**453---SYSTEME D'EXPLOITATION - e-learning - Séquence 13**

**Programmation de scripts – 2**

**Légende :**

Explications sur fond blanc

Exercice à accomplir sur fond rouge clair

Une ou plusieurs solutions possibles pour l’exécution de commande sur fond vert clair

Une solution possible pour la rédaction du script sur fond jaune clair

Une variable d’un shell est une zone dans l’espace mémoire alloué par le système au processus shell. Elle est identifiée par un nom. Une valeur lui est affectée (stockée dans la zone). Tout processus, donc en particulier un shell, possède un espace mémoire dit d’environnement, contenant des variables d’environnement. Cet espace est hérité de processus shell père en processus shell fils.

**Quelques commandes :**

WHO(1)

NOM

who - Montrer qui est connecté

SYNOPSIS

who [OPTION] ... [ FICHIER | PARAM1 PARAM2 ]

DESCRIPTION

Afficher les informations sur les utilisateurs actuellement connectés.

WC(1)

NOM

wc - Afficher le nombre de lignes, de mots et d'octets d'un fichier

SYNOPSIS

wc [OPTION] ... [FICHIER] ...

wc [OPTION] ... --files0-from=FICHIER

DESCRIPTION

Afficher le décompte de lignes, de mots et d'octets pour chaque FICHIER

et le nombre total de lignes si plus d'un FICHIER est indiqué.

L'entrée standard est lue quand FICHIER est omis ou quand FICHIER vaut « - ».

Un mot est une séquence non vide de caractères délimités par des espaces.

Les options permettent de choisir quels décomptes sont affichés.

Ils le sont toujours dans l'ordre suivant : lignes, mots, caractères,

octets, longueur maximale des lignes.

La commande « set » permet d’afficher l’ensemble des variables du shell.

La commande « unset » permet de détruire une variable ou une fonction.

SET

NAME

set - manipulate shell variables and functions

SYNOPSIS

set [--abefhkmnptuvxBCHP] [-o option-name] [argument ...]

DESCRIPTION

If no options or arguments are supplied, set displays the names and values of all shell

variables and functions, sorted according to the current locale, in a format that can

be reused as input. When options are supplied, they set or unset shell attributes.

UNSET

NAME

unset - unset values and attributes of variables and functions

SYNOPSIS

unset [-fv] name ...

DESCRIPTION

Each variable or function specified by name shall be unset.

If -v is specified, name refers to a variable name and the shell shall unset it and remove it

from the environment. Read-only variables cannot be unset.

If -f is specified, name refers to a function and the shell shall unset the function

definition.

If neither -f nor -v is specified, name refers to a variable; if a variable by that name

does not exist, it is unspecified whether a function by that name, if any, shall be unset.

La commande « env » permet d’afficher les variables d’environnement du shell.

La commande « export » permet de mettre une variable dans l’espace d’environnement.

ENV(1)

NOM

env - Exécuter un programme dans un environnement modifié

SYNOPSIS

env [OPTION]... [-] [NOM=VALEUR]... [COMMANDE] [PARAM]...

DESCRIPTION

Initialiser chaque VARIABLE à la VALEUR dans l'environnement

et exécuter la COMMANDE.

Si aucune COMMANDE n'est fournie, afficher les variables d'environnement.

EXPORT

NAME

export - Set export attribute for shell variables

SYNOPSIS

export [-fn] [name[=value] ...] or export -p

DESCRIPTION

Set export attribute for shell variables.

Marks each NAME for automatic export to the environment of subsequently

executed commands. If VALUE is supplied, assign VALUE before exporting.

La commande « read » permet de créer une variable et lui affecter une valeur lue à partir de l’entrée standard (c’est-à-dire le clavier).

READ(2)

NAME

read - read from a file descriptor

Les trois premiers « file descriptors » (variable système fd), dans le système d’exploitation, [fd = 0, 1, 2] concernent les trois fichiers spéciaux de périphérique caractère suivants :

entrée standard (STDIN) [fd = 0] [lecture d’information à partir du clavier]

sortie standard (STDOUT) [fd = 1] [écriture d’information sur le terminal]

sortie d’erreur standard (STDERR) [fd = 2] [écriture de message d’erreur sur le terminal]

**LABORATOIRE – Programmation de scripts – 2**

**Dans la distribution Debian GNU/Linux :**

Dans la suite, le nom de l’utilisateur ordinaire sera représenté par le mot « user ».

Connectez-vous dans un terminal

« | » symbolise un « pipe », c’est-à-dire un « tube de communication » entre deux processus.

Avec echo, who et wc, affichez : « Nombre de connexions : … »

echo -n "Nombre de connexions : " ; who | wc -l

echo -e "Nombre de connexions : \c" ; who | wc -l

Créez une variable nommée : « A » et affectez-lui comme valeur la chaîne de caractères : « AVIS A LA POPULATION »

A="AVIS A LA POPULATION"

Affichez la valeur de la variable A

echo $A

Essayez d’afficher une page de manuel concernant la commande set

man set

Lancez un navigateur et, dans un moteur de recherche sur Internet, recherchez une page de manuel concernant la commande set

Essayez d’afficher une page de manuel concernant la commande unset

man unset

Lancez un navigateur et, dans un moteur de recherche sur Internet, recherchez une page de manuel concernant la commande unset

Affichez les valeurs de toutes les variables du shell

set

Affichez les valeurs de toutes les variables du shell en filtrant

set | more

Affichez les valeurs des 10 premières variables du shell

set | head

Recherchez la valeur de la variable A

set | grep A=

Détruisez la variable A

unset A

Visualisez l’effet de la suppression en recherchant à nouveau la valeur de la variable A

set | grep A=

Créez une variable numérique nommée : « varnum » et affectez-lui la valeur 135

varnum=135

Recherchez la valeur de la variable varnum

set | grep varnum=

Affichez la valeur de la variable varnum

echo $varnum

printf "%d" $varnum

Avec printf, affichez la valeur de la variable varnum, cadrée à droite sur 10 positions, avec passage à la ligne consécutif

printf "%10d\n" $varnum

Avec printf, affichez la valeur de la variable varnum, cadrée à gauche sur 10 positions, avec passage à la ligne consécutif

printf "%-10d\n" $varnum

Affichez le manuel de la commande env

man env

Essayez d’afficher une page de manuel concernant la commande export

man export

Lancez un navigateur et, dans un moteur de recherche sur Internet, recherchez une page de manuel concernant la commande export

Affichez les valeurs de toutes les variables d’environnement du shell

env

Affichez les valeurs de toutes les variables d’environnement du shell en filtrant

env | more

Recherchez la valeur de la variable d’environnement LANG

env | grep LANG=

Affichez le manuel de la commande man et constatez quelle est la langue utilisée

man man

Modifiez la valeur de la variable d’environnement LANG en lui affectant la valeur « C » (la langue ISO, c’est-à-dire l’anglais)

LANG=C

Vérifiez la modification en recherchant la valeur de la variable d’environnement LANG

env | grep LANG=

Constatez l’effet en affichant le manuel de la commande man

man man

Restaurez la langue française de Belgique dans l’environnement

LANG=fr\_BE.UTF-8

Vérifiez la restauration en recherchant la valeur de la variable d’environnement LANG

env | grep LANG=

Constatez l’effet en affichant le manuel de la commande man

man man

Vérifiez qu’aucune variable nommée : « AAA » n’existe dans l’environnement

env | grep AAA=

Créez une nouvelle variable nommée : « AAA » et affectez-lui comme valeur la chaîne de caractères : « aaaaaaa »

AAA=aaaaaaa

Exportez la variable AAA dans l’environnement

export AAA

Recherchez la valeur de la nouvelle variable d’environnement AAA

env | grep AAA=

Affichez une page de manuel concernant l’appel système read ou la commande read

man 2 read

man read

Lancez un navigateur et, dans un moteur de recherche sur Internet, recherchez une page de manuel concernant la commande read

Dans ~, enregistrez sous le nom « Bonjour » un fichier de script interactif qui affiche : « Bonjour, quel est votre nom ? » suivi de la saisie du nom sur la même ligne du terminal, puis qui affiche : « Bienvenue, votre nom est : … » dans une autre ligne

gedit Bonjour

#!/bin/bash

# Un script interactif

echo

echo -n "Bonjour, quel est votre nom ? " # affiche la phrase sans saut de ligne

read nom # crée la variable nom et lui affecte

# une valeur lue sur l’entrée standard

echo

echo "Bienvenue, votre nom est : $nom" # affiche la phrase avec la valeur

echo

Exécutez un dump du fichier Bonjour sur le terminal

cat Bonjour

Lancez un shell bash qui exécute Bonjour

bash Bonjour

Déconnectez-vous

exit