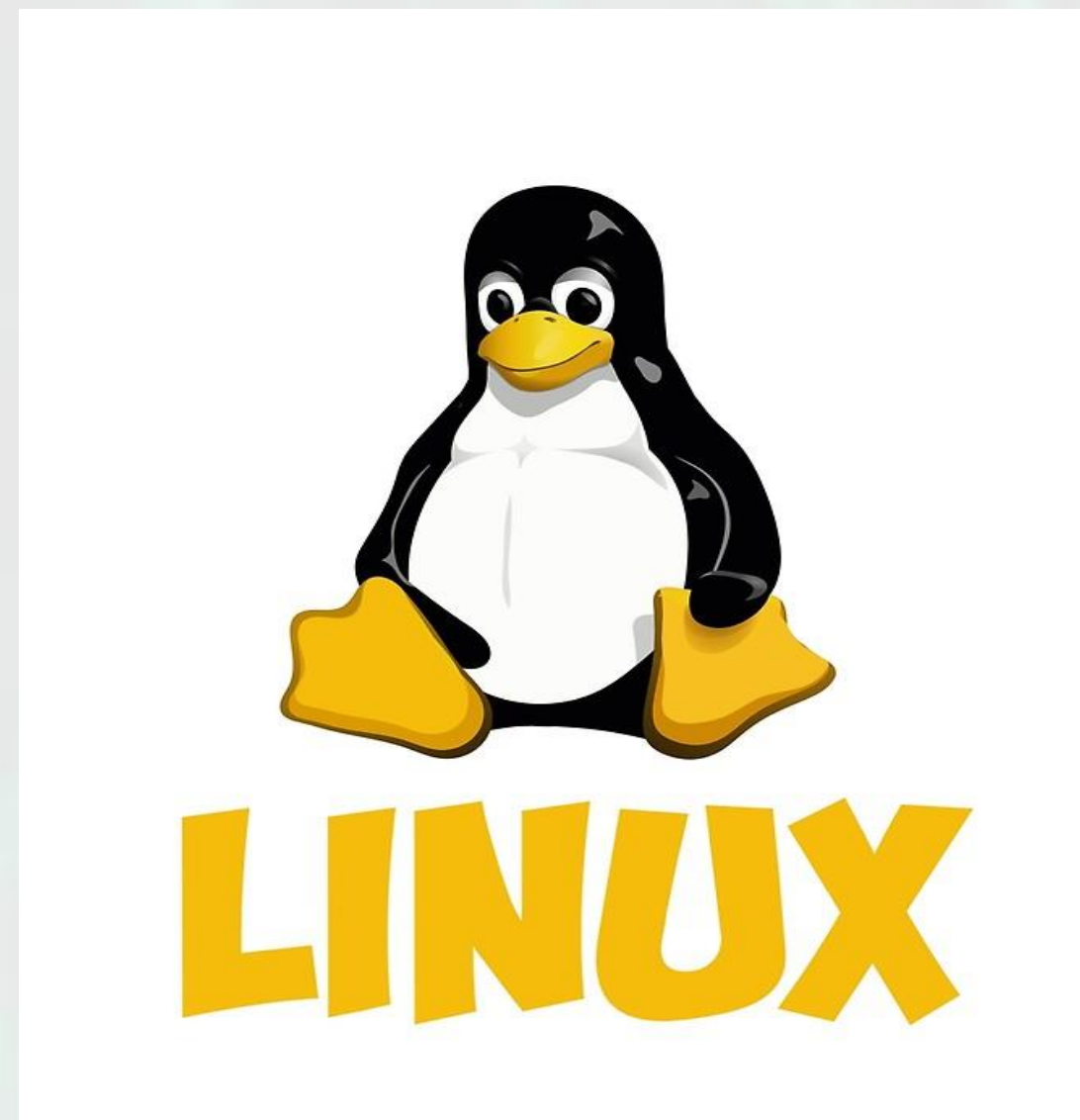
- A Linux Distribution has thousands of dollars worth of software for no cost.
- Linux is a complete operating system:
 - Stable - the crash of an application is much less likely to bring down the OS under Linux.
 - Reliable - Linux servers are often up for hundreds of days compared with the regular reboots required with a Windows system.
 - Extremely powerful
- Linux provides a complete development environment.

Why Linux ? (2)



- Excellent networking facilities
- Ideal environment to run servers such as a web server, or an ftp server.
- A wide variety of commercial software is available if not satisfied by the free software
- Easily upgradeable. Supports multiple processors.
- True multi-tasking, multi-user OS.
- An excellent window system called X, the equivalent of Windows but much more flexible.
- Full source code is provided and free.

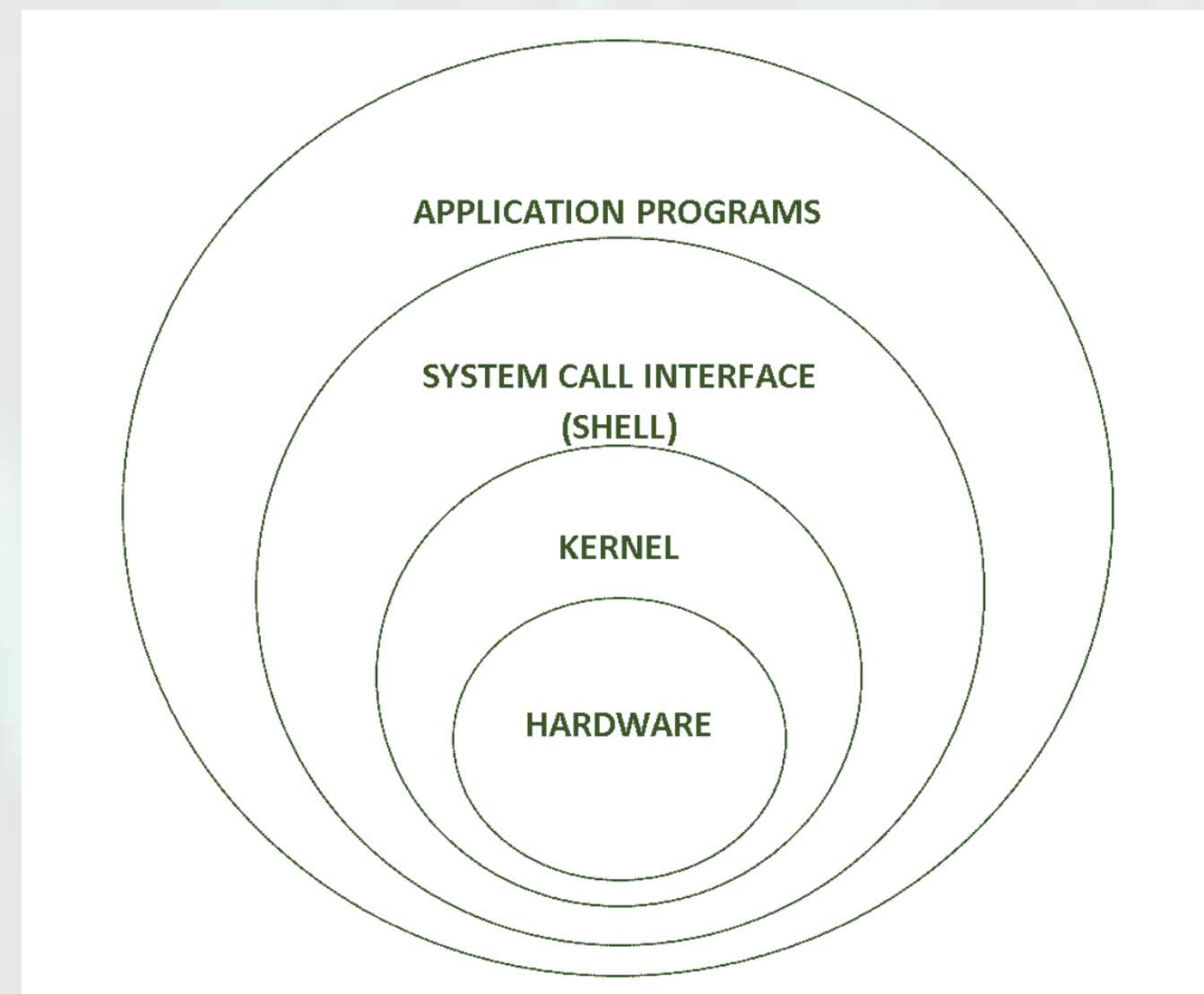
Linux distributions



- Mandrake: <http://www.mandrakesoft.com/>
- RedHat: <http://www.redhat.com/>
- Fedora: <http://fedora.redhat.com/>
- SuSE/Novell: <http://www.suse.com/>
- Debian aka Ubuntu: <http://www.debian.org/>
- CentOS: <http://centos.org/>

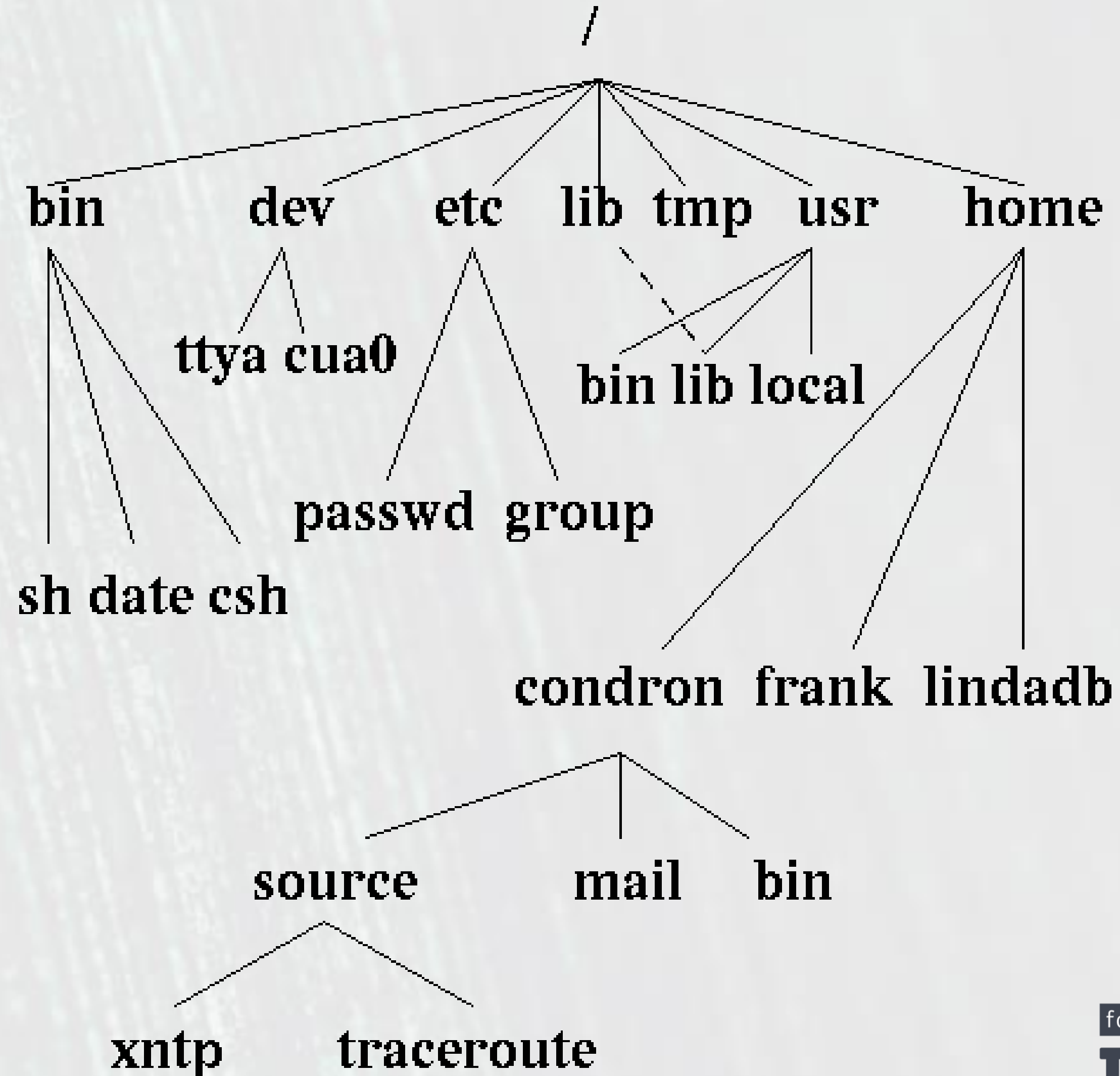
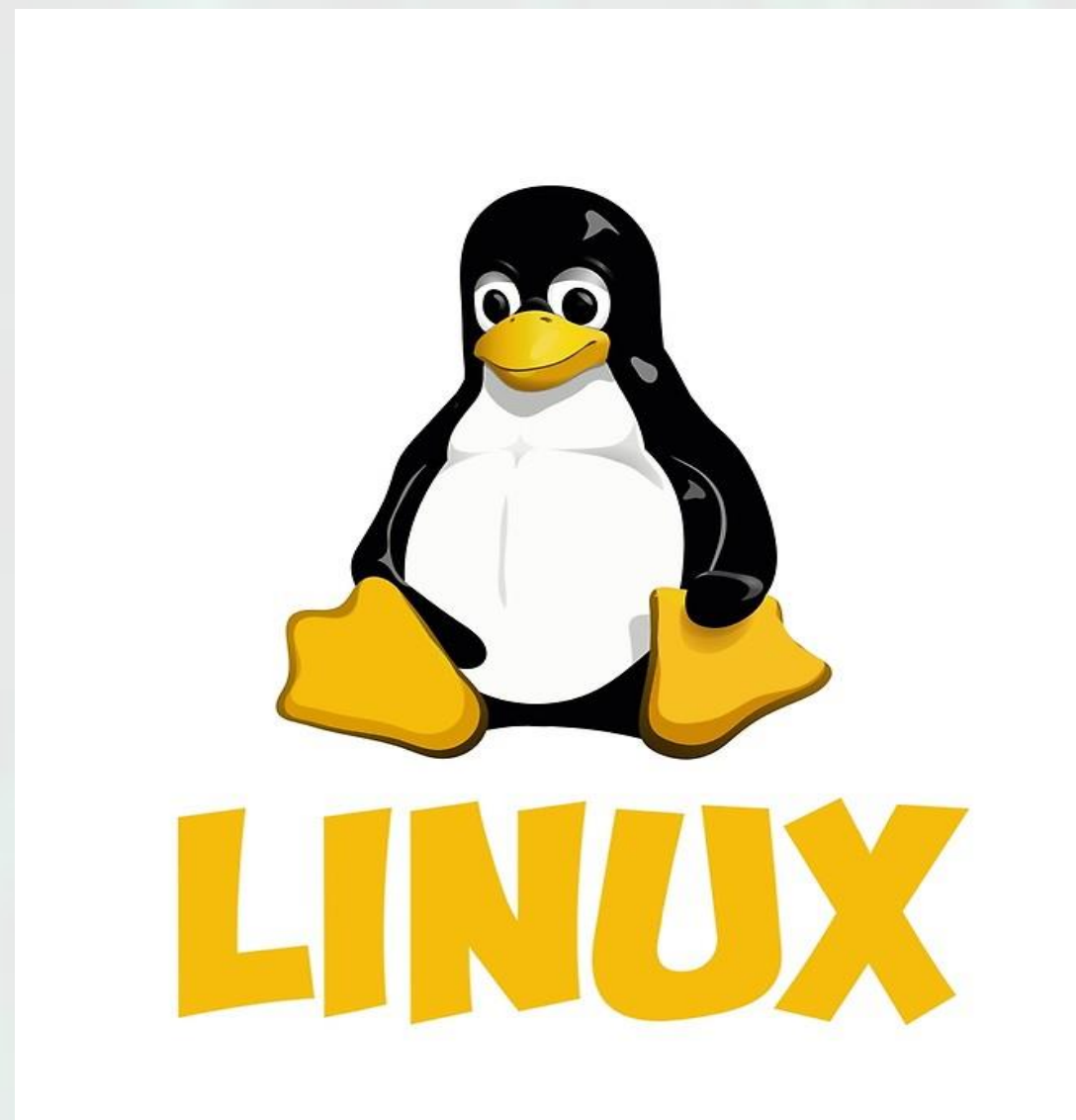
Note: Red Hat Enterprise Linux is a Enterprise targeted Operating System. It based on mature Open Source technology and available at a cost with one year Red Hat Network subscription for upgrade and support contract.

Linux OS structure

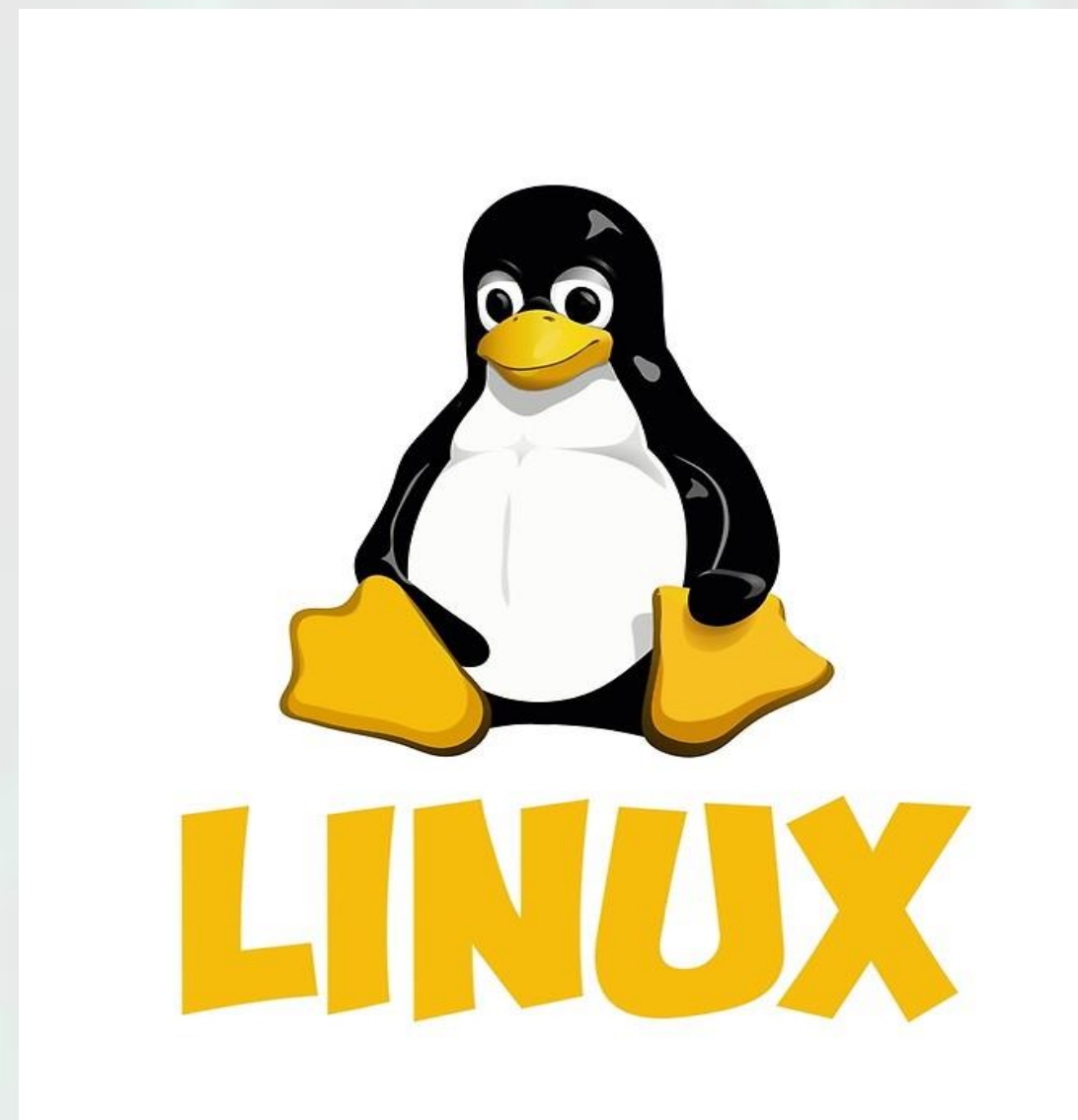


- **Hardware:** all the parts of a computer including clocks, timers, devices, RAM, HDD, SDD etc.,
- **Kernel:** interacts directly with the hardware of the computer. Responsible for Memory management, controlling access to the computer, maintaining the file system, handling interrupts, handling errors, performing input, output services and allocate the resources of the computer among users.
- **Shell:** acts as a mediator between the kernel and the user. It reads the commands and then interpret them and sends a request to execute a program.
- **Application programs:** utility programs which runs on Shell like Your web browser, media player, text editor etc.,

Linux file system

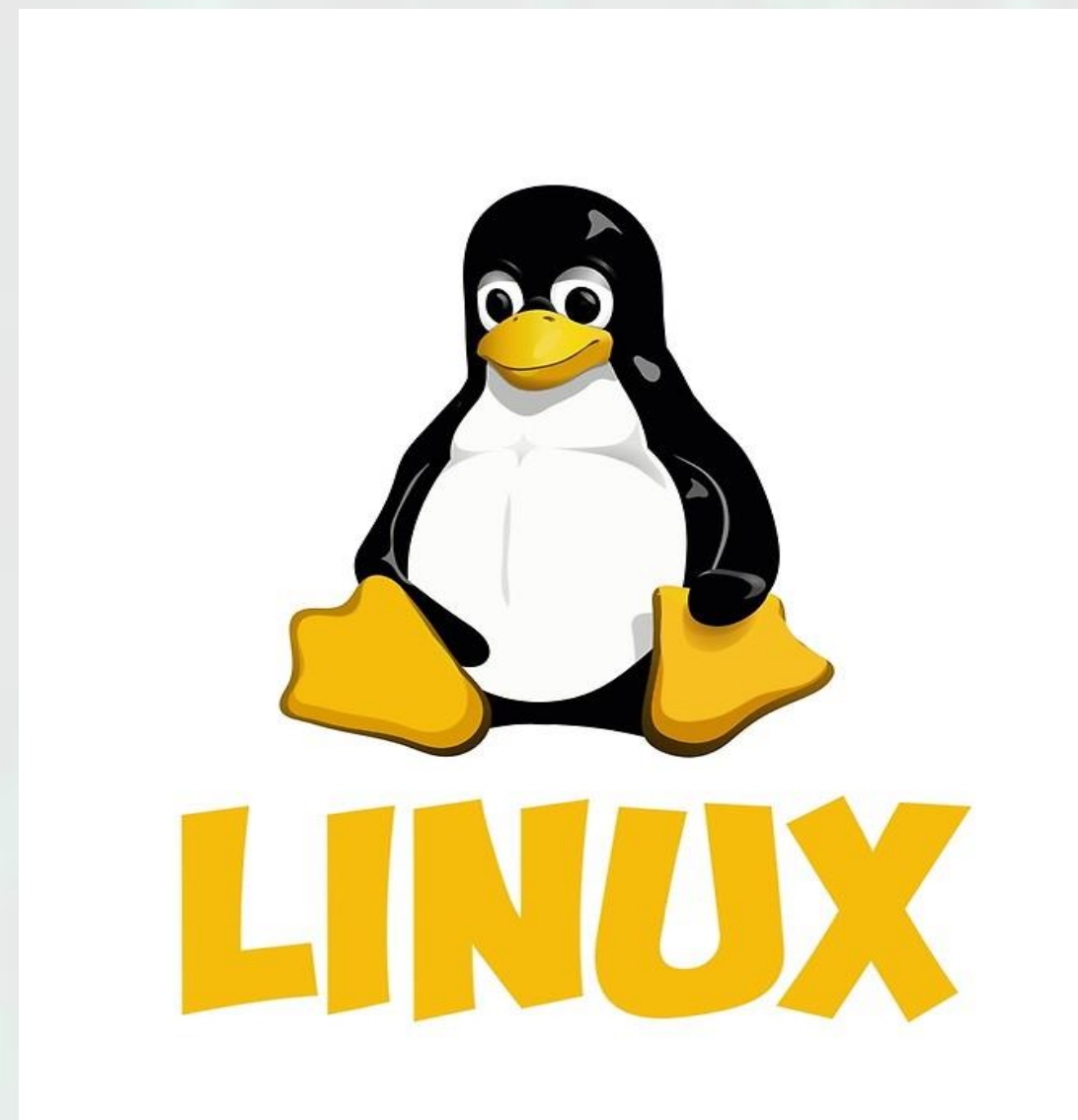


Linux file system



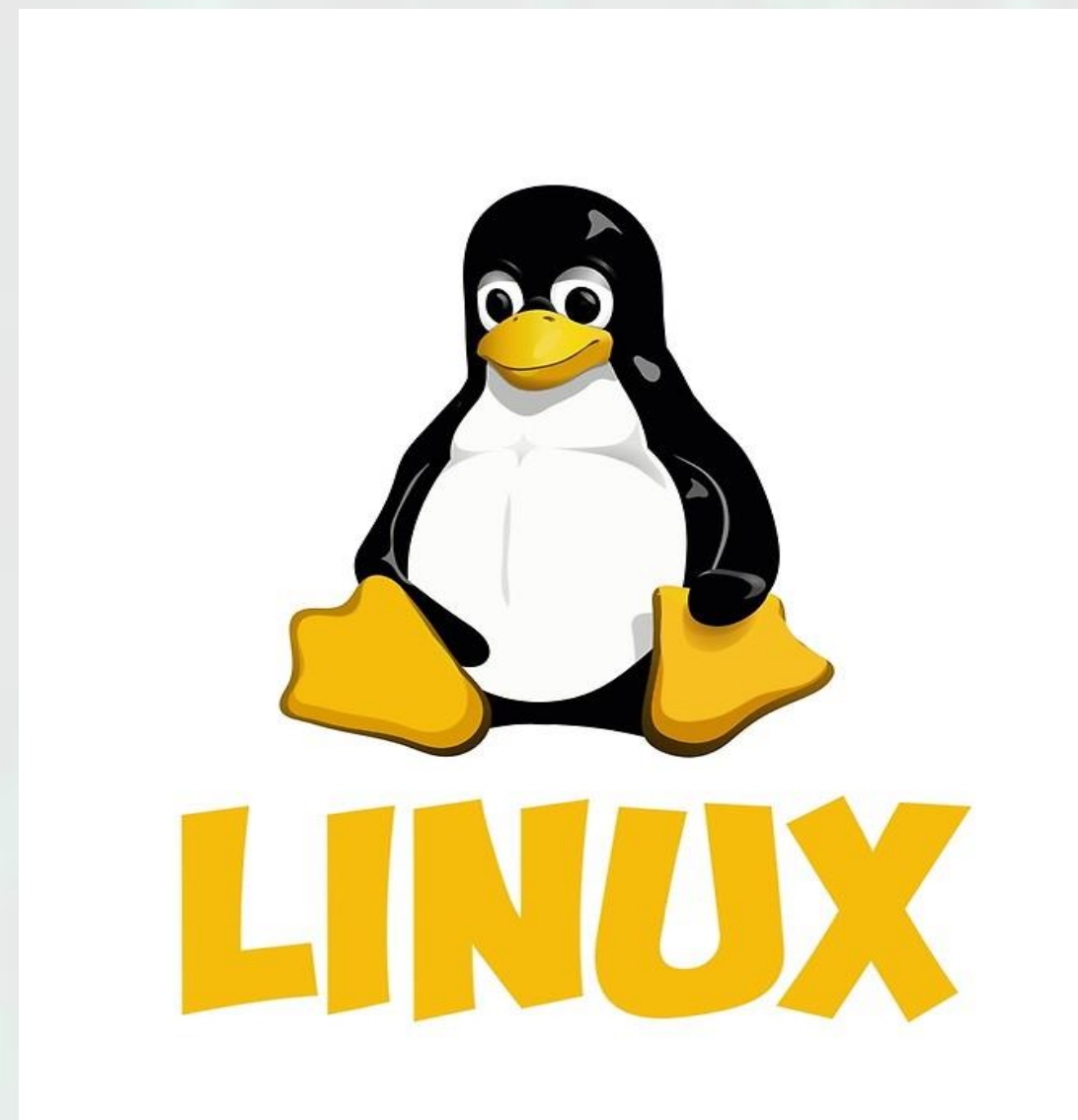
- The Unix file system looks like an inverted tree structure.
- You start with the root directory, denoted by /, at the top and work down through sub-directories underneath it.
- Each node is either a file or a directory of files, where the latter can contain other files and directories.
- You specify a file or directory by its path name, either the full, or absolute, path name or the one relative to a location.
- The full path name starts with the root, /, and follows the branches of the file system, each separated by /, until you reach the desired file,
e.g.: /home/condron/source/xntp

Linux file system(2)



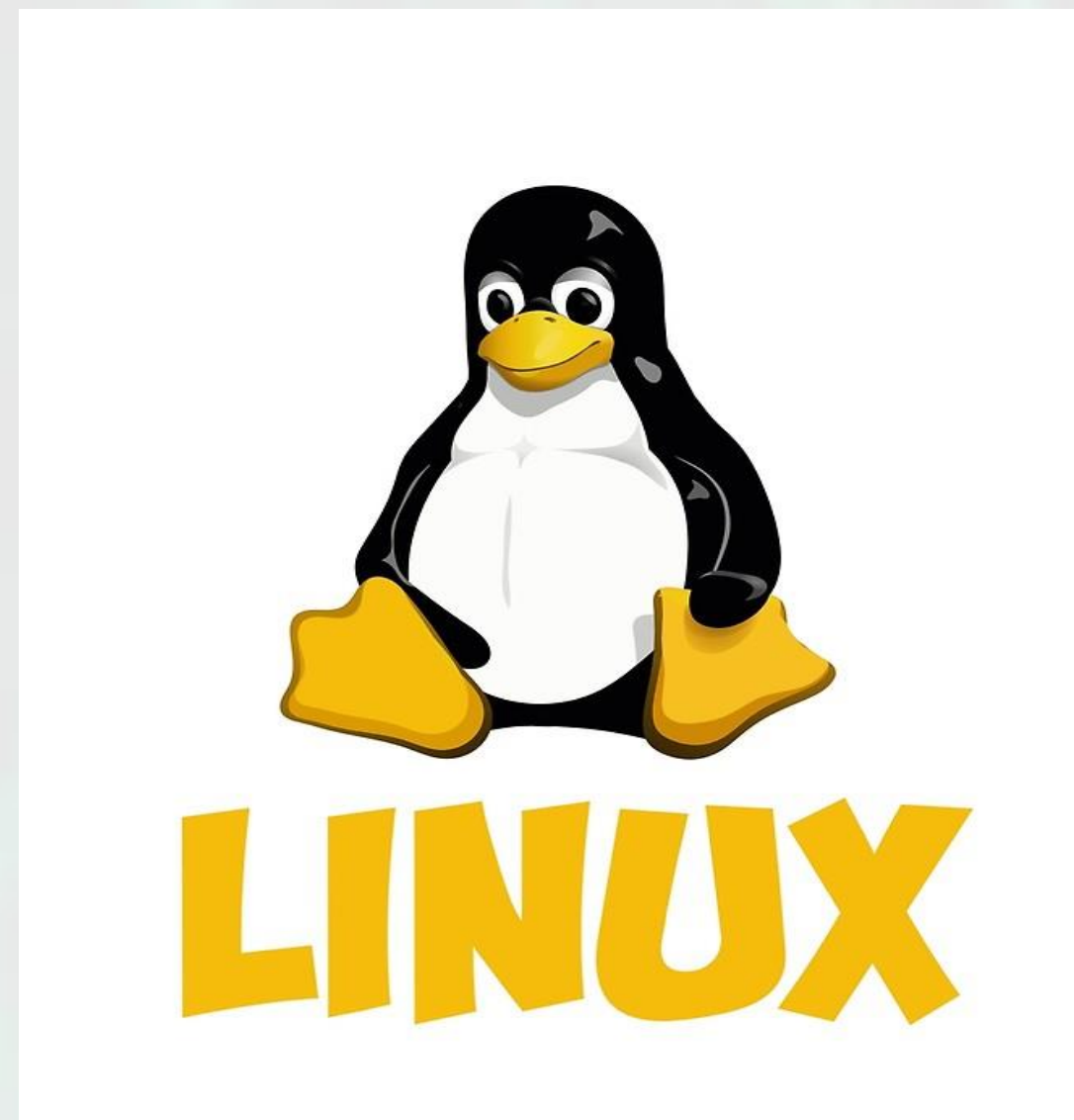
- A relative path name specifies the path relative to another, usually the current working directory that you are at. Two special directories :
 - . the current directory
 - .. the parent of the current directory
- So if I'm at /home/frank and wish to specify the path above in a relative fashion I could use: ../condron/source/xntp
- This indicates that I should first go up one directory level, then come down through the condron directory, followed by the source directory and then to xntp.

Linux directories



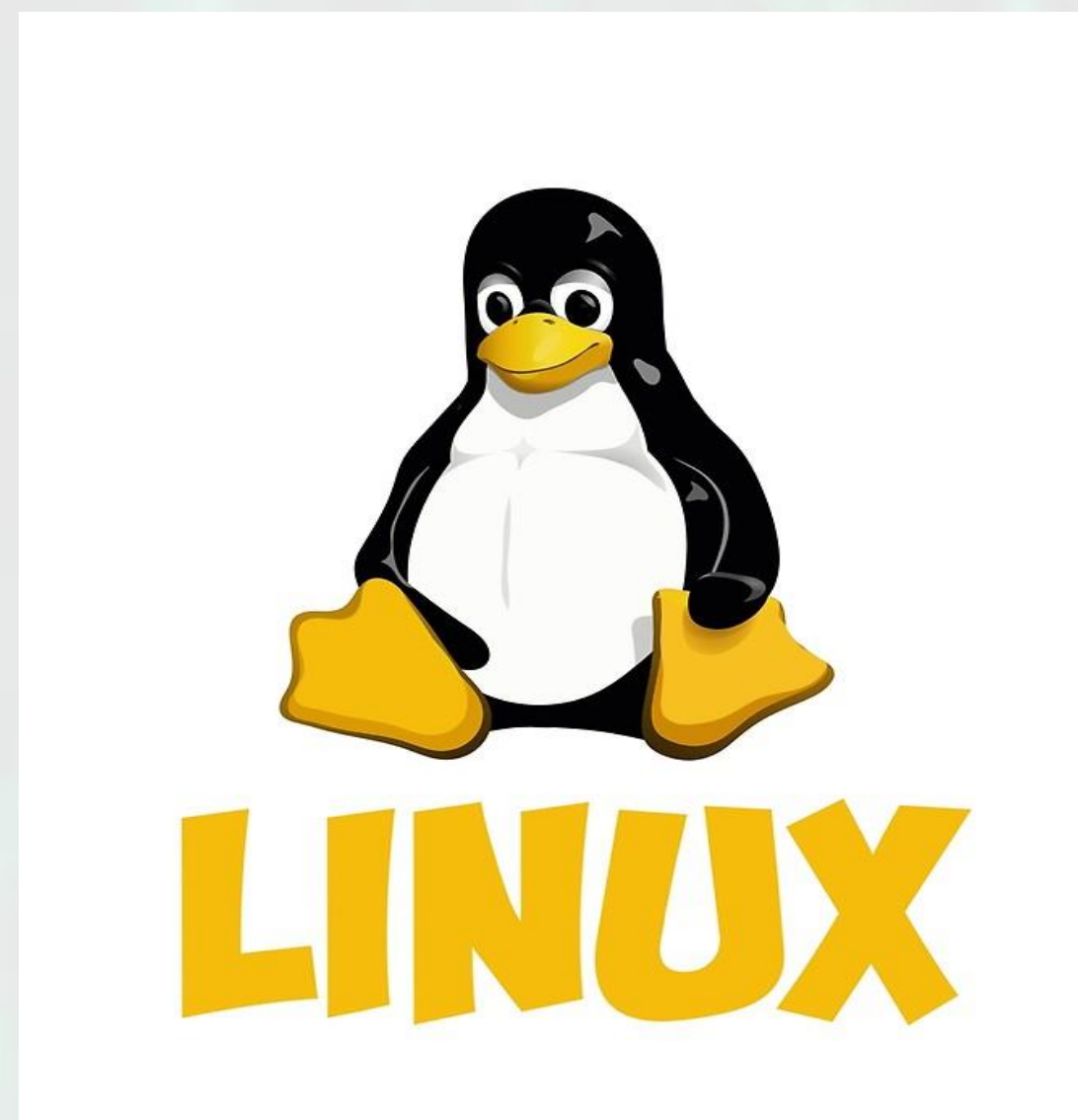
- **/**: The ancestor of all directories on the system; all other directories are subdirectories of this directory, either directly or through other subdirectories.
- **/bin**: Essential tools and other programs (or binaries).
- **/dev**: Files representing the system's various hardware devices. For example, you use the file `/dev/cdrom` to access the CD-ROM drive.
- **/etc**: Miscellaneous system configuration files, start up files, etc.

Linux directories(2)



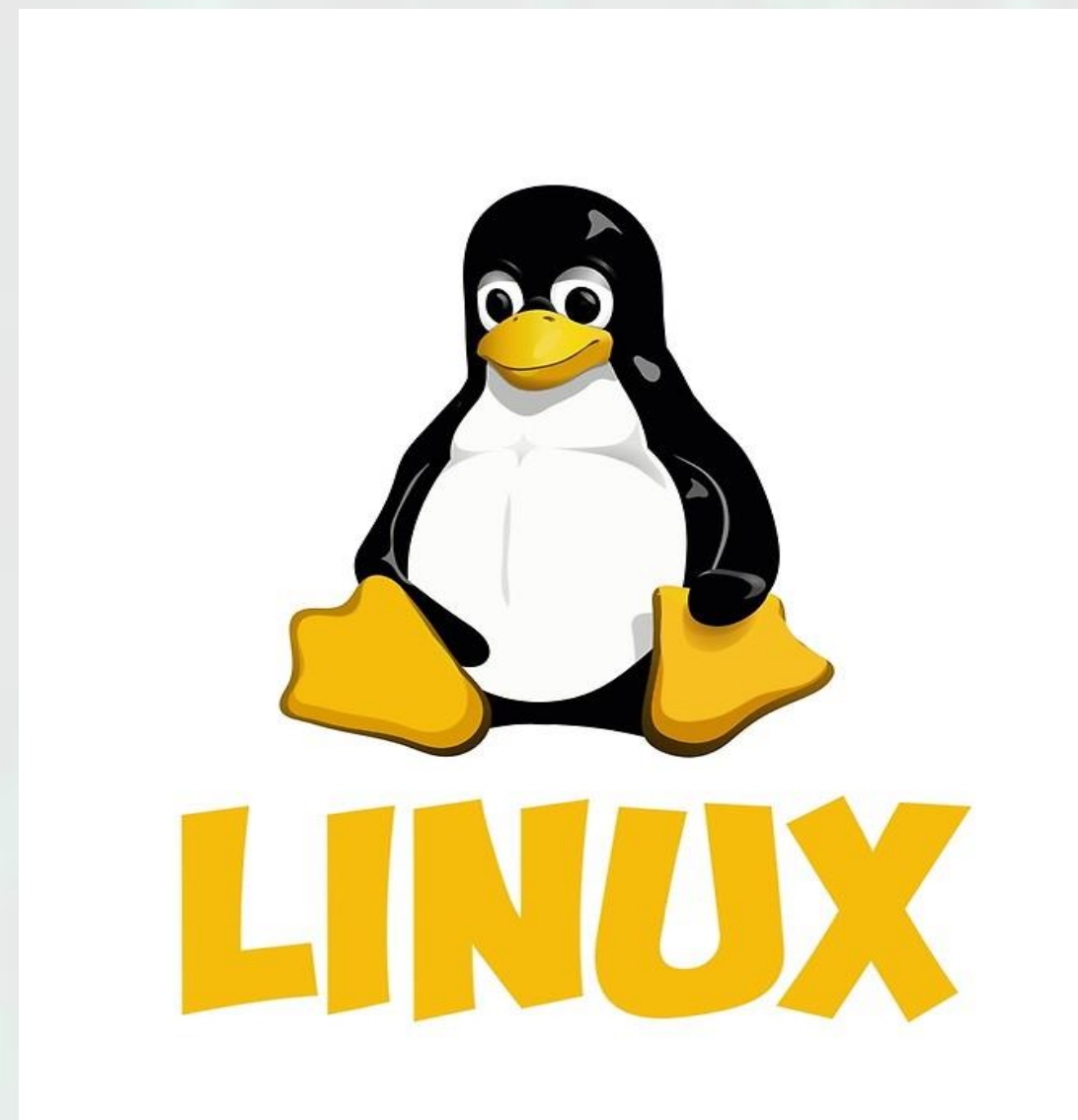
- **/home**: The home directories for all of the system's users.
- **/lib**: Essential system library files used by tools in `/bin`.
- **/proc**: Files that give information about current system processes.
- **/root**: The superuser's home directory, whose username is root. (In the past, the home directory for the superuser was simply `/`; later, `/root` was adopted for this purpose to reduce clutter in `/`.)
- **/sbin**: Essential system administrator tools, or system binaries.
- **/tmp**: Temporary files.
- **/usr**: Subdirectories with files related to user tools and applications.

Users and Groups



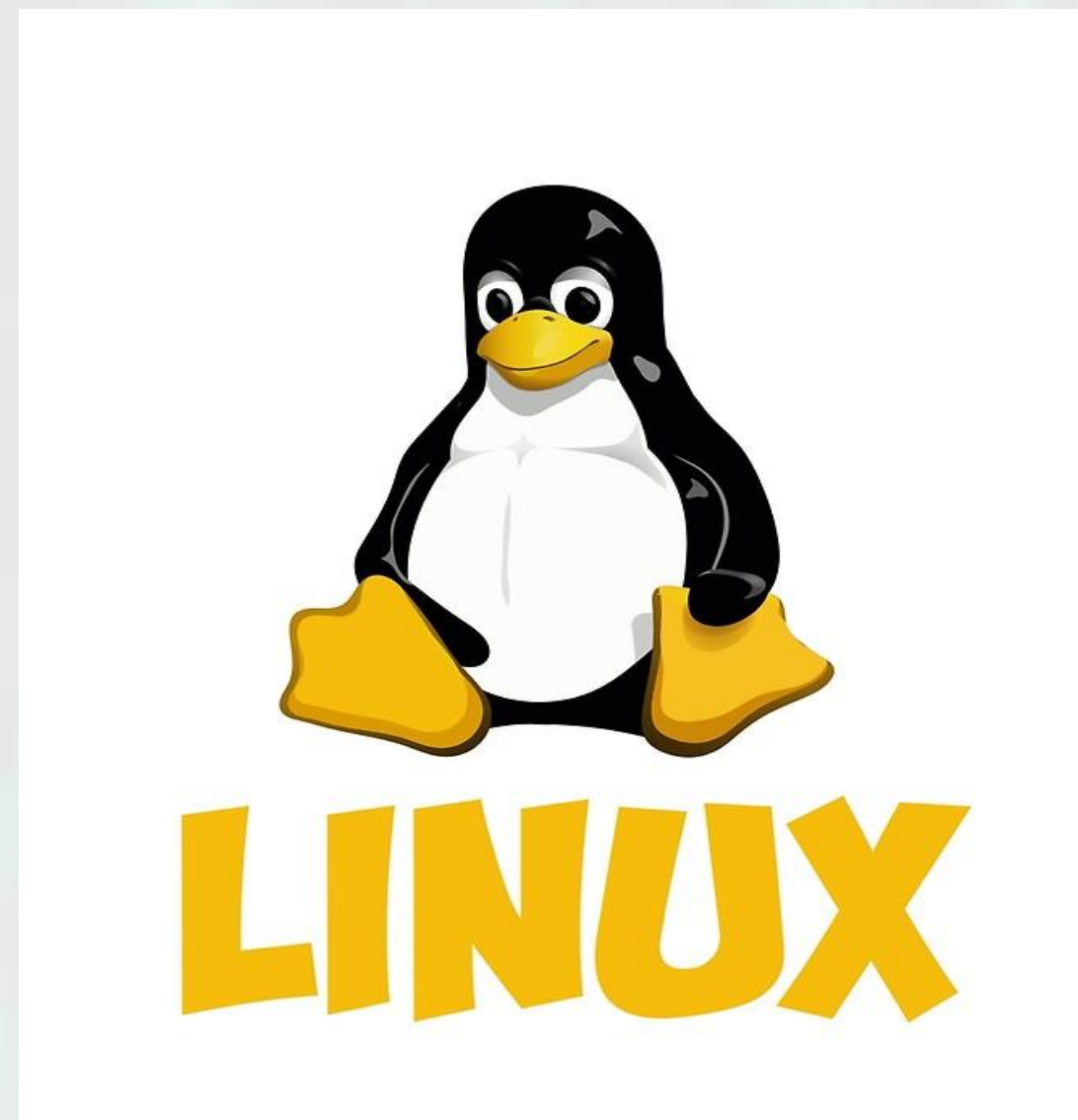
- In UNIX/LINUX, there is a concept of user and an associated group
- The system determines whether or not a user or group can access a file or program based on the permissions assigned to them.
- Apart from all the users, there is a special user called Super User or the root which has permission to access any file and directory

Access Permissions



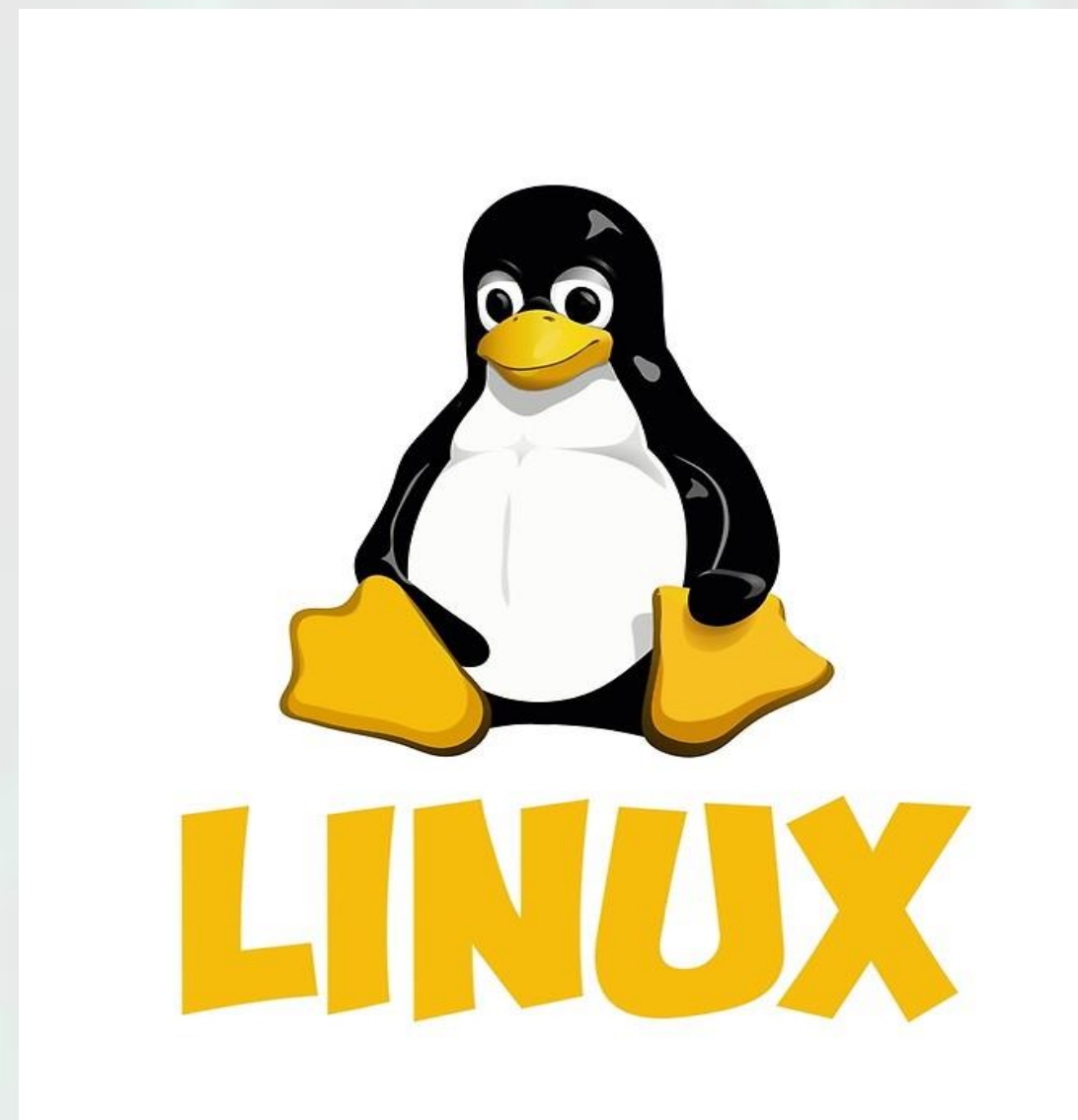
- There are three permissions for any file, directory or application program.
- The following lists the symbols used to denote each, along with a brief description:
 - r — Indicates that a given category of user can read a file.
 - w — Indicates that a given category of user can write to a file.
 - x — Indicates that a given category of user can execute the file.
- Each of the three permissions are assigned to three defined categories of users.
- The categories are:
 - owner — The owner of the file or application.
 - group — The group that owns the file or application.
 - others — All users with access to the system.

Access Permissions(2)



- One can easily view the permissions for a file by invoking a long format listing using the command `ls -l`.
- For instance, if the user emma creates an executable file named test, the output of the command `ls -l test` would look like this: **`rw-rw-r-x 1 emma Oct 11 12:25 test`**
- The permissions for this file are listed at the start of the line, starting with `rw-r`. This first set of symbols defines owner access. The next set of `wx` symbols defines group access. The last set of symbols defines access permitted for all other users.
- This listing indicates that the file is readable, writable, and executable by the user who owns the file (user emma) as well as the group owning the file (which is a group named student). The file is also world-readable and world-executable, but not world-writable.

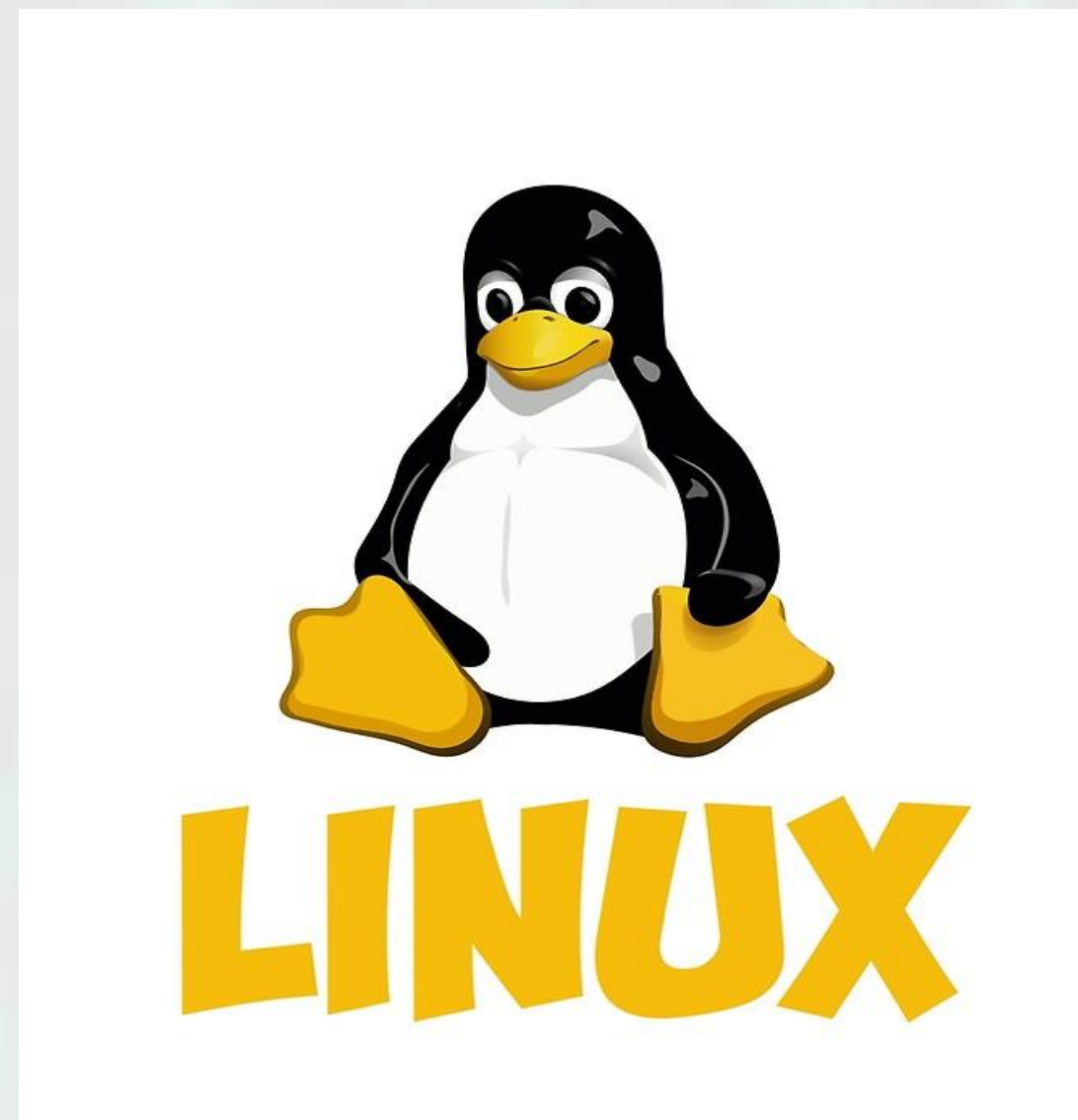
Basic Linux commands



Following are the common Linux commands –

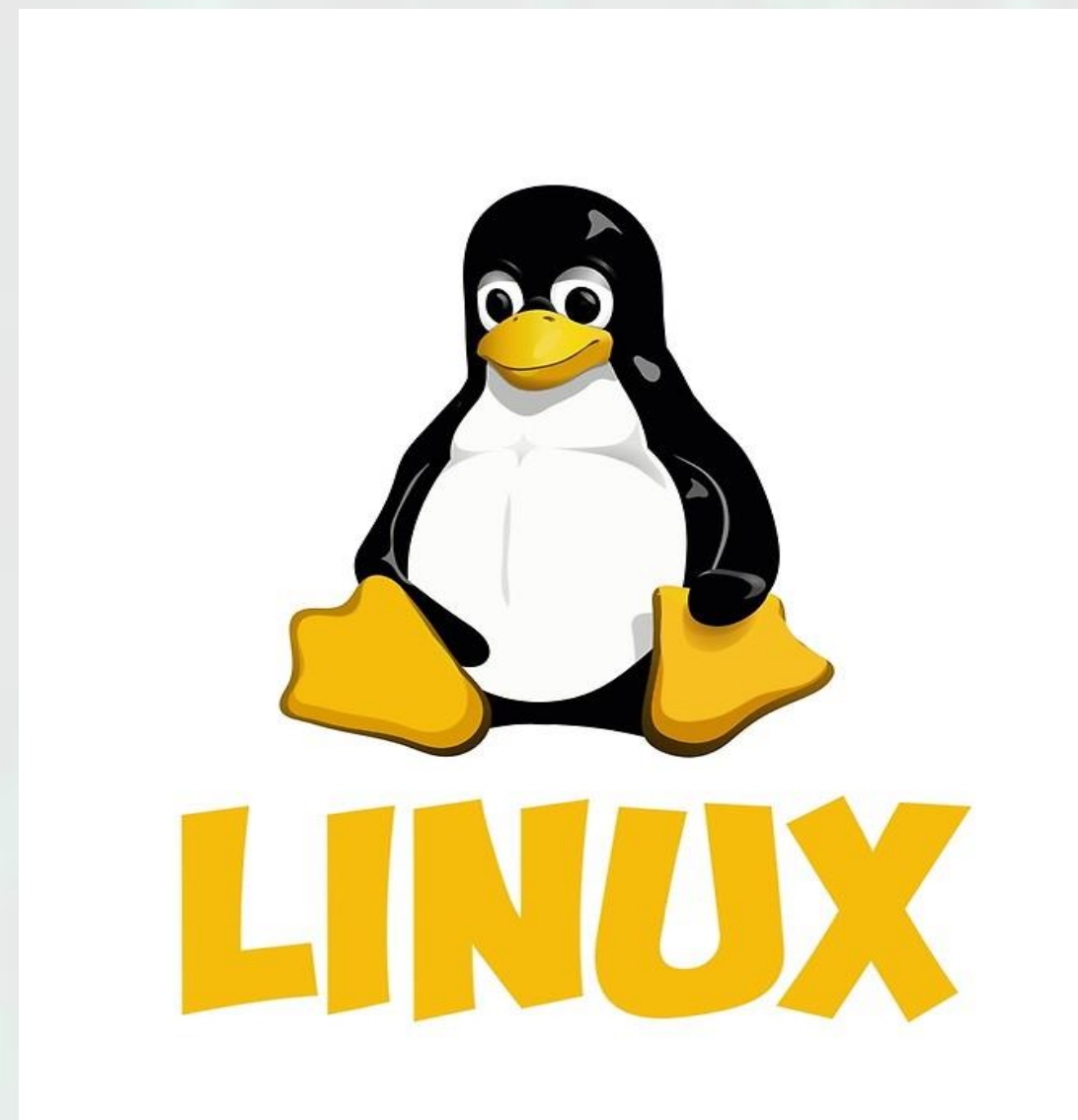
- **ls** - listing all the files in the directory
- **man** - manual for help regarding a particular command
- **cp** - copy a file from one location to another
- **mv** - move a file from one location to another
- **pwd** – to know the present working directory
- **cd** - change directory to a given path
- **mkdir** - to create a new directory
- **rmdir** - to remove a directory
- **chmod** - change the permissions of a file or directory

Basic Linux commands(2)



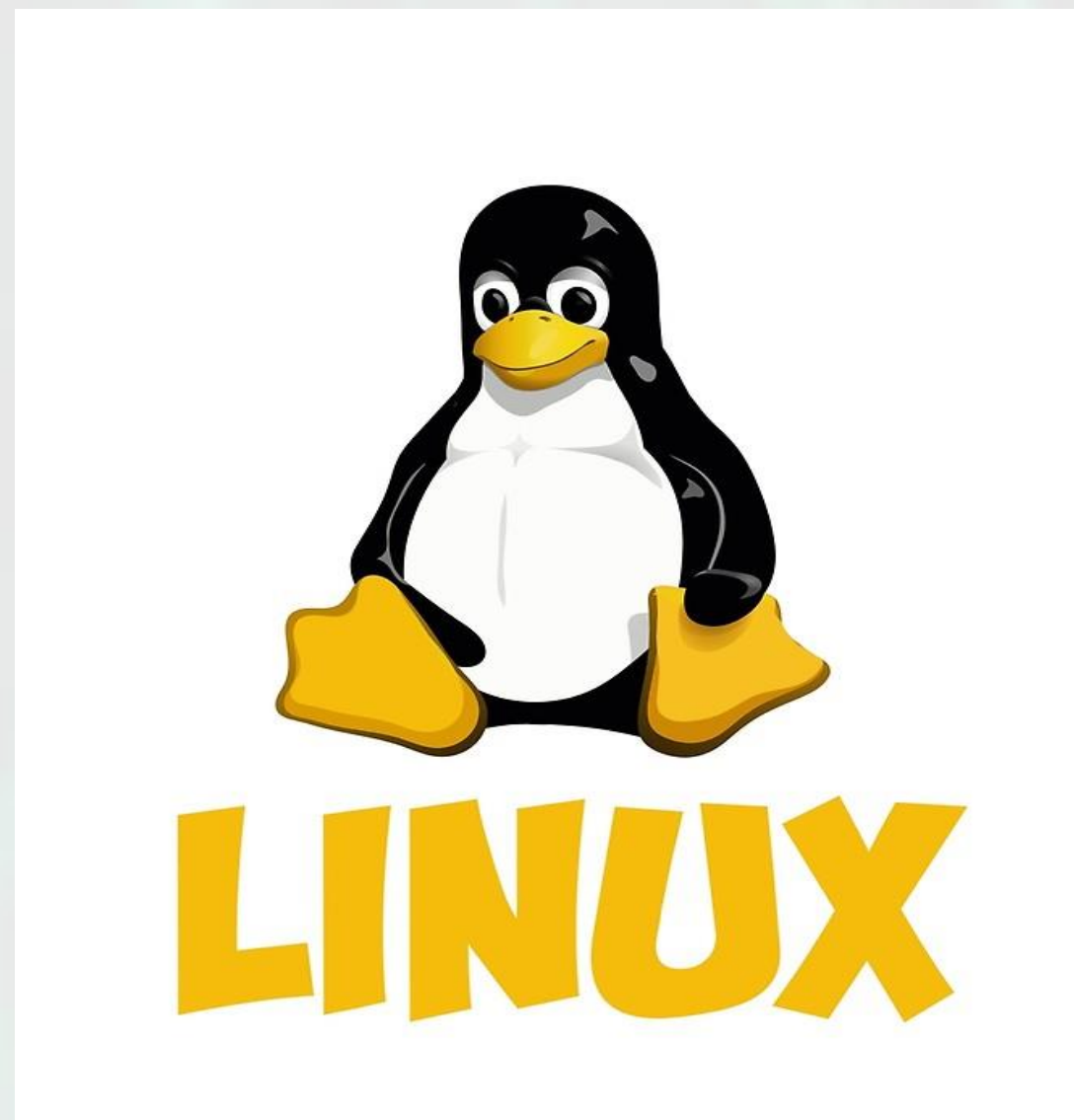
- **cat** - Dumping contents of file to standard output
- **less** - View a big file in parts by moving forward and backwards
- **head** - View first 10 lines of a file by default. `head -n50 file.txt` gives first 50 lines in file.txt
- **tail** - View last 10 lines of a file by default.
- **ps** - To view current running processes
- **top** - To view CPU usage of all the processes
- **kill** - to terminate a process using process id

vi editor



- The visual editor on the Linux.
- The vi editor comes with every version of Linux or Unix.
- Using vi is similar to using other editors in that you can see your file on the screen.
- The vi editor is the most popular editor in linux.
- The current version is really "vim", but to invoke it simply type "vi".
- The vi editor is not a text formatter (like MS Word, Word Perfect, etc.) you cannot set margins centre headings etc...)

vi editor commands



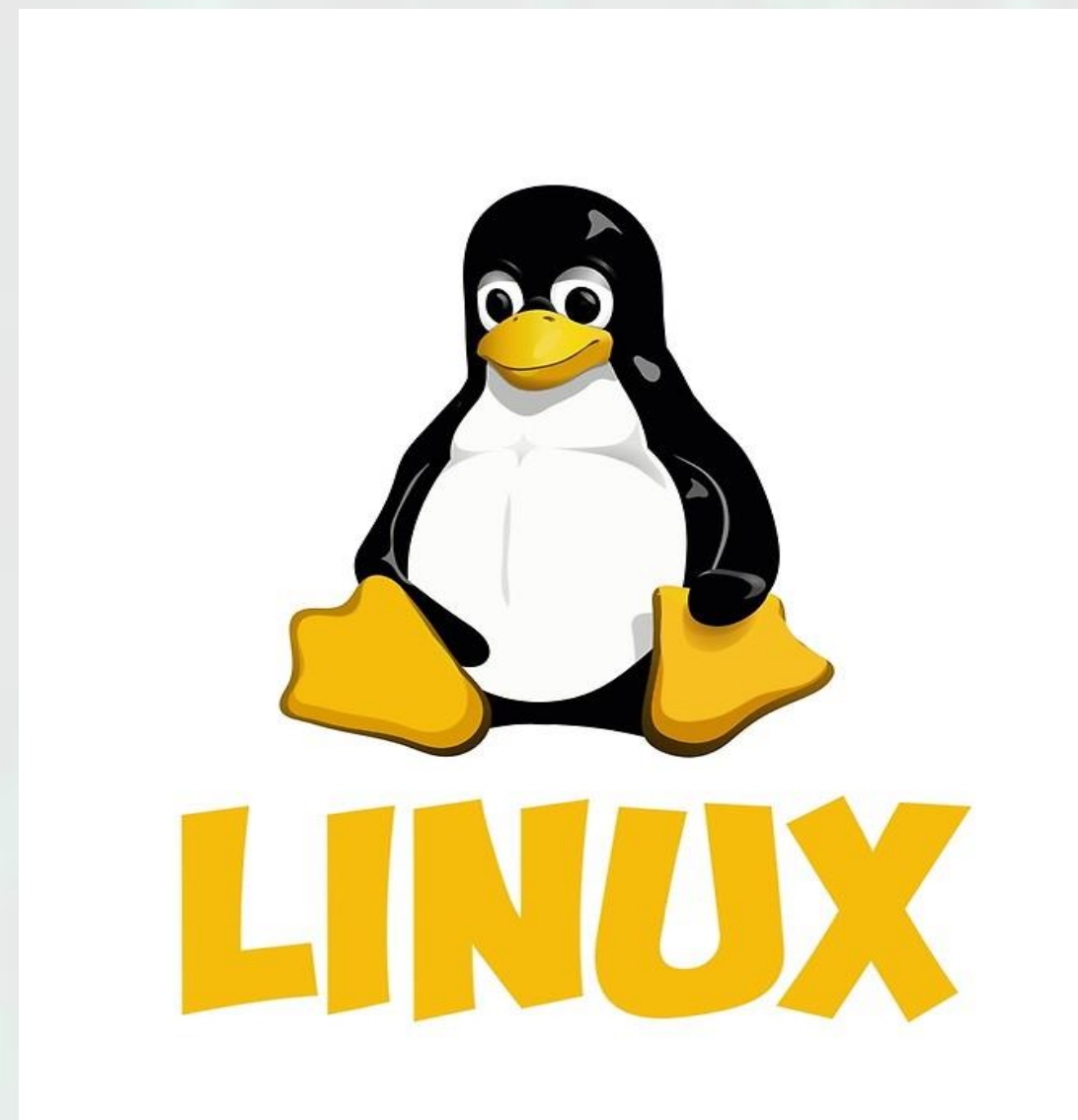
Open vi

- **vi** - opens vi
- **vi filename** - opens vi on file_name
- **view filename** - opens vi in read-only mode on file name

Close vi: press ESC key and

- **:q** - quit (it will query you if changes were made, y or n)
- **:q!** quit without saving
- **:x** - save and exit

Environment variables



- Environment variables are global settings that control the function of the shell and other Linux programs. They are sometimes referred to as global shell variables.
- Some Important Environment Variables –
 - HOME** - Your home directory (often abbreviated as "~")
 - TERM** - The type of terminal you are running (for example vt100, xterm, and ansi)
 - PWD** - Current working directory
 - PATH** - List of directories to search for commands

Master Executive di II Livello
BIG DATA ANALYSIS AND
BUSINESS INTELLIGENCE

Vamsi Krishna Varma Gunturi
Data science intern at ISTAT
vamsivarmagunturi@gmail.com

Grazie

fondazione

INOIT
TORVERGATA