

## Introduction to Linux: *A New Hope*

David Beserra \☺-☺-☺/



# Agenda

- Introduction
  - What is Linux?
  - Graphical Interfaces
  - Equivalent Software
- Directory Structure
- Essential Commands





# What is Linux?

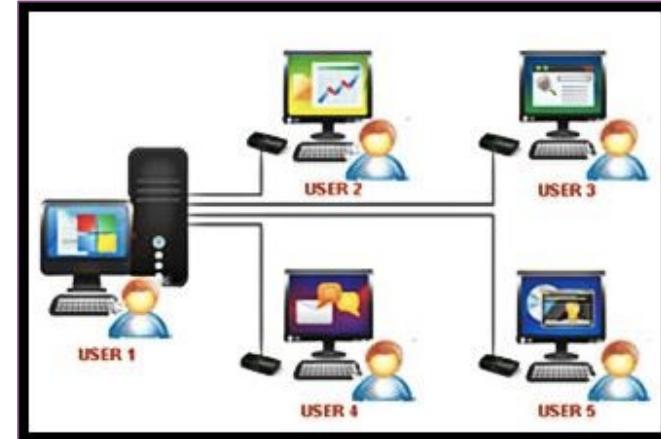
- Linux is an open-source operating system, a variant of Unix, that uses the GNU/GPL license.
- It was created in the early 1990s by a Finnish student named Linux Torvalds.
- Widely used in the academic and business worlds.





# What is Linux?

- **Multitasking** operating system: allows for the execution of different tasks simultaneously.
- **Multi-user** operating system: allows multiple users to access the system simultaneously (via terminals).



# +

# Graphical Interfaces

- The program responsible for the graphical part of Linux is XFree.
  - Note: XFree86 is a free and open-source implementation of the X Window System, which provides the graphical interface on Linux and other Unix-like operating systems.
- XFree contains specifications for creating and manipulating windows, providing tools for window managers to implement a graphical interface.
- Ease of creating new graphical interfaces.

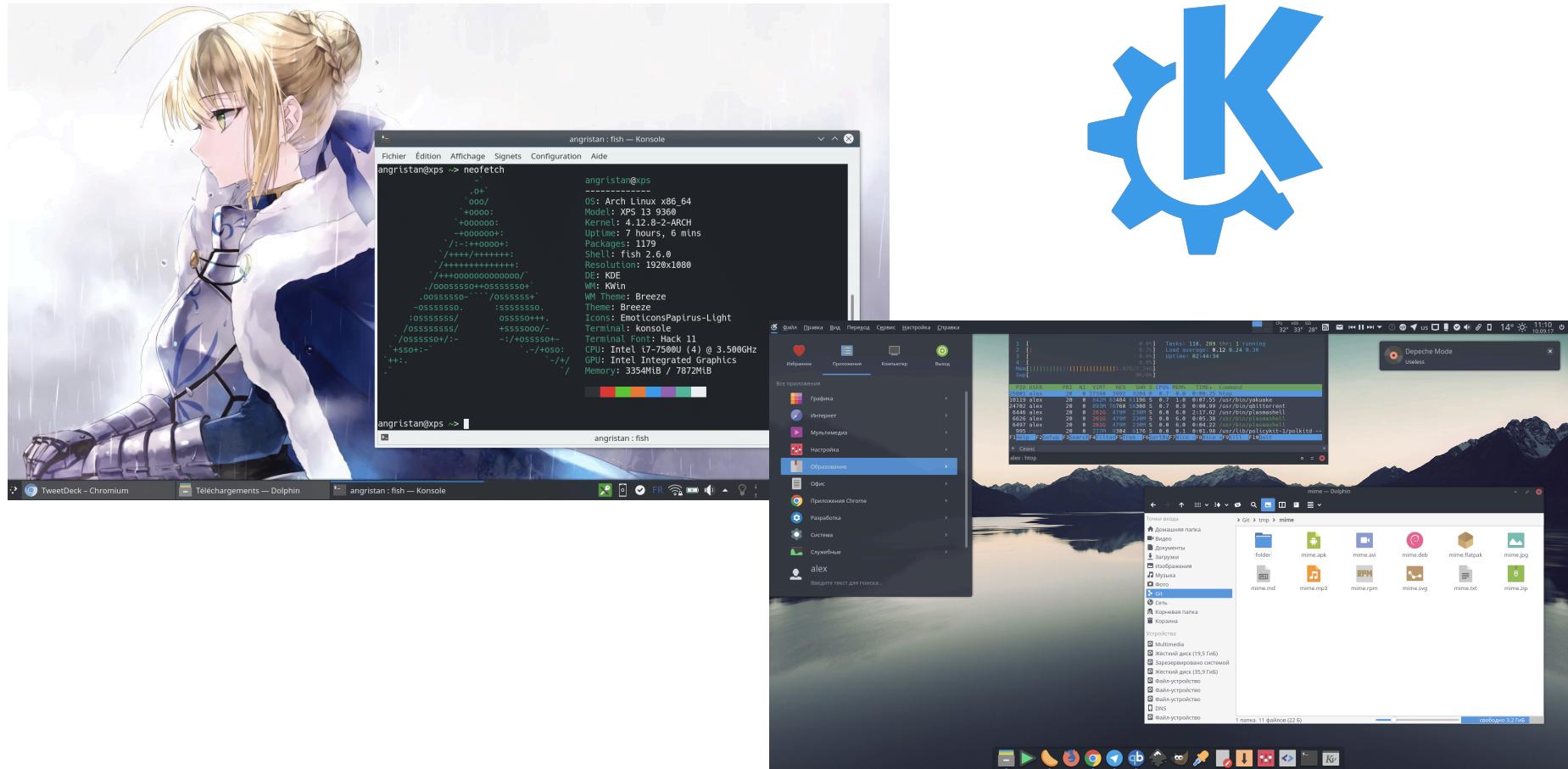
# + Graphical Interfaces

- It is possible to use several different graphical interfaces for Linux.
  - KDE
  - Gnome
  - Unity
  - Xfce
  - WindowMaker
  - Etc...



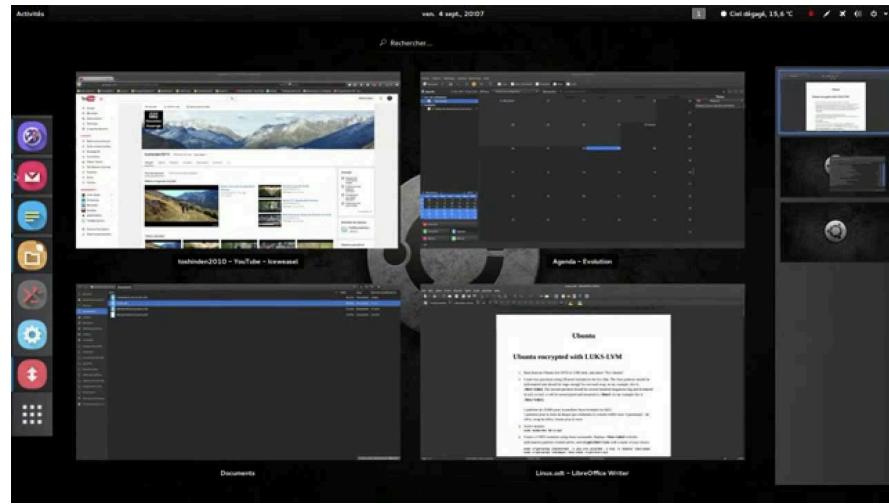
# + Graphical Interfaces

## ■ KDE



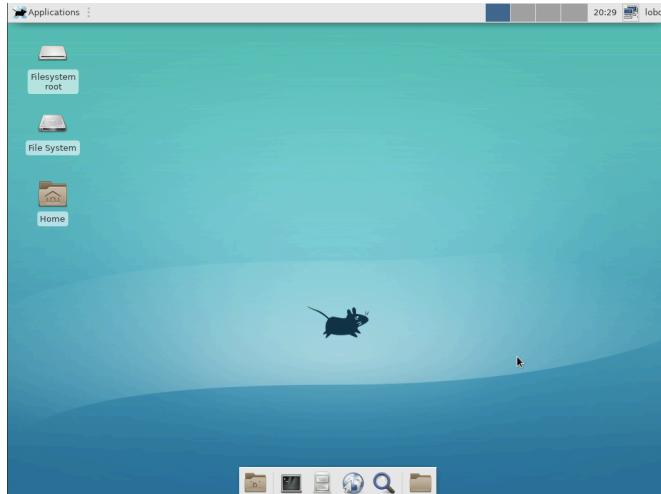
# + Graphical Interfaces

## ■ GNOME

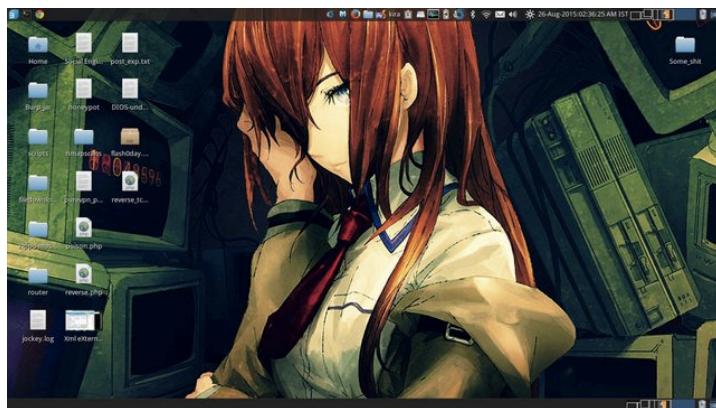
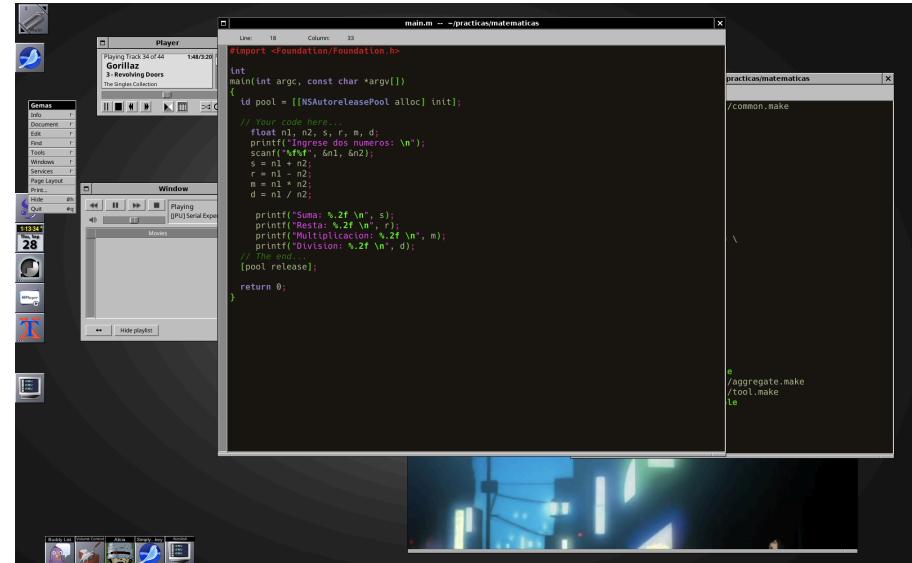


# + Graphical Interfaces

## ■ Xfce



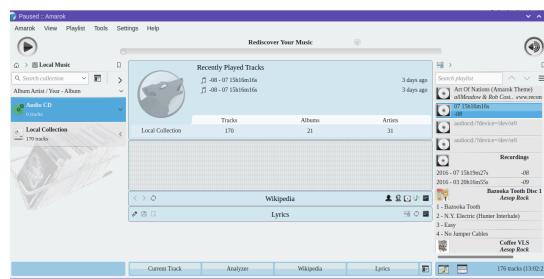
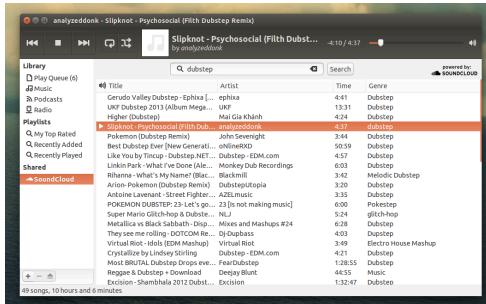
## ■ WindowMaker



# + Graphical Interfaces

## ■ Audio players

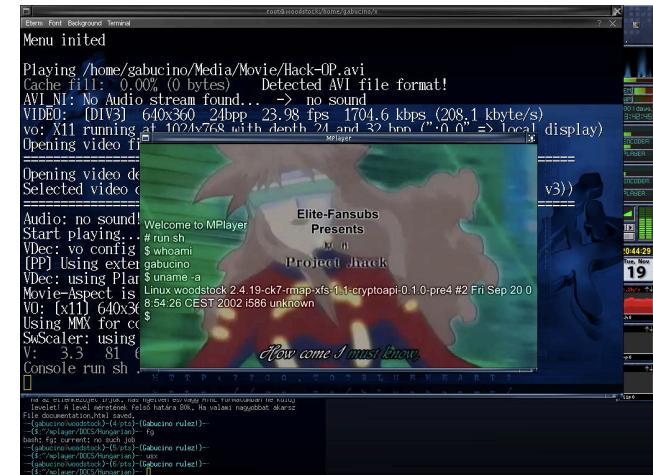
- Xmms (<http://www.xmms.org/>)
- Rhythmbox (<http://www.gnome.org/projects/rhythmbox/>)
- Amarok (<http://amarok.kde.org/>)
- BMPx ([http://bmpx.beep-media-player.org/site/BMPx\\_Homepage](http://bmpx.beep-media-player.org/site/BMPx_Homepage))



# + Graphical Interfaces

## ■ Video players

- Mplayer (<http://www.mplayerhq.hu/homepage/design7/news.html>)
- Totem (<http://www.gnome.org/projects/totem/>)
- Xine (<http://xinehq.de/>)
- VLC (<http://www.videolan.org/vlc/>)



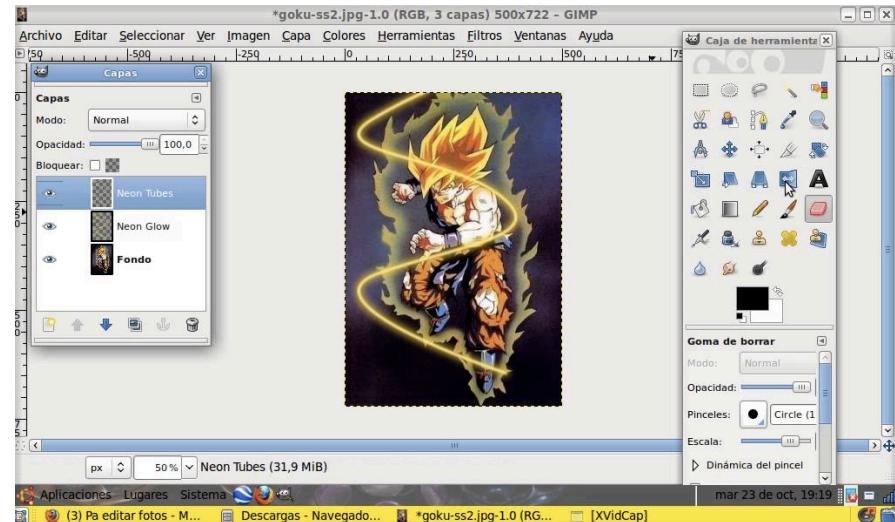
# + Equivalent Software

## ■ Image viewers.

- gThumb (<http://gthumb.sourceforge.net/>)
- Gwenview (<http://gwenview.sourceforge.net/>)
- digiKam (<http://www.digikam.org/>)

## ■ Image editors.

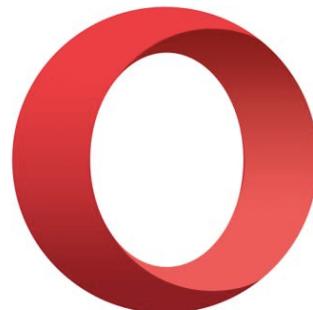
- Gimp (<http://gimp.org/>)



# + Equivalent Software

## ■ Browsers

- Firefox, Opera, Chromium, Chrome, Iceweasel, Konqueror, etc...



This is how Mozilla Firefox was created 😂



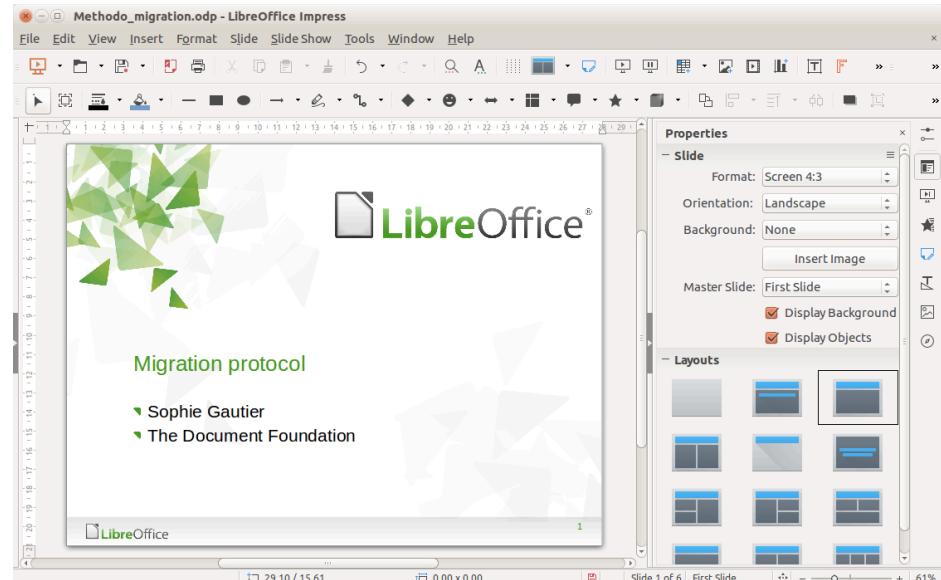
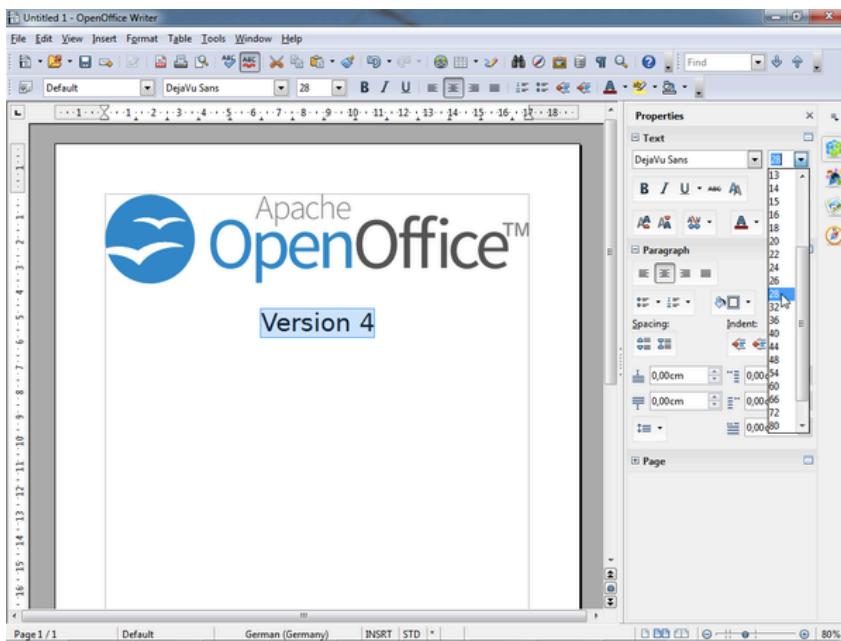
100%

# +

# Equivalent Software

## ■ Office

- Open Office
- Libre Office



# + Directory Structure.

/

- Root directory

/ home

- User's directory
- Ex: an user with username 'david' has the following home directory : /home/david

/ bin

- Contains programs (executables) that can be used by all users of the machine.
- 'bin' is a mnemonic for 'binaries', which are executable files.

# +

# Directory Structure.

- /usr
  - Contains other directories related to the user.
  
- /sbin
  - Stores files that are automatically executed by the operating system.

- /dev
  - /dev stores devices.
    - *What the hell?*
    - In Linux, devices are considered as files.
  - Exemples:
    - /dev/hd0 (a hard drive)
    - /dev/null (trash)

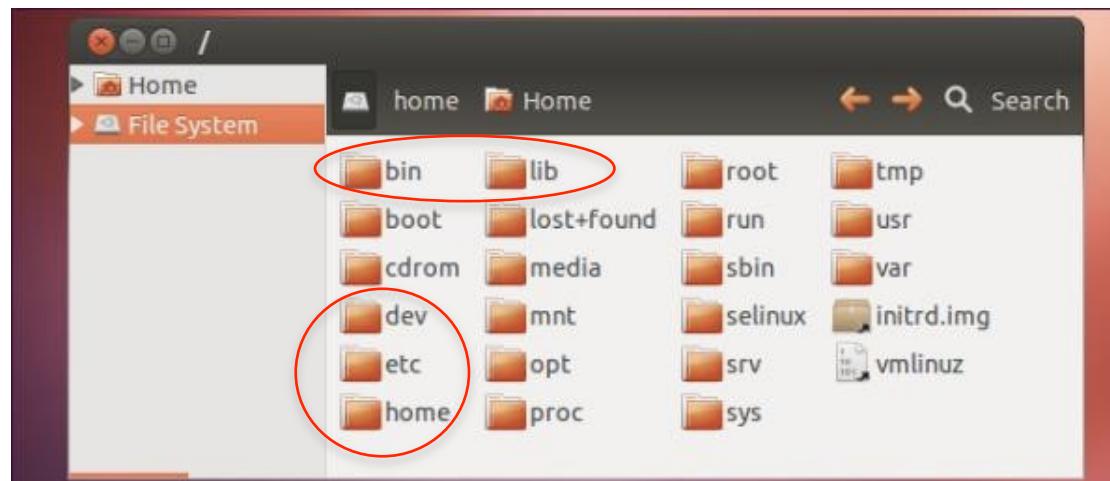
# + Directory Structure

## /etc

- Stores system configuration files.
- Text files for configuring system settings.

## /lib

- Folder containing shared libraries (usually .so files) used by programs.





# Ubuntu

- What is Ubuntu?
  - Ubuntu is a Linux-based operating system managed by Canonical Ltd.
  - It can be acquired by downloading it from the internet
    - <https://www.ubuntu.com/download>
- Ubuntu is updated every 6 months



# + Ubuntu

## ■ Variants

- Edubuntu: Ubuntu with interface Gnome
- Kubuntu: Ubuntu with interface KDE
- Xubuntu: Ubuntu with interface xfce (lighter)
- Ubuntu Touch: mobile version



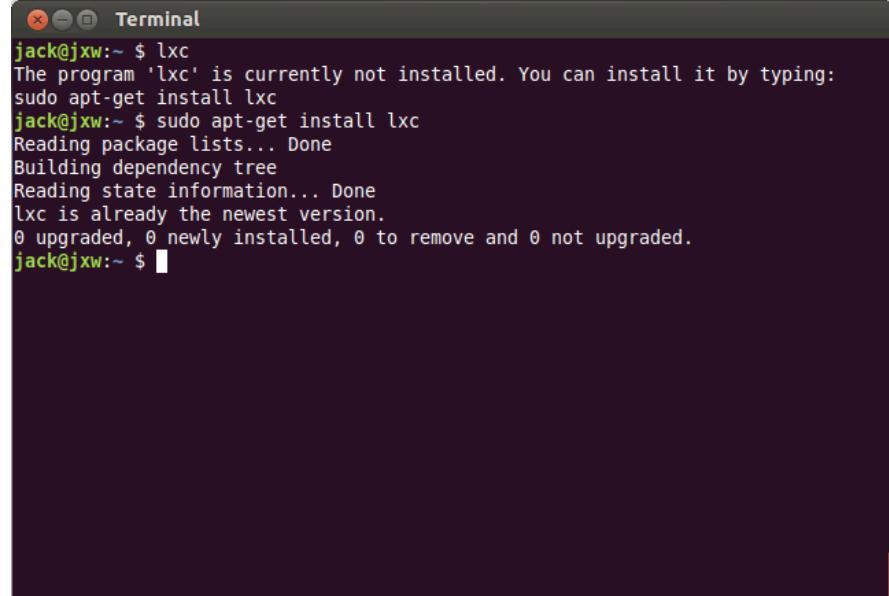


# Basic Commands

- Text mode commands in Linux:
  - They are special words that represent a certain action.
  - They usually have:
    - Options (characters preceded by '-')
    - Arguments (text strings or numbers)
  - Example:
    - ls -a
    - apt-get install
  - They are "invoked" in the b(or terminal).s

# + Commandes de base

- Shell is a command interpreter
- There are many shell types
  - **bash**
  - csh
  - tcsh
  - ksh
  - sh



```
jack@jxw:~ $ lxc
The program 'lxc' is currently not installed. You can install it by typing:
sudo apt-get install lxc
jack@jxw:~ $ sudo apt-get install lxc
Reading package lists... Done
Building dependency tree
Reading state information... Done
lxc is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
jack@jxw:~ $
```



# Basic Commands

## help

- Displays information about internal bash commands.

## man

- Displays a man page (manual page)

### Exemples:

- *man ls*
- *info ls*
- *help ls*

## info

- The "info" command in Linux is a program that provides a more organized and structured way to access documentation about various software packages and topics.

These commands are used to display the documentation of other commands





# Basic Commands

Commands for manipulating files.

## cd (*change directory*)

- **cd [directoryName]**
- Is used to access certain directories.
- Example:
  - cd /home/david/classes
- To go back to the previous directory, use cd ..
- To go back two levels, use cd ../../..

## pwd (*print working directory*)

- Displays the current working directory.

## ls

- Shows the contents of a directory.
- Options:
  - -a: displays all files, including hidden ones
  - -l: detailed listing
  - -h: displays file sizes in human-readable format (e.g. 1K, 2M, 3G)



# Basic Commands

Exemple: pwd, ls, cd

```
paris1@paris1-VirtualBox: ~
paris1@paris1-VirtualBox:~$ pwd
/home/paris1
paris1@paris1-VirtualBox:~$ ls /home/paris1/
auuuuu examples.desktop Modèles phones.txt Téléchargements
Bureau fichier Musique prova.sh Vidéos
compress.tar.gz glassfish4 oi Public
Documents Images partage script1.sh
paris1@paris1-VirtualBox:~$ ls
auuuuu examples.desktop Modèles phones.txt Téléchargements
Bureau fichier Musique prova.sh Vidéos
compress.tar.gz glassfish4 oi Public
Documents Images partage script1.sh
paris1@paris1-VirtualBox:~$ cd /
paris1@paris1-VirtualBox:/$ pwd
/
paris1@paris1-VirtualBox:/$ ls /
bin cours home lib64 mnt root snap tmp vmlinuz
boot dev initrd.img lost+found opt run srv usr
cdrom etc lib media proc sbin sys var
paris1@paris1-VirtualBox:/$ cd /home/paris1/
paris1@paris1-VirtualBox:~$ pwd
/home/paris1
paris1@paris1-VirtualBox:~$
```

# + Basic Commands

Commands for manipulating files and directories

## mkdir

- "mkdir [directoryName]"
- is used to create a new directory.
- Example: "mkdir EPITA".

## rmdir

- "rmdir [directoryName]"
- is used to remove an **empty** directory.
- Example: "rmdir EPITA".

## cp (copy)

- "cp [options] [source] [destination]"
- is used to copy files and directories from one location to another, with the same name or a different name.
- Example: "cp /home/david/file /home/beserra/file2".



# Commandes de base

Exemple: mkdir, rmdir, cp

```
paris1@paris1-VirtualBox: ~/Documents
paris1@paris1-VirtualBox:~/Documents$ pwd
/home/paris1/Documents
paris1@paris1-VirtualBox:~/Documents$ ls
paris1@paris1-VirtualBox:~/Documents$ mkdir amenouche
paris1@paris1-VirtualBox:~/Documents$ ls
amenouche
paris1@paris1-VirtualBox:~/Documents$ rmdir amenouche
paris1@paris1-VirtualBox:~/Documents$ ls
paris1@paris1-VirtualBox:~/Documents$ cp /home/paris1/oi oi
paris1@paris1-VirtualBox:~/Documents$ ls
oi
paris1@paris1-VirtualBox:~/Documents$ cp /home/paris1/oi ola
paris1@paris1-VirtualBox:~/Documents$ ls
oi ola
paris1@paris1-VirtualBox:~/Documents$ █
```



# Commandes de base

Commands for manipulating files and directories

**mv**

- "mv [options] [destination]"
- is used to move or rename a file.
- Example: "mv texts.txt text2.txt".

**touch**

- "touch [fileName]"
- is used to create an empty file.
- Example: "touch file".

**cat**

- "cat [fileName]"
- displays the contents of a file in the shell.

**id**

- "id" provides information about the current user, including the login name, user ID, group name, and group ID.



# Commandes de base

Exemple: mv, cat, touch, id

```
paris1@paris1-VirtualBox: ~/Documents
mparis1@paris1-VirtualBox:~/Documents$ mkdir berrada
paris1@paris1-VirtualBox:~/Documents$ ls
berrada oi ola
paris1@paris1-VirtualBox:~/Documents$ cp oi berrada/oi; ls
berrada oi ola
paris1@paris1-VirtualBox:~/Documents$ ls berrada
oi
paris1@paris1-VirtualBox:~/Documents$ mv ola berrada/ola; ls
berrada oi
paris1@paris1-VirtualBox:~/Documents$ ls berrada
oi ola
paris1@paris1-VirtualBox:~/Documents$ cat berrada/oi
En fait, je me demande si quelqu'un a essayé de vérifier quoi veut dire "oi".

paris1@paris1-VirtualBox:~/Documents$ touch bunaZiua
paris1@paris1-VirtualBox:~/Documents$ ls
berrada bunaZiua oi
paris1@paris1-VirtualBox:~/Documents$ id
uid=1000(paris1) gid=1000(paris1) groupes=1000(paris1),4(adm),24(cdrom),27(sudo)
,30(dip),46(plugdev),113(lpadmin),128(sambashare)
paris1@paris1-VirtualBox:~/Documents$ █
```

# Basic Commands



## Commandes pour manipuler des fichiers

### rm

- "rm [fileName/directoryName] »
- is used to delete a file or directory.
- To delete recursively, use "rm -rf".
- With this command, you can also delete non-empty directories ;)

### more

- "more [fileName]"
- displays files in a paginated format.
- To exit, press "q".

### tail

- "tail [options] <file> »
- Displays the last few lines of a file.
- Options:
- -n: displays the last 'n' lines of a file
- -n +x: displays all the lines of a file starting from line x.

### less

- "less [fileName]"
- displays files in a way similar to "more".
- For compressed files (.gz), you can use "zless".



# Basic Commands

Exemple: rm

```
paris1@paris1-VirtualBox: ~/Documents
paris1@paris1-VirtualBox:~/Documents$ ls
berrada bunaZiua oi
paris1@paris1-VirtualBox:~/Documents$ rm bunaZiua
paris1@paris1-VirtualBox:~/Documents$ ls
berrada oi
paris1@paris1-VirtualBox:~/Documents$ rmdir berrada/
rmdir: échec de suppression de 'berrada/': Le dossier n'est pas vide
paris1@paris1-VirtualBox:~/Documents$ rm berrada/
rm: impossible de supprimer 'berrada/': est un dossier
paris1@paris1-VirtualBox:~/Documents$ rm -rf berrada/
paris1@paris1-VirtualBox:~/Documents$ ls
oi
paris1@paris1-VirtualBox:~/Documents$ █
```



# Commandes de base

## Exemple: tail, more

```
paris1@paris1-VirtualBox:~/Documents$ ls
oi
paris1@paris1-VirtualBox:~/Documents$ tail -n 2 oi
Depuis les primordes de l'histoire humaine, l'homme a essaye de dire n'importe quoi. Cette message
Je sais tres bien que ce texte est depourvu de signification. Et alors? On s'en fiche! Ole
paris1@paris1-VirtualBox:~/Documents$ tail -n 1 oi
Je sais tres bien que ce texte est depourvu de signification. Et alors? On s'en fiche! Ole
paris1@paris1-VirtualBox:~/Documents$ tail oi
En fait, je me demande si quelqu'un a essayé de vérifier quoi veut dire "oi".
De fato, eu me pergunto se alguém tentou verificar o que significa "coucou".
De hecho, yo me pregunto si alguien ha intentado verificar lo que significa "hi"
In fact, I ask to myself if someone tried to figure out what means "hola"

Depuis les primordes de l'histoire humaine, l'homme a essaye de dire n'importe quoi. Cette message
Je sais tres bien que ce texte est depourvu de signification. Et alors? On s'en fiche! Ole
paris1@paris1-VirtualBox:~/Documents$ more oi
En fait, je me demande si quelqu'un a essayé de vérifier quoi veut dire "oi".
De fato, eu me pergunto se alguém tentou verificar o que significa "coucou".
De hecho, yo me pregunto si alguien ha intentado verificar lo que significa "hi"
In fact, I ask to myself if someone tried to figure out what means "hola"

Depuis les primordes de l'histoire humaine, l'homme a essaye de dire n'importe quoi. Cette message
Je sais tres bien que ce texte est depourvu de signification. Et alors? On s'en fiche! Ole
paris1@paris1-VirtualBox:~/Documents$ █
```



# Commandes de base

## Commandes pour manipuler des fichiers

- "Locate" command is used to search for files.
  - **locate [fileName]**
  - Before using the "locate" command, it is necessary to call another command "updatedb" (to update the database used by locate).
- "whereis" command is used to find binary files, source code, and command manual pages.
- Some options include:
  - -b: search for binary files only
  - -m: search for source code only
  - -s: search for manual pages only
  - -l: display a list of search paths used by whereis.





# Commandes de base

Exemple: locate, whereis

A screenshot of a terminal window titled "paris1@paris1-VirtualBox: ~/Documents". The window contains the following text:

```
paris1@paris1-VirtualBox:~/Documents$ locate auuuuuu
/home/paris1/auuuuuu
paris1@paris1-VirtualBox:~/Documents$ whereis grep
grep: /bin/grep /usr/share/man/man1/grep.1.gz /usr/share/info/grep.info.gz
paris1@paris1-VirtualBox:~/Documents$ whereis ls
ls: /bin/ls /usr/share/man/man1/ls.1.gz
paris1@paris1-VirtualBox:~/Documents$ █
```



# Basic Commands

Commands for process management.

- ps
- Displays the active processes on the computer and shows which user a process belongs to.
- The option "-u" is used to display the user information of the processes. The full command would be:
- "ps -u"
- This will display a list of active processes on the system, along with information about the user to which each process belongs.



# Basic Commands

Commands for process management.

## ■ ps (example)

The screenshot shows a terminal window titled "root@vz226b.liquidweb.com:/root — ssh — 108x28". The command "ps aux" is run, displaying a list of processes. The output is as follows:

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.0	1412	480	?	S	12:38	0:00	init [3]
root	13644	0.0	0.1	2060	1104	?	S	12:38	0:00	/bin/bash /etc/rc.d/rc 3
root	13996	0.0	0.0	1480	500	?	S	12:38	0:00	syslogd -m 0
named	14010	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14011	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14012	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14013	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14014	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14015	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14016	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
named	14017	0.0	0.2	19100	2948	?	S	12:38	0:00	/usr/sbin/named -u named
root	14028	0.0	0.0	1404	360	?	S	12:38	0:00	/usr/sbin/courierlogger -pid=/var/spool/authd
root	14029	0.0	0.0	1736	548	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	14036	0.0	0.0	1736	168	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	14037	0.0	0.0	1736	168	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	14038	0.0	0.0	1736	168	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	14039	0.0	0.0	1736	168	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	14040	0.0	0.0	1736	168	?	S	12:38	0:00	/usr/libexec/courier-authlib/authdaemond
root	32256	0.0	0.0	3532	984	?	S	12:39	0:00	/usr/sbin/sshd
root	32265	0.0	0.0	2060	1016	?	S	12:39	0:00	/bin/sh /usr/bin/mysqld_safe --datadir=/var/l
mysql	32286	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l
mysql	32289	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l
mysql	32290	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l
mysql	32291	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l
mysql	32292	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l
mysql	32293	0.0	1.4	64668	14884	?	S	12:39	0:00	/usr/sbin/mysqld --basedir=/ --datadir=/var/l



# Basic Commands

Commands for process management.

- "top"
- displays the currently running programs, their usage time, CPU usage, and many other pieces of information.
- To exit the screen, press "q" (quit).

```
root@host:~#
top - 08:29:17 up 23 days, 3:00, 1 user, load average: 1.01, 1.14, 1.20
Tasks: 221 total, 2 running, 219 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.5 us, 0.5 sy, 0.1 ni, 90.8 id, 1.1 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 10074248 total, 788400 free, 4057348 used, 5228500 buff/cache
KiB Swap: 5177340 total, 2636084 free, 2541256 used. 5442796 avail Mem

      PID USER      PR  NI    VIRT    RES   SHR S %CPU %MEM     TIME+ COMMAND
 3979 [REDACTED]  20   0 505108 51524 19984 S 18.3  0.5  0:00.55 php-fpm
 3977 [REDACTED]  20   0 504968 51872 20220 S 17.6  0.5  0:01.12 php-fpm
 3920 [REDACTED]  20   0 508252 59048 24020 S 17.3  0.6  0:06.63 php-fpm
 3945 [REDACTED]  20   0 508520 57948 22724 S 5.6  0.6  0:04.43 php-fpm
32295 mysql     20   0 5685172 1.880g 6252 S 5.0 19.6 1198:20 mysqld
1237 root      25   5 1047504 133440 3860 S 3.7  1.3 136:48.90 cdp-2-6
 9 root      20   0     0     0     0 S 0.3  0.0  81:45.27 rcu_sched
1236 root      0 -20   0     0     0 S 0.3  0.0  10:40.33 hcp_watchdog
1390 root      20   0 1013492 129108 444 S 0.3  1.3  0:00.17 hcp_io/1/1
1397 root      20   0 1013492 129108 444 S 0.3  1.3  0:00.18 hcp_io/1/6
1488 nobody    20   0 2046816 20512 7420 S 0.3  0.2  0:04.79 httpd
1516 nobody    20   0 2046816 20132 7380 S 0.3  0.2  0:04.46 httpd
1547 nobody    20   0 2046816 16004 2992 S 0.3  0.2  0:04.58 httpd
1665 nobody    20   0 2046816 19344 7408 S 0.3  0.2  0:03.83 httpd
3978 root      20   0 155792 2228 1472 R 0.3  0.0  0:00.03 top
4749 cpanel+  20   0 4731944 144624 1856 S 0.3  1.4  28:07.74 java
23165 cpanelc+ 20   0 11564 4664 4356 S 0.3  0.0  16:05.41 p0f
 1 root      20   0 191188 3084 1736 S 0.0  0.0  12:11.19 systemd
 2 root      20   0     0     0     0 S 0.0  0.0  0:01.66 kthreadd
 3 root      20   0     0     0     0 S 0.0  0.0  0:05.80 kssoftirqd/0
 5 root      0 -20   0     0     0 S 0.0  0.0  0:00.00 kworker/0:0H
 7 root      rt  0     0     0     0 S 0.0  0.0  0:12.37 migration/0
 8 root      20   0     0     0     0 S 0.0  0.0  0:00.00 rcu_bh
10 root      rt  0     0     0     0 S 0.0  0.0  0:07.78 watchdog/0
11 root      rt  0     0     0     0 S 0.0  0.0  0:06.89 watchdog/1
12 root      rt  0     0     0     0 S 0.0  0.0  0:04.42 migration/1
```



# Commandes de base

Commands for process management.

## kill

- "kill [options] [pid]"
  - The option "-9" sends a "kill" signal to a process, which terminates the process immediately without giving it a chance to save any data.
- You can view the process ID (PID) of each process using the "ps" command. For example:
  - "ps aux | grep [processName]"
  - This will display a list of all the processes running on the system that match the specified process name, along with their PIDs.
  - Once you have identified the PID of the process you want to terminate, you can use the "kill" command to terminate it.

```
howto geek@ubuntu:~$ kill firefox
bash: kill: firefox: arguments must be process or job IDs
howto geek@ubuntu:~$ ps -A | grep firefox
 3684 ?        00:00:10 firefox
howto geek@ubuntu:~$ kill 3684
howto geek@ubuntu:~$
```



# Commandes de base

- ">" and ">>" are used to redirect the output of a command to a file.
- ">" overwrites the contents of the file with the output of the command, while ">>" appends the output to the end of the file.
- Here is an example of how to use these commands:
  - "command > outputFile.txt"
    - This will run the "command" and write the output to "outputFile.txt", overwriting the file if it already exists.
  - "command >> outputFile.txt"
    - This will run the "command" and append the output to "outputFile.txt", creating the file if it doesn't exist.
- One use case for this is to store the output of an installation in a file, to check for possible compilation errors.



# Commandes de base

Exemple: >, >>

```
paris1@paris1-VirtualBox: ~/Documents
paris1@paris1-VirtualBox:~/Documents$ cat fichier
Ici je parle n'importe quoi
et ici aussi, haha
paris1@paris1-VirtualBox:~/Documents$ ls /home >> fichier
paris1@paris1-VirtualBox:~/Documents$ cat fichier
Ici je parle n'importe quoi
et ici aussi, haha
etudiant
paris1
paris1@paris1-VirtualBox:~/Documents$ ls /home/etudiant > fichier
paris1@paris1-VirtualBox:~/Documents$ cat fichier
exemple01.txt
exemple02.txt
exemple03.txt
feuille-de-calcul
personalfile.doc
paris1@paris1-VirtualBox:~/Documents$
```



# Basic Commands -Metacaracteres

- Metacharacters are used to simplify the coding and execution of commands, and make it easier to manage files and directories in the operating system.
- There are several metacharacters used in Linux, including:
  - \* (wildcard): matches any number of characters in a filename or directory name.
  - ? (question mark): matches any single character in a filename or directory name.
  - [ ] : matches any character in a range of characters specified inside the brackets.
  - { } (curly braces): used for command grouping and expansion.
- For example, the command "ls \*.txt" will list all files in the current directory that end with ".txt".



# Basic Commands - Metacaracteres

\*:

■ Exemple:

- # ls \*
- # cours1.txt cours2.txt  
cours3.txt cours4.txt

[ ] et -

■ Exemple:

- # ls cours[2-4].txt
- # cours2.txt cours3.txt  
cours4.txt



# Basic Commands - Metacaracteres

:

## ■ Exemple:

- # ls cours[2:4].txt
- # cours2.txt cours4.txt

?

## ■ Exemple:

- # ls cours[2:4].???
- # cours2.txt cours4.txt



# Commandes de base

## Exemple: metacaracteres

```
paris1@paris1-VirtualBox: /home/etudiant
paris1@paris1-VirtualBox:/home/etudiant$ ls
exemple01.txt  exemple03.txt      personalfile.doc
exemple02.txt  feuille-de-calcul
paris1@paris1-VirtualBox:/home/etudiant$ ls *
exemple01.txt  exemple03.txt      personalfile.doc
exemple02.txt  feuille-de-calcul
paris1@paris1-VirtualBox:/home/etudiant$ ls *.txt
exemple01.txt  exemple02.txt  exemple03.txt
paris1@paris1-VirtualBox:/home/etudiant$ ls *.*??
exemple01.txt  exemple02.txt  exemple03.txt  personalfile.doc
paris1@paris1-VirtualBox:/home/etudiant$ ls exemple0[2:3].txt
exemple02.txt  exemple03.txt
paris1@paris1-VirtualBox:/home/etudiant$ █
```



# Basic Commands

## wc

- # wc
- Yes, "wc" is a command that can be used to count the number of lines, words, or bytes in a file.

## df

- # df
- Displays the partitions in use.

## clear

- # clear
- Clears the visible content in the shell.

## du

- # du
- Displays the size of a directory.



# Basic Commands

Exemple: wc, du, df

```
paris1@paris1-VirtualBox:~/ionescu$ cat hello.txt
Voici une phrase écrite n'importe comment
Voici une autre phrase écrite n'importe comment
paris1@paris1-VirtualBox:~/ionescu$ wc -w hello.txt
13 hello.txt
paris1@paris1-VirtualBox:~/ionescu$ wc hello.txt
 2 13 92 hello.txt
paris1@paris1-VirtualBox:~/ionescu$ du /home/paris1/ionescu/
8      /home/paris1/ionescu/
paris1@paris1-VirtualBox:~/ionescu$ df
Sys. de fichiers            blocs de 1K  Utilisé Disponible Utile Monté sur
udev                           1002940      0   1002940    0% /dev
tmpfs                          204836     3716   201120    2% /run
/dev/sda1                      18446076  7242000  10244028   42% /
tmpfs                          1024164    10228  1013936    1% /dev/shm
tmpfs                           5120        4    5116    1% /run/lock
tmpfs                          1024164      0   1024164    0% /sys/fs/cgroup
[Reaktor]_Space_Adventure_Cobra_TV_[1080p][x265][10-bit]  487358464 265404288  221954176   55% /media/sf_[Reakto
r]_Space_Adventure_Cobra_TV_[1080p][x265][10-bit]
tmpfs                          204836      92   204744    1% /run/user/1000
paris1@paris1-VirtualBox:~/ionescu$
```



# Basic Commands

`free`

- Displays the amount of available RAM.

`history`

- Displays the last executed commands.

`date`

- Displays the current date and time.

# + Commandes de base

Exemple: free, date, history

```
paris1-VirtualBox: ~/ionescu
paris1@paris1-VirtualBox:~/ionescu$ free
      total        utilisé         libre      partagé   tamp/cache  disponible
Mem:    2048332       1487724      136176      28744      424432      355072
Partition d'échange:  2095100      169300     1925800
paris1@paris1-VirtualBox:~/ionescu$ free -m
      total        utilisé         libre      partagé   tamp/cache  disponible
Mem:      2000          1452          132          28          414          346
Partition d'échange:  2045          165        1880
paris1@paris1-VirtualBox:~/ionescu$ date
mardi 7 avril 2020, 14:25:32 (UTC+0200)
paris1@paris1-VirtualBox:~/ionescu$ history
 1  cd jdk1.8.0_111
 2  ls
 3  sudo add-apt-repository ppa:webupd8team/java
 4  sudo apt-get update
 5  sudo apt-get install oracle-java8-installer
 6  sudo apt install oracle-java8-set-default
 7  sudo reboot
 8  free
 9  java --version
10  sudo apt-get install oracle-java8-installer
11  sudo apt install oracle-java8-set-default
12  java --version
13  java -version
14  javac -version
15  ls
16  cd Téléchargements
17  ls
18  unzip glassfish-4.1.1*zip
19  glassfish4/bin/asadmin start-domain
20  cd
```



# Basic Commands

**ATENTION: THIS COMMAND IS VERY IMPORTANT!**

## ■ sort

- The "sort" command allows to sort the lines of a file.
- Exemples

```
sort « file »          (normal sort)
sort -r « file »       (reverse sort)
sort -k2 -t: /etc/passwd   (uses the second field of the file to sort it, and use
                           ':' as the delimiter.)
Sort -k2 -t: -n /etc/passwd (Same as before, but with numeric sorting.)
```

# Basic Commands

## Exemple:sort

```
paris1-VirtualBox: ~/ionescu
paris1@paris1-VirtualBox:~/ionescu$ cat agenda.txt

Piccolo-Daimaoh : 756 : guerrier.nakusejin@namek.nk
Goku-Son : 8001 : goku@dbz.jp
David-Beserra : 666 : david@beserra.edu.br
Jhonny-LeBavard : 070 : jhonny.bavard@crazyman.fr
Pepito-ElBueno : 099 : pepito.es.bueno@hotmail.es
Madh-Lheine : 1992 :mdl@stu.de
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
paris1@paris1-VirtualBox:~/ionescu$ sort agenda.txt

David-Beserra : 666 : david@beserra.edu.br
Goku-Son : 8001 : goku@dbz.jp
Jhonny-LeBavard : 070 : jhonny.bavard@crazyman.fr
Madh-Lheine : 1992 :mdl@stu.de
Pepito-ElBueno : 099 : pepito.es.bueno@hotmail.es
Piccolo-Daimaoh : 756 : guerrier.nakusejin@namek.nk
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
paris1@paris1-VirtualBox:~/ionescu$ sort -r agenda.txt
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
Piccolo-Daimaoh : 756 : guerrier.nakusejin@namek.nk
Pepito-ElBueno : 099 : pepito.es.bueno@hotmail.es
Madh-Lheine : 1992 :mdl@stu.de
Jhonny-LeBavard : 070 : jhonny.bavard@crazyman.fr
Goku-Son : 8001 : goku@dbz.jp
David-Beserra : 666 : david@beserra.edu.br

paris1@paris1-VirtualBox:~/ionescu$ sort -k2 -t: -n agenda.txt

Jhonny-LeBavard : 070 : jhonny.bavard@crazyman.fr
Pepito-ElBueno : 099 : pepito.es.bueno@hotmail.es
David-Beserra : 666 : david@beserra.edu.br
Piccolo-Daimaoh : 756 : guerrier.nakusejin@namek.nk
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
Madh-Lheine : 1992 :mdl@stu.de
Goku-Son : 8001 : goku@dbz.jp
```



# Basic Commands

## ■ cut

- This command is used to cut fields to select what you want from a file.

## ■ Ex:

```
david@senjyogahara: cat list-of-smartphones-2011.txt
Model:Company:Price:Camera:4G
iPhone4:Apple:1000$:Yes:Yes
Galaxy:Samsung:900$:Yes:Yes
Optimus:LG:800$:Yes:Yes
Sensation:HTC:400$:Yes:Yes
iPhone4S:Apple:1100:Yes:Yes
N9:Nokia:400:Yes:Yes
```

```
$ cut -d: -f1 list-of-smartphones-2011.txt
Model
iPhone4
Galaxy
Optimus
Sensation
iPhone4S
N9

$ sed 's/:/\t/g' list-of-smartphones-2011.txt | cut -f 1
Model
iPhone4
Galaxy
Optimus
Sensation
iPhone4S
N9

$ cut -c 1-9 list-of-smartphones-2011.txt
Model:Com
iPhone4:A
Galaxy:Sa
Optimus:L
Sensation
iPhone4S:
N9:Nokia:
```



# Basic Commands

- **grep**
  - Global Regular Expression Pattern match
  - On peut utiliser pour chercher des strings ou des expressions régulières dans un fichier.
- Ex: chercher la string "david" dans un fichier
  - Ex: grep "david" /etc/passwd
- **ATTENTION:** les metacaractères utilisés par grep ne sont pas nécessairement les mêmes qu'on utilise pour les autres commandes Linux.
  - <https://www.javatpoint.com/linux-regular-expression>
  - <https://opensourceforu.com/2012/06/beginners-guide-gnu-grep-basics/>



# Commandes de base

## Exemple: grep

Ex1: Show the lines in agenda.txt that contain "jp".

Ex2: Show the lines in agenda.txt that start with ":" contain a sequence of 4 numbers, and end with " :".

```
paris1@paris1-VirtualBox:~/ionescu$ cat agenda.txt
Piccolo-Daimao : 756 : guerrier.nakuseijin@namek.nk
Goku-Son : 8001 : goku@dbz.jp
David-Beserra : 666 : david@beserra.edu.br
Jhonny-LeBavard : 070 : jhonny.bavard@crazyman.fr
Pepito-ElBueno : 099 : pepito.es.bueno@hotmail.es
Madh-Lheine : 1992 : mdl@stu.de
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
paris1@paris1-VirtualBox:~/ionescu$ grep -e jp agenda.txt
Goku-Son : 8001 : goku@dbz.jp
Pikachu-Menouche : 865 :pikapi@pikaaaachu.jp
paris1@paris1-VirtualBox:~/ionescu$ grep " : .... : " agenda.txt
Goku-Son : 8001 : goku@dbz.jp
Madh-Lheine : 1992 : mdl@stu.de
```



# Basic Commands

| (pipe)

- Used to redirect the output of one command to another command.
  - Exemple: ps | grep firefox

The screenshot shows a terminal window with a dark background and light-colored text. The window title is "howtogeek@ubuntu: ~". The user has run several commands:

```
howtogeek@ubuntu:~$ kill firefox
bash: kill: firefox: arguments must be process or job IDs
howtogeek@ubuntu:~$ ps -A | grep firefox
  3684 ?        00:00:10 firefox
howtogeek@ubuntu:~$ kill 3684
howtogeek@ubuntu:~$
```

# + Commandes de base

## Exemple: pipe

```
paris1@paris1-VirtualBox:~$ ps -eo pid,user,start | grep "root" | grep "7:1:..." >> listeProcesse
paris1@paris1-VirtualBox:~$ cat listeProcesse
 647 root      07:10:00
 686 root      07:10:01
 688 root      07:10:01
 693 root      07:10:01
 697 root      07:10:01
 708 root      07:10:01
 713 root      07:10:01
 780 root      07:10:01
 796 root      07:10:01
 818 root      07:10:02
 825 root      07:10:02
 839 root      07:10:02
1026 root      07:10:03
1205 root      07:10:05
1291 root      07:10:07
1313 root      07:10:08
2364 root      07:11:08
2429 root      07:11:09
2778 root      07:15:01
```

In this example, we:

- (i) Used the command **ps** to display certain information about running processes (their IDs, users, and start times);
- (ii) Used the pipe command "**|**" to redirect the output of **ps** to the command **grep**;
- (iii) Used the command **grep** to select the processes created by the root user;
- (iv) (iv) Used the pipe command "**|**" to redirect the output of the **grep** command to another **grep** command;
- (v) Used the command **grep** to select the processes created by root that started after 7:10;
- (vi) Used the command "**>>**" to redirect the output of this last **grep** command to a file.



# Basic Commands: compression and decompression of files

- Traditionally, file compression on Linux is done in two steps:
  - Agglutination of files and directories into a single file, using the tar command
  - Compression of this single file (using commands such as gzip, bzip2, among others)



# Basic Commands: compression and decompression of files

- The "tar" command stands for "tape archiving", and with the appropriate options, it allows you to store multiple files and directories into a single file.
- Syntax:
  - To create a .tar file:
    - `tar -cvf file.tar dir1 dir2 dir3 ...`
  - To open a .tar file:
    - `tar -xvf file.tar`



# Basic Commands: compression et décompression de fichiers

- Options de la commande tar:
  - c: créer un nouveau fichier .tar
  - x: extrait les données du fichier .tar
  - t: répertorie où se trouve le contenu que fera partie du fichier .tar
  - f: indique que le prochain paramètre à utiliser est le nom du fichier .tar.
  - z: pour compresser/décompresser des fichiers gérés avec gzip



# Basic Commands

- A widely used procedure is the combined use of tar and gzip commands, using the "z" and "c" options of the tar command.
- Thus, to obtain a compressed file "file.tar.gz" with all the contents of the file directory, simply run the following command:
  - `tar -czvf file.tar.gz`.
- To extract, use:
  - `tar -zxvf file.tar.gz`.

# + Commandes de base

## Exemple: tar

```
paris1-VirtualBox: ~/ionescu
paris1@paris1-VirtualBox:~/ionescu$ ls
coucou.txt hello.txt hola.txt oi.txt saveHere
paris1@paris1-VirtualBox:~/ionescu$ tar -cvf groupe.tar *.txt
coucou.txt
hello.txt
hola.txt
oi.txt
paris1@paris1-VirtualBox:~/ionescu$ ls
coucou.txt groupe.tar hello.txt hola.txt oi.txt saveHere
paris1@paris1-VirtualBox:~/ionescu$ tar -xvf groupe.tar -C /home/paris1/ionescu/saveHere
coucou.txt
hello.txt
hola.txt
oi.txt
paris1@paris1-VirtualBox:~/ionescu$ ls saveHere/
coucou.txt hello.txt hola.txt oi.txt
paris1@paris1-VirtualBox:~/ionescu$ █
```



# Basic Commands

- To shut down the computer:
  - halt
  - shutdown -h now
- To reboot, use:
  - reboot
  - shutdown -r now
- What is the difference between shutdown and halt?
  - Google Time! xD
  - PS. This may come up on the exam ;)