

|  |
| --- |
| HIGH TECH COMPASS : 1 RUE  DU CENTRE 93160 NOISY LE GRAND  E-mail: administration@htcompasse.fr – Web: [www.htcompass](http://www.htcompass).fr |

|  |
| --- |
| Shell correction-04 |
| Présenté par : Prénom Nom |

**Exercice**

**00 - Tester les différentes variables spéciales suivantes :**

$?

$#

$0

$n (n: valeur numérique, sauf zéro)

$!

**01 - Faire une fonction qui permet de calculer la moyenne arithmétique des cinq chiffres donnés en paramètre**

if test -z $1 || test -z $2 || test -z $3 || test -z $4 || test -z $5 ;then

echo "paramêtre manquant"; exit 101

else

echo $[($1+$2+$3+$4+$5)/5]

fi

**02 - Faire une fonction qui compare deux chaines de caractères et qui retourne « chaine1 » est plus grande que « chaine2 »**

#!/bin/bash

(( nombre1 = ${#1} ))

#puis

(( nombre2 = ${#2} ))

#puis

if (( "$nombre1" > "$nombre2" )); then

echo plus grande

elif (( "$nombre1" > "$nombre2" )); echo plus petit then

else echo égale

fi

**03 - Faire un programme qui permet de conjuguer les verbes du premier groupe (ER). Le verbe à conjuguer sera passé en paramètre. Seuls les verbes en ER devront être accepté, afficher un message d’erreur sinon.**

#!/bin/bash

terminaison="er"

echo -n "Entrez un verbe du 1er groupe: "

read var1

if [[ $var1 != \*$terminaison ]];

then echo "Ce n'est pas un verbe du 1er groupe!";

else

long=${#var1} ;

var1="${var1:0:long-2}";

echo "je " "$var1""e" ;

echo "tu " "$var1""es" ;

echo "il " "$var1""e" ;

echo "nous" "$var1""(e)ons" ;

echo "vous" "$var1""ez" ;

echo "ils " "$var1""ent";

fi

**04 – Faire une recherche d’un fichier en paramètre et affiche toute l’arborescence si présent.**

Locate fichier.sh

**05 - Faire une fonction qui convertit une valeur binaire (8 digits) et retourne sa valeur décimale.**

#!/bin/bash

stringZ=$1

n=1;

for i in 7 6 5 4 3 2 1 0;

do

var=${stringZ:i:1}

dec=$[( $dec + ( $var \* $n ))]

n=$[( $n \* 2 )]

done

echo "Valeur décimale: $dec"

**06 - Faire une fonction qui convertit une valeur décimale et retourne sa valeur binaire (8 digits).**

echo -n "Entrez un nombre entre O et 254:"

read n

if (( n > 255 )); then echo "valeur incorrect"; exit 101

fi

for e in 0 1 2 3 4 5 6 7

do

r=$[($n % 2)]

n=$[($n - r)/2]

hex="$r$hex"

done

echo "valeur binaire: $hex"

**07 – Faire un script qui génère un courrier de publipostage (création de plusieurs fichier texte) à partir d’une liste de contact (civilite; nom;prenom;adresse;codePostale;ville), contenu dans un fichier « contact.csv », données en paramètre.**

Exemple Template :

|  |
| --- |
| <civilite> <nom> <prenom>  <adresse>  <cp> <ville>  Madame, monsieur,    C'est avec grand plaisir que nous acceptons votre candidature  au sein de HTC FORMATION.    Cordialement,    L'équipe. |

while IFS=';' read idx civilite nom prenom adresse cp ville;

do

cp template.txt $nom"".txt";

sed -i -e "s/<civilite>/$civilite/g" $nom".txt";

sed -i -e "s/<nom>/$nom/g" $nom".txt";

sed -i -e "s/<prenom>/$prenom/g" $nom".txt";

sed -i -e "s/<adresse>/$adress/g" $nom".txt";

sed -i -e "s/<cp>/$cp/g" $nom".txt";

sed -i -e "s/<ville>/$ville/g" $nom".txt";

done < contact.csv

**08 - Faire un script qui se connecte à PostgreSQL exécute une requête et retourne le résultat**

Dans un script: plsql.sh

#psql.exe" -h %server% -U %username% -d %database% -p %port%

#server :192.168.1.142

#username :postgres

#database : db\_groupe1

#port : 5432

#password : admin@123

PGPASSWORD=admin@123 psql -h 192.168.1.142 -U postgres -d db\_groupe1 -p 5432 <<$

SELECT \*

FROM produit

EOF

Remarque : Script batch

SET PGPASSWORD=admin@123

psql -h 192.168.1.249 -U postgres -d db\_groupe1 -p 5432 -c "select \* from bureau;"

**Exercices supplémentaires non corrigés.**

09 Faire une fonction qui retourne le dernier jour du mois passé en paramètre (format : lundi 31 janvier)

10 faire une fonction qui calcule le nombre de jours ouvrés entre deux dates passées en paramètre