

//Aprendre linux//
 ///

```
1)pwd: Affiche le repertoire courant
2)ls: liste le contenu du repertoire courant

2-l)les parametres
**ls -l: liste le contenu du repertoire courant en plus les dossier caches
**ls -lt: triichage detaille plus la taille du dossier
**ls -lh: affichage detaille plus la taille du dossier d'une maniere lisible
**ls -lt ou -lt --fulltime: affiche detaille trier par date
**ls -l | grep jpg | wc -l :afficher le nombre d'image '.jpg' existant dans un dossier

3*)-color=(param): Prend deux param (none ou auto)--met des couleurs dans la console

4*)du -h(pour la lisibilité),sh(total des tailles): Affiche la taille occupée Les dossiers

5*)cat : Affiche le contenu d'un fichier tout d'un coup

5-l)les parametres
**cat -n: Affiche le contenu d'un fichier avec numero sur chaque ligne

6*)less: Affiche le contenu d'un fichier page par page

6-l)les raccourcis
**Espace: defiler vers la page suivante
**Enter: defiler vers la ligne suivante
**d:Affiche les onze ligne suivantes soit la moitié de la console
**q: Retourne à la ligne precedente
**u: retourne aux onze lignes precedente
**g: Pour quitter la lecture
**/soutit du mot à rechercher: permet de rechercher une experssion dans le contenu et 'n' pour passer à la prochaine occurence

7*)touch :Pour creer un fichier

8*)mkdir :Pour creer un dossier

8-l)parametres
** -p: permet de creer un dossier et ses sous dossier comme-->mkdir -p dossier/SousDossier

9*)cp fic_a_copier copie_fic : permet de copier un fichier

9-l)parametres
** -R: copier un dossier avec tout ses contenus
** -r: copier un dossier (meme fonctionnement que cp)

10*)mv: deplacer un fic ou dossier (meme fonctionnement que cp)
**NB: cette commande permet aussi renommer un fic comme: mv sekouze.txt sniper.txt

11*)rm: Supprimer un ou +eurs fichier,dossier sans avertissement

11-l)parametres:
** -i: Pour avertir l'utilisateur
** -f: Forcer la suppression
** -r: Pour supprimer un dossier
** -rf: supprimer tout les fichiers et dossiers du repertoire courant

12*)sudo commande: executer une commande en tant que root temporairement

13*)sudo su : devenir root definitivement

14*)adduser nom_utilisateur: Pour ajour un utilisateur

16*)passwd nom_utilisateur:changer le mot de pass

17*)deluser nom_utilisateur :supprimer un utilisateur

18*)deluser --remove-home nom_utilisateur: supprimer un utilisateur en plus tout son repertoire

19*)addgroup nom_group: ajouter un groupe

20*)usermod:permet de modifier un utilisateur

20-l)parametre:
** -l: pour renomer le nom
** -g:pour changer de groupe
****ex
usermod -i keita sekouze
usermod -g NouveauGroupe sekouze
**Remarque: il est aussi possible qu'un utilisateur appartienne à plusieurs groupe comme: usermod -G groupe1,groupe2,groupe3 sekouze

21*)shown nom_utilisateur_nom_fichier: permet de changer le propriétaire d'un fichier(executer en tant qu'admin)

22*)chmod droit_nom_fichier: permet de donner un certain droit à un fichier ense basant sur des chiffre(4-->sr,z-->sw,x-->l) ou des lettre(u,g,o avec les symbole "-">
>ajouter, "+" retirer et "-">affacter")

23*)nano : lancer l'editeur de texte console

25)nano nom_fichier.extension: lancer l'ancer l'editeur de texte console avec un fichier

26*)nano -n: Pour autoriser la saours dans l'editeur de texte

27*)nano -i: Pour activer l'indentation

28*)gcc nomFci.c -o nomExe: permet de compiler un fichier c

29*)apt-get install nom_appli: Pour installer un paquet

30*)apt-cache search nom_appli: Pour chercher un paquet si on ne connaît pas son vrai om

31*)apt-get update : pour mettre à jours le depot

32*)apt-get upgrade: Pour mettre tout les paques intallés tout d'un coup

33*)nanonon_commande: montrer le SYNOPsis(Les différentes facons d'utiliser une commande données) d'une commande

4*)apropos [domaine de recherche (SousVideo .....)]permet de trouver trouver toute les commandes liées à la commande de recherche

35*)commande -h ou --help: joue le meme role que man

36*)whatis commande: affiche le role d'une commande d'ou l'entete de 'man'

37*)locate nom_à_rechercher: Permet de rechercher un fic ou dossier peut importe son emplacement

38*)updatedb: executé en tant que admin,permet de repertorier lse dossier et fic recent de maniere à pouvoir les recherer avec la commande locate hhhh ;)

39*)find -name nom_à_rechercher: permet de rechercher un fic ou dossier dans l'emplacement parent

40*)find emplacement -name nom_à_rechercher: permet de rechercher un fic ou dossier dans un emplacement perci avec le nom exatme

41*)find emplacement -name "nom_à_rechercher": permet de rechercher les fic ou dossiers dans un emplacement perci ayant meme nom commençant par "nom_à_rechercher"
le contraîr c'est "nom_à_rechercher"---> se terminant bien sure

42*)find / -name nom_à_rechercher: Pour chercher sur tout le disque dur

43*)find -size +10M -rechercher tout les fic ou dossiers dont la taille depasse 10Mo hhhh on peut utiliser 'K' pour Ko ou encore 'G' pour Go

44*)find -name nom_à_rechercher -type d:rechercher uniquement les dossiers

45*)find -name nom_à_rechercher -type f:rechercher uniquement les fichiers

46*)grep mot_à_chercher fichier : cherche et affiche toutes les lignes dans lesquelles le mot_à_rechercher contient

46-l-*) Parametres:
** -i: pour ignorer les maj et min
** -n: Pour afficher les numero de ligne
** -v: pour effectuer une recherche sur le contraîr du mot c'est à dire trouver toutes les ligne sur lesquelles le mot_à_rechercher ne contient pas
** r ou grep: rechercher dans tous les fichiers et sous-dossiers
** -E ou egrep: Pour utiliser les expressions regulieres

47*)sort nom_fic: Permet de trer le contenu d'un fichier (odre croissant par default)

47-l-*) Parametres:
** -o: Permet d'enregistrer le tri dans les fichier
** -r: inverser le tri
** -R: tri aleatoire
** -n: tri des nombre

48*)wc nom_fic: oermet compter les nombre de (ligne,mot et octed) dans un fichier

48-l-*) Paramatres:
** -l: nombre de ligne
** -w: nombre de mot
** -c: nombre d'octet
** -m:nombre de caractere

49*)cut -c 2-5: couper les caracteres de 2 à 5 de chaque ligne

50*)cut -d , -f 1 notes.csv : couper un groupe de mots selon un delimiteur ben precis souvent dans les fichier (.csv).
Nb: le -d indique 'delimiteur' le -f indique la position a laquelle nous voulons couper dans notre cas le 1 et notre delimiteur est ','.
il est aussi possible de couper à +eurs positions en utilisant la virgule comme: -f 1,3

60*)commande>nom_fc.extension: permet d'enregistrer l'execution d'une commande dans un fichier

61*)commande>nom_fc.extension: Le double chevron >> sert lui aussi à rediriger le résultat dans un fichier, mais cette fois à la fin de ce fichier.

62*)commande 2>nom_fic.extension: pour rediriger le flux de sortie d'erreur dans un fichier

63*)commande 2>>nom_fic.extension: pour rediriger le flux de sortie d'erreur à la fin d'un fichier

64*)commande>nom_fc.extension 2>6l: pour effectuer la fusion des deux flux de sortie

65*)commande<< valeur_du_clavier: Lire depuis le clavier progressivement ex:
sort -n << FIN-->lire les nombres au clavier jusqu'à FIN et tri c'est ces nombres

66*)commande1 | commande2 | commande3 | commande.....: le | pipe permet de chaîner les commande en suivant le principe suivant:
le resultat d'une commande conduit à la commande d'entre de l'autre ex:
cut -d , -f 1 notes.csv | sort

67*)w : permet de voir l'etat du robot ,cet etat se caracterise comme suivant:
"date: la date actuelle du system
*up-->(uptime): represente la durée du fonctionnement du
*load average: est l'indice de charge qui comporte trois valeurs
**1ere valeur: la moyenne des charges apres 1min
**2eme valeur: la moyenne des charges apres 5min
**3eme valeur: la moyenne des charges apres 15min

68*)tload : Pour connaître l'evolution graphique du robot

69*)who : connaître les utilisateurs connectés, ce qui sont entrain de faire et depuis combien de temps

70*)ps :permet de lister les processus qui tournent sur le robot meme chose que la commande <top-->pour lister dynamiquement les processus du robot.
NB: on peut utiliser les parametres: (-ef)--->pour listr tout les processus et (-ejh)--->lister sous forme d'arbre

80*)ctr+c: permet d'arreter l'execution de la commande en couleurs

81*)kill PID: Pour tuer completement un processus le (PID)--->id du processus à tuer par fait (ps -u nom_utilisateur)
pour afficher tout les processus de l'utilisateur

82*)halt : pour arreter le robot cette commande est executer en tant quequ'admin

83*)reboot: de meme que la commande halt cette commande permet de redemarer le robot

84*)commande &: pour lancer un processus en arriere plan

85*)nohup commande: Pour lancer un processus en arriere plan mais d'un maniere dettaché c'est à dire que meme si l'on ferme la console le processus continuera.It la sortie est automatiquement rediriger vers le fic nohup.out

86*) ctrl+z : pour mettre en pause un processus

87*)bg : permet de passer un processus en arriere plan

88*)jobs :pour connaître les processus qui tournent en arriere plan

89*)fg : pour mettre un processus en premier plan s'il s'agit d'un seul sinon(fg %numero_du_processus)

90*)screen :pour lancer un emulateur de console par default il n'est pas installé (sudo apt-get install screen)

90-l-*) Quelques commandes les plus utiles de 'screen':
**ctrl+a+c : pour ouvrir une nouvelle fenetre
**ctrl+a+w : lister la liste des fenetre ouvertes
**ctrl+a+h : pour renommer la fenetre actuelle
**ctrl+a^n : pour passer à la fenetre suivante
**ctrl+a+p : pour passer à la fenetre precedente
**ctrl+a(0-9): pour à une fenetre en precisant son numero
**ctrl+a.w : pour choisir la fenetre ou l'on veut aller
**ctrl+a.k : pour fermer la fenetre actuelle
**ctrl+a.s : pour spliter l'emulateur de la console en deux
**ctrl+a.tab : pour passer à la fenetre suivante
**ctrl+a.X : pour fermer la fenetre splittée

91*)date symbole +%M%M: pour personnaliser l'affichage d'une date ex:
date +%H%> pour afficher l'heure de la date actuelle
date +%H%M%VS%> --> pour affichage la date sous le format hh:mm:ss

92*)sudo date MDDhhmmYY: Pour modifier la date du systeme, ces formats correspondent respectivement :
MM--> le mois
DD--> le jours
hh--> L'heure
mm--> la minute
YY--> annee(facultative)

93*)at temps_execution: Pour executer une commande à un moment bien precis, par la suite vous tapez les commandes à executer,en suite vous tapez ctrl+d pour arreter ex:
at 15:33 -->pour executer les commandes à 15h:13m
at 20:45 11/12/20-->pour executer les commandes le 12/11/2020 à 20h:45m

94*)atq: pour lister toutes les commandes qui seront executer à un temps precis avec leur numero

95*)atrm numero)commande: pour supprimer la commande qui sera executer à un moment donné

96*)sleep temps: pour endormire le programme à un temps donné ex: echo sekouze;sleep 10(par default en seconde);echo sniper.
"les differentes unités utilisées sont: m-->(minute),s-->(seconde);h-->(heure);d-->(jours)

97*)tar -cvf nom_dossier_archive.tar nom_dossier: pour archiver un dossier en 'tar'-->(forme compressée),Les differentes options signifines respectivement
c-->(creer un archive),v-->afficher les details,f-->(assembler l'archive dans un fichier)

98*)tar -cvf nom_dossier_archive.tar fichier1 fichier2 fi3 ... nom_dossier/ :pour archiver les fichier (meme chose que le precedent)

99*)tar -tf nom_dossier.tar : pour afficher les contenu d'une archive

100*)tar -rvf nom_dossier_archive.tar fichier_supplementaire : pour ajouter un fichier à une archive

101*)tar -xvf nom_dossier_archive.tar : pour extraire une archive

102*)gzip nom_dossier_archive.tar : permet de compresser une archive en 'gzip'

103*)gunzip nom_dossier_archive_compressée : pour decompressée une archive compressée

104*)bzip2 nom_dossier_archive.tar : meme fonctionnement que gzip (mais compression plus puissant et lente)

105*)bunzip2 nom_dossier_archive_compressée : pour decompresser

106*)tar -zxvf nom_dossier_archive.tar.gzip nom_dossier: pour archiver en meme temps compresser de meem que (-zxvf)-->pour decompresser

107*)tar -jcvf nom_dossier_archive.tar.bzip2 nom_dossier: meme priciple que le precedent

108*)zcat nom_fichier_compressé: equivalent à 'cat' mais pour lire le con d'un fichier compressé de meme que (zmore et zless)

109*)unzip nom_dossier.zip : pour deziper un dossier.zip. Attention 'unzip' n'est pas installé par default

110*)unzip -l nom_dossier.zip :pour lister le contenu

111*)tar -rvf nom_dossier.zip : pour zipper un dossier.Attention 'zip' n'est pas installer par default

112*) unrar e nom_dossier.rar : meme principe que 'unzip'(pas intallé par default) mais pour les .rar et il n'est pas possible creer des .rar avec le terminal de meme on peut utiliser 'unrar l nom_dossier.rar' pour lister le contenu

113*)sudo apt-get install openssl-server: permet d'installer le serveur ssh par default le serveur ssh est demarqué lors du demarage du robot

114*)sudo /etc/init.d/ssh start : pour demarner son serveur ssh

115*)sudo /etc/init.d/ssh stop : de meme que le precedent cette commande permet d'arreter le serveur

116*)sudo /etc/init.d/ssh reload : pour recharger toutes les modifications qui seront effectuer au sein du serveur comme:(/etc/ssh/ssh_config-->fic de ssh_config)

117*)ssh login@ip : pour se connecter sur une machine à travers une autre equiée de linux
(il faut remplacer login par votre net et ip par l'adresse ip de votre robot).

118*)ssh login@localhost : pour de connecter depuis sa machine(une sorte d'emulateur)

119*)logout ou ctrl+d : pour quitter

120*)ssh-keygen -t rsa : pour generer une paire de clé à fin de se connecter,le rsa est utiliser pour l'algorithme (assynchrone)
on peut aussi utiliser 'dsa'

121*)ssh-copy-id -i id_rsa.pub login@ip : pour envoyer la clé public se trouvant dans le dossier cachée (.ssh) au serveur . Par la suite vous pouvez vous connecter comme vous le faite au paravant on vous demandera votre passphrase au lieu du mot passe

122*)ssh-copy-id -i id_rsa.pub *p 14521 login@ip : pour effectuer l'envoi à travers un autre port,

123*)ssh-add : pourr.pouvoir se connecter comme <<agent ssh>> par la suite vous serez demander du passphrasees

124*)wget adresse_du_fic: pour telecharger un fichier

126*)wget -c adresse_du_fic: pour rprendre un telechargement arreté

126*)scp fichier_origine copie_destination : pour copier un dossier de son robot à un autre à travers un reseau.il s'utiliser sur ssh
ex:scp image.png mateo2l@85.123.10.201:/home/mateo2l/images/

126*)scp login@ip:image.png copie_image_sur_mon_pc.png: pour copier un fichier d'un autre robot vers le sien à travers un reseau

127*)scp -P numero_port login@ip:fichier destination: meme chose que le precedent mais sur un port different

127*)ftp ftp.nom_de_domaine :pour se connecter à un serveur à fin de transfere des fichiers
NB: pour taper les commandes de puis chez vous il faudra les preceder avec le symbole '$'

127*)put nom_fic: permet de transfere des fichier à travers le protocol ftp

127*)get nom_fic: permet de telecharger de fichier à travers le protocol ftp

127*)sftp login@ip : pour se connecter à travers le protocol securisé

127*)sftp -oPort=27401 login@serveur : pour se connecter à travers le protocol securisé sur un autre port

127*)ssh nom_domaine ou ip : permet d'afficher a partir du nom de domaine son ip et inversement

127*)sudo nano /etc/hosts :pour voir les adresses ip personnalisée sur le robot. On peut aussi au d'autre ip et leurs pour forcer le lien

128*)whois : cette tres importante permet d'afficher toutes les informations concernant un nom de nom_domaine pas install par default

129*)ifconfig: pour gerer la connexion reseau d'un robot auxquels on est connecté

**details
*eth0-->correspond au reseau par cable (RJ45)
*lo-->boucle locale
*wlan0--> reseau wifi

180*)ifconfig interface etat: pour activer/desactiver un reseau ce pendant (interface-->correspond(eth ou lo ou wlan)) et (etat-->(up ou down))

181*)netstat -parametre: pour connaître la statistique de reseau.il existe plusieurs parametres ce pendant les plus impotants sont:
* -i :affiche les infos pour chaque interface sous forme de tableau
* -uta : liste toutes les connexios ouvertes

182*)iptables -L :pour afficher toutes les regles qui se passent sur le trafic actuel

183*)iptables -F : pour reinitialiser tout les trafics .Attention ne faite que si vous en avez vramient besoin

184*) ./configure : se trouvant dans le dossier source d'un logiciel permet de compiler à fin de savoir si toutes les dependance sont prete

185*)sudo make install : pour install un logiciel compilé depuis le source

186*)apt-get install vim : pour installer l'editeur de texte pour programmeur

187*)
/////////////////////////////////////////////////////////////////vim l'editeur de texte pour les developpeurs/////////////////////////////////

1*)les touches (hjkL)<----->(les directions):permettent de deplacer la curseur

2*)q : pour quitter

3*)esc : pour revenir en mode normal

4*)i : pour insérer du texte

5*)A: pour Ajouter du texte une sorte d'insertion

6*)wq : pour enregistrer et quiter.

7*)x : pour effacer une lettre

8*)wd : pour effacer un mot

9*)ds : pour effacer un ligne

nb: on peut resumer le principe d'effacement dans vim comme suit:
d-----> l'operation d'effacement
w -> jusqu'au début du prochain mot, en EXCLUANT son premier caractère.
e -> jusqu'à la fin du mot courant, en EXCLUANT son dernier caractère.
$ -> jusqu'à la fin de la ligne, en INCLUANT son dernier caractère.

10*)numero+operation : pour quantifier les operations

ex: 2dd-->effacer deux ligne à la fois
d2w-->pour effacer deux mots à la fois.

11*)u : pour annuler les modifications

12*)U :pour annuler toutes les modifications

13*)p : pour coller ce qui vient d'etre effacer

14*)r+caractere: permet de remplacer une lettre se trouvant sous le curseur par 'caractere'.

15*)ce : pour changer le mot de la lettre courante jusqu'à la fin du mot.

16*)ctrl + g : pour afficher :l'etat actuel du curseur. Et numero+G-->pour se deplacer vers la ligne souhaitée

17*)gg: pour se deplacer vers la fin du fichier.

18*) /mot_à_chercher: pour rechercher un mot

19*) ?phrase_à_rechercher : pour rechercher une phrase.

20*)% : pour trouver < ou [ ou { correspondant.

21*) Tapez :s/ancien/nouveau/g pour remplacer 'ancien' par 'nouveau'

22*) :l+commande_externe(comme :ls,cd,pwd.....): pour executer une commande commande_interne

23*) :w nom_fic: pour enregistrer le contenu courant dans un fichier

24*)v+v+deplacement + w FICHER : pour enregistrer une portion du contenu

25*) o :pour ouvrir une nouvelle au desous du curseur en mode insertion et 'O'--> pour faire la meme chose au dessus

26*) e :pour se deplacer de mot en mot tout comme 'w' aussi

27*) a :pour insérer du text apres la position du curseur

28*) R :pour remplacer caractere par caractere

29*) v: mettre en mode Visuel(selection)

30*)Jy : pour copier du texte

31*) p : pour coller du texte

32*)set ic : pour regler l'option ignore casse à fin d'ignorer les casse au niveau des recherches

33*):sp : pour diviser l'ecran de vim en deux verticalement

34*):vsp :pour diviser l'ecran de vim en deux horizontalement

35*):set syntax=ON :pour activer l'option de la coloration syntaxique

36*) : set number :pour activer l'option numerotation des lignes

37*):set mouse=a : pour activer l'option de la souris

///////////////////////////////////////////////////////////////// fin de vim ///////////////////////////////////

188*)
///////////////////////////////// programmation shell ///////////////////////////////////

1*)sudo apt-get install nom_du_shell : pour installer un shell .nb: tout les shell se trouvent dans /bin/nom_du_shell

2*)chsh nom_du_shell: pour changer de shell ex: chsh /bin/ksh

3*)#!/bin/nom_du_shell : pour charger le type de shell utiliser pour un script dans 'vim'

4*)bash -x nom_du_script: pour lancer le script en mode debouage

5*)variable = valeur : pour declarer une variable et l'affecter une valeur

6*)echo message : pour afficher un message à l'ecran

7*)echo $variable: pour afficher une variable

8*)principe des quotes:
il en existe trois à savoir : simple(''),double("") et accent grave(``)
** ---->la variable n'est pas analysé meme $ est afficher dans le message
** ---->la variable est analysé et son contenu est affiché
** ---->la demande à bash d'executer le contenu de la variable

9*) read nom_de_la_variable : permet de faire une sasio dans le terminal ,on peut eventuellement lire +eurs variables à la fois comme:
read vl v2 v3 .....

9-l*)les Parametres
**read -p 'message' <----->(prompt): pour afficher un message avant la saisie
**read -n : pour limiter le nombre de caractere à saisir
**read -t <---->(timeout): pour limiter le temps de saisie
**read -s : ne pas afficher la saisie

10*) let : pour manipuler des operations comme ce-ci:
let "a=b"
let "a=9"
let "c=a+b"
echo "C=c"

****les operateurs
+->somme
-->difference
*-> produit
/->puissance
/-> quotient
%-> modulo

NB: on peut aussi utiliser des operation de contraction comme : let "a=b"

11*)env : pour afficher les variables d'environnement

12*)variable=('v1' 'v2' 'v3' 'v.....'): pour declarer un tableau

13*)echo {variable[]}] pour afficher tout les contenu du tableau

14*)Branchement conditionnel

14-l*)le si:
if [ test ]
then
echo "C'est vrai"
fi

**cette instruction permet des teste une variable à trvers la condition 'if'
nb: le 'fi' indique la fin de la condition et then celui du debut, il est tres important de savoir que si le 'if' est sur se trouvent sur la meme ligne il ne faut surtout pas oublier de mettre ';' à la fin de 'if'

14-2*)le sinon:
if [ test ]
then
echo "C'est vrai"
else
echo "C'est faux"
fi

14-3)le sinon-si:
if [ test ]
then
echo "Le premier test a été vérifié"
elif [ autre_test ]
then
echo "Le second test a été vérifié"
elif [ encore_autre_test ]
then
echo "Le troisième test a été vérifié"
else
echo "Aucun des tests précédents n'a été vérifié"
fi

15*) Les differents types de teste:

15-l*)teste sur les chaines:
** = :verifie si deux chaines sont identiques
** !=:verifie si deux chaines sont differentes
** -z :si une chaine est vide
** -n :si une chaine est non vide

15-2*)teste sur les nombres:
** -eq(equal): pour verifier l'egalite entre deux nombres
** -ne(unequal): inegalite
** -lt(lowerthan): inferieur ou egale
** -gt(greatherthan): superieur
** -ge (greaterequal): superieur ou egale

15-3)teste sur les fichiers:
** -e : si un fichier existe
** -d : si le fichier est repertoire
** -f : si le fichier est un fichier
** -L : si le fichier est un raccourci
** -r : si le fichier est lisible
** -w :le fichier est modifiable
** -x :si le fichier est executable
** -nt(newerthan) : si le fichier1 est plus recent que fichier2
** -ot(olderthan) : si le fichier1 est plus vieux que fichier2

16*)&& et || :pour le regroupement de +eurs conditions

17*) : pour insérer un teste

18*)les conditions par case:
"case : permet de tester les valeurs d'une meme variable cas par cas c'est à peuu pres comme le switch comme dans les autres langages

ex:
case $l in
"Bruno")
echo "Salut Bruno !"
;;
"Michel")
echo "Bien le bonjour Michel"
;;
"Jean")
echo "Hé Jean, ça va ?"
;;
*)
echo "J'te connais pas, ouste !"
;;
esac

ex2:
case $l in
"Chien" | "Chat" | "Souris")
echo "C'est un mammifere"
;;
"oiseau")
echo "C'est un oiseau"
;;
"Moineau" | "Pigeon")
echo "C'est un oiseau"
;;
*)
echo "Je ne sais pas ce que c'est"
;;
esac

19*)les boucles

19-l-*)while(tant que):
+++++synthaxe
while [ test ]
do
echo "Action en boucle"
done
nb: presque meme principe que les conditions.L'experssion 'until' est l'inverse du while

19-2)la boucle for:
for variable in 'valeu1' 'valeu2' 'valeu3'
do
echo "La variable vaut 'variable'"
done

19-3) le for classique:
for i in `seq 1 10`
do
echo $i
done
---Resultat: affiche tout les nombres entre 1 et 10

20*)Les fonction:
**il existe deux manieres de creer une fonction
**1)
maFonction ()
{
bloc d'instructions
}

**2)
#appel de ma fonction
maFonction
{
}

#appel de la fonction
maFonction
```