



PROGRAMACIÓN DE SISTEMAS

SHELL BASH

Mg. Edith Giovanna Cano Mamani

INTRODUCCIÓN

Esta basado en el sistema operativo UNIX, creado en 1969 en los laboratorios Bell de la empresa AT&T

- Con Shell scripts podremos definir variables y asignarle valores o permitir a que un usuario inserte valores de forma interactiva
- Se dispone bucles y estructuras de control para realizar operaciones aritméticas o de comparación
- Se puede tener varios shells scripts ejecutándose al mismo tiempo.
- La finalidad es combinar secuencia de comandos de Linux para realizar tareas determinadas.

VARIABLES

- Create a variable
 - Variablename=value (no spaces, no \$)
 - read variablename (no \$)
- Access a variable's value
 - \$variablename
- Set a variable
 - Variablename=value (no spaces, no \$ before variablename)

EJEMPLOS:

```
#!/bin/sh
```

```
myvar="Hi there"
```

```
echo $myvar
```

```
echo "$myvar"
```

```
echo '$myvar'
```

```
echo \ $myvar
```

```
echo $0
```

```
echo "$0"
```

```
echo '$0'
```

```
echo \ $0
```

```
echo date
```

```
echo "date"
```

```
echo 'date'
```

```
echo \date
```

```
echo `date`
```

```
echo Enter some text
```

```
read myvar
```

```
echo $myvar
```

```
exit 0
```

PARÁMETROS POSICIONALES

Positional Parameter	What It References
<code>\$0</code>	References the name of the script
<code>\$#</code>	Holds the value of the number of positional parameters
<code>\$*</code>	Lists all of the positional parameters
<code>\$@</code>	Means the same as <code>\$*</code> , except when enclosed in double quotes
<code>"\$*"</code>	Expands to a single argument (e.g., <code>"\$1 \$2 \$3"</code>)
<code>"\$@"</code>	Expands to separate arguments (e.g., <code>"\$1" "\$2" "\$3"</code>)
<code>\$1 .. \${10}</code>	References individual positional parameters
<code>set</code>	Command to reset the script arguments

EJEMPLOS

```
#!/bin/bash
echo \$home is $HOME
echo \$PATH is $PATH
echo \$PS1 which is my prompt is $PS1
echo \$# is number of parms: $#
echo \$\$ is PID of this shell $$
echo I could create a file called /tmp/tmpfile_$$
echo \$0 is $0
echo $1 is $1
echo $2 is $2
echo $3 is $3
echo $4 is $4
echo "\$IFS is "
echo "START $IFS END"
echo '$@' is "$@"
echo '"$*" ' is $*
echo '$*' is "$*"
IFS=:
echo "\$IFS is "
echo "START $IFS END"
echo '$@' is $@
echo '"$@" ' is "$@"
echo '$*' is $*
echo '"$*' ' is "$*"
echo loop through '$@' no quotes
```

```
for i in $@
do
echo $i
done
echo loop through '"@" ' with quotes
for i in "$@"
do
echo $i
done

echo loop through '$*' no quotes
for i in $*
do
echo $i
done
echo loop through '"$*' ' with quotes
for i in "$*"
do
echo $i
done

unset IFS
set $(date)
echo "I reset parms to the date, so now the parms are "
for i in $*
do
echo $i
done
```

VARIABLES Y PROCESOS

Las variables se comparten solo con su propio proceso, a menos que se exporten

- `x=Hi` – define `x` en el proceso actual
- `sh` – Lanza un nuevo proceso
- `echo $x` – no se puede ver `x` del proceso principal
- `x=bye`
- `<ctrl d>` -- salir del nuevo proceso
- `echo $x` -- ver `x` en el proceso anterior que n
- `demoShare` – cannot see `x`
- `. demoShare` – run with dot space runs in current shell
- `export x` – exports the variable to make available to its children
- `demoShare` – now it can see `x`

wget

<http://home.adelphi.edu/~pe16132/csc271/note/scripts/demoShare>

EJERCICIOS

Read from stdin (screen)

Read until new line

Format

Significado

`read answer`

Lee una línea de stdin en la respuesta variable

`read first last`

Lee una línea desde stdin hasta el espacio en blanco, colocando la primera palabra en primer lugar y el resto de la línea en último

`read`

Lee una línea de stdin y la asigna a `REPLY`

`read -a`

`arrayname`

Lee una lista de palabras en una matriz llamada `arrayname`

`read -p prompt`

Imprime un aviso, espera la entrada y almacena la entrada en `REPLY`

`read -r line`

Permite que la entrada contenga una barra invertida.

<http://home.adelphi.edu/~pe16132/csc271/note/scripts/nosy>

TAREA_01

- Desarrollar los siguientes ejercicios:

https://drive.google.com/drive/folders/1l5MhR4y77H5bIVcxUVvO30k6FzL7dy_0?usp=sharing

TAREA_02

- Desarrollar los siguientes ejercicios:
 - https://tldp.org/LDP/intro-linux/html/sect_03_06.html

REPASO

- <https://www.andrew.cmu.edu/course/15-123-kesden/applications/ln/lecture2.html>