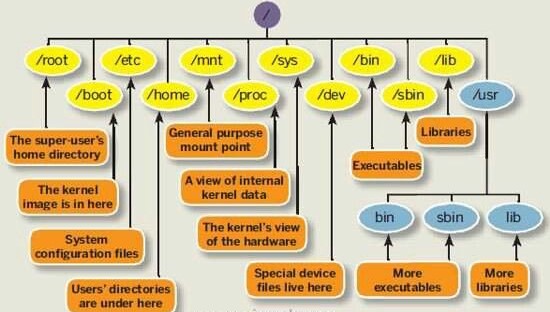
**LINUX PLUS NOTES:**

****

* **sudo apt update && sudo apt upgrade -y:** Update computer at one line
* **hostname -I:** Return IP address
* **sudo !!:** Last command will execute by sudo
* **exit():** To left python shell in mac
* **Tab:** Automatic completion of command, file name
* **Sequential commands (separate with”;”):** command1; command2; command3
* **ping** [**www.clarusway.com**](http://www.clarusway.com) :  Troubleshooting, testing, and diagnosing network connectivity issues.
* **ssh username@host\_ip**: To use computer of others

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* **man:** Displays the manual page from system's documentation

Previous and Next page press **‘p’ and ‘n’** and press **‘q’** to exit

* **--help**: Which gives a short explanation about how to use the command and a list of available options.
* **info ls:**
* **whatis:** Displays brief information about a command
* **apropos:** Search the manual page names and descriptions (search keyword in man page)

Ex: apropos ls🡪give information all command include ls

* **concatenate**: One of the basic uses of cat is to concatenate files into a bigger (or complete) file.

**Users:**

* **whoami:** indicate user name
* **who**: details about who is logged on the system
* **w:** command will inform you who is logged on and what they are doing
* **id:** show all your id
* **sudo su:** Root user in local home directory (enter password) **ctrl+d** to exit
* **sudo su -:** To switch any user from root user
* **su user1**: Go user1 shell (type **exit**: to exit from user1)
* **su -u user1:** Go to home directory of user1
* **sudo -s:** Start shell as root
* **sudo -u root:** Run as root / same **sudo**
* **useradd user1:** Create user without home directory, less secure

**useradd -m user1:** Forcing creation user1 with home directory

**useradd -m -d /home/house user1:** Settlingname of home directory

**useradd -m -d /home/house -c “user1 developer” user1**

* **adduser:** creates a /home/user directory automatically
* **userdel -r user1:** delete user with home directory
* **cat /etc/passwd:** Showsusers are inside home directorywith database
* **tail -3 /etc/passwd:** Shows last 3 users in home directory
* **usermod:** modify user’s properties
* **sudo usermod -l yasin user2**: change the name of the user2 to yasin
* **passwd user1:** Add/change the password to user
* **cat /etc/sudoers:** Allows users to run various commands as the root user without needing password. There is wheel group define in **sudoers**
* **cut -d: -f1 /etc/passwd:** list the users
* **grep ^PASS /etc/login.defs:** password settings

**Groups:**

* **groupadd:** create a new group
* **usermod -a -G group1 user1:** user1 become member of group1
* **groups**: display the list of groups
* **groupmod**: change the group name
* **groupdel**: delete group
* **gpasswd -a/-d “name”**: add or remove user from group
* **/etc/login.defs**: Futures of users/groups can change
* **sudo nano login.defs:** Change yes/no to create home directory of new user.

**Group Management:**

* **groups**
* shadow file >> kullanıcı paraloları şifreli bir şekilde /etc/shadow dosyasında bulunur. sadece root yetkisi ile okunabilir
* **EC2 instances futures can be change (color, name):**

export PS1="\[\033[1;31m\]\u@my-linux \[\033[1;37m\]\W: \$ "  
export PS1="\[\033[1;32m\]\u@my-ubuntu \[\033[1;37m\]\W: \$ "

**Find:**

* **find:** is an command for searching file(s) and folder(s) using filters such as size , access time , modification time.
* **find .** : current directory
* find /home -name fil\\*: find all the files that start with “fil” in the /home directory (-i name: case insensitive for files)

Environmental Variables:

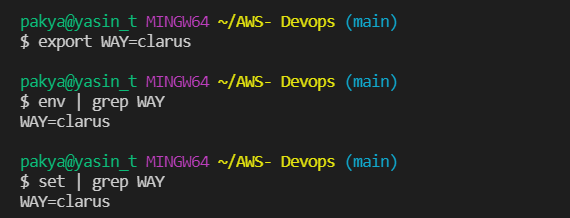
* echo $SHELL or which bash: Give us which shell using
* cat /etc/shells: Give us other shells
* bash: Open new shell
* shell variables: Temporally use/available, they will not execute any script even defined any variable.

When we call shell variable with echo $ sign must write beginning of variable. (Ex: echo $My\_var)

* environment variables: Permanently use/available (PATH, USER, HOME, EDITOR etc.)
* env: Display and manipulate the environment variables
* printenv “variable”: Value of a particular environment variable
* set: Set the value to the shell/environment variables
* unset: Unset the values of variable



* Manipulation of Shell variable. If you open new shell, you can’t see it.
* export: Sets environment variables



* Manipulation of environment variable. If you open new shell, you can see it.
* printenv PATH: Path variable use to find where executable files store
* Create **file.sh** inside of **new folder**. If we add folder to the end of path way, we can execute the **file.sh** script file.

**Quoting:**

Quoting is used to disable special treatment of certain characters.

The double quote protects everything enclosed except $,’,”,\

The single quote protects everthing

* **echo “$SHELL”:** /bin/bash
* **echo ‘$SHELL’:** $SHELL

**Grep:**

* grep 'word' filename: search any line that contains the word in filename on Linux.

**grep -i** Returns the results for case insensitive strings

**grep -n** Returns the matching strings along with their line number

**grep -v** Returns the result of lines not matching the search string

**grep -c** Returns the number of lines in which the results matched the search string