

Sistema Operativo Linux

Ejercicio 3. Comandos de Linux

1. Abra una pantalla de terminal, ya sea desde su interfaz de escritorio o pulsando las teclas **[Ctrl]+[Alt]+[T]**

2. Manejo de la pantalla de la terminal

a) El comando **echo** muestra en pantalla los caracteres que son entrados desde el teclado. Este comando también interpreta caracteres de control si el parámetro **-e** está presente. Imprimiremos el siguiente mensaje en dos líneas:

"El texto siguiente será impreso en dos líneas"

```
echo -e "El texto siguiente será\nimpreso en dos líneas"
```

b) El siguiente texto forma parte de un shell script:

```
echo -e "Primera línea\c"
```

```
echo "Segunda línea"
```

El carácter de control **\c** permite que el segundo texto se imprima a continuación del primero

c) Ejecución de un comando como parte de otro comando

"Mi grupo es <nombre> que se corresponde con el número <nº>"

```
echo -e "Mi grupo es \c"; id -g -n; echo -e "que se  
corresponde con el numero \c"; id -g
```

"Mi directorio actual es <directorio>"

```
echo -e "Mi directorio actual es \c"; pwd
```

"Listado de usuarios conectados:"

```
echo "Listado de usuarios conectados:" ; finger
```

"Hay <n> usuarios conectados en este momento"

```
echo "Hay `who | wc -l` usuarios conectados en este momento."
```

Visualizar en caracteres grande "Mi id es <id>"

```
banner "Mi id es `logname`"
```

Mostrar el nombre del directorio actual, sin la ruta de acceso

```
basename $PWD
```

Mostrar la ruta de acceso al directorio actual, sin el nombre del directorio

```
dirname $PWD
```

3. Manejo de directorios

a) Camino absoluto o ruta absoluta

El término camino absoluto o ruta absoluta es utilizado para señalar la ruta hacia un archivo o hacia un directorio partiendo de la raíz del sistema de archivos (*filesystem*) de Linux, que se conoce como la "raíz" o "**root**" y se denota por un *forward slash* `"/`. Una ruta absoluta siempre tendrá como inicio el `"/`

Por ejemplo:

```
/home/usuario/directorio1/directorio2/archivoA
```

b) Camino relativo o ruta relativa

El término camino relativo o ruta relativa es utilizado para señalar la ruta hacia un directorio o hacia un archivo partiendo de la posición actual del usuario en el sistema de archivos.

Considerando el ejemplo anterior, si el usuario está ubicado en este momento en el directorio1, y quiere señalar la ruta hacia el archivoA, usando ruta relativa escribiría::

```
directorio2/archivo
```

c) Cuando iniciamos una sesión de trabajo, el sistema nos sitúa en un directorio llamado de directorio de conexión. La ubicación de dicho directorio se establece en `/etc/passwd`

La variable de entorno que almacena la ubicación del directorio de conexión es `$HOME`

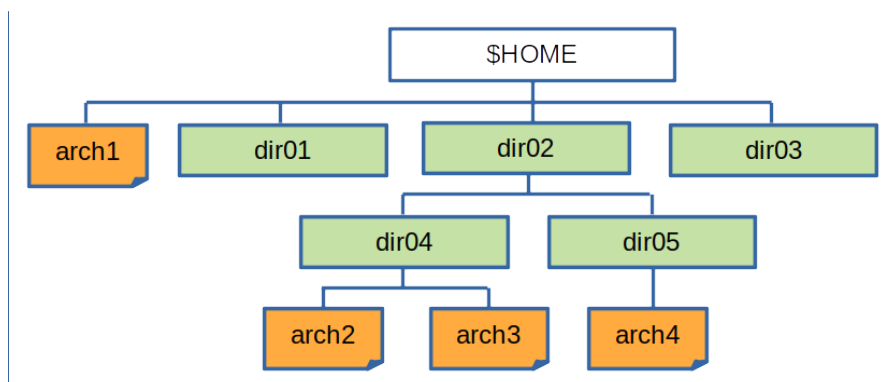
El comando `echo $HOME` mostrará la ruta hacia nuestro directorio de conexión

d) Listar el contenido del directorio de conexión

```
ls $HOME
```

e) El comando `ls $HOME` mostrará el listado de los directorios y archivos contenido en el directorio de conexión

f) Crear la siguiente estructura de directorios y archivos que se indican en el gráfico. "dir??" indica que se trata de un directorio, mientras que "arch?" indica que se trata de un archivo.



El comando **mkdir** se utiliza para crear directorios (llamadas también "carpetas"), mientras que el comando **touch** crea un archivo vacío. El comando **cd** permite movernos entre los directorios. A éste comando se le acompaña con una ruta absoluta o una ruta relativa, según sea el caso

```
cd $HOME          (puede utilizarse cd sin parámetros, que nos lleva al
                  directorio de conexión)
```

```
mkdir dir01 dir02 dir03
```

```
touch arch1
```

```
cd dir02
```

```
mkdir dir04 dir05
```

```
touch dir04/arch2 dir04/arch3 dir05/arch4
```

- g) desde el directorio **\$HOME**, muestre la estructura de directorios con inicio en **dir02**

```
cd $HOME
```

```
ls -R dir02
```

- h) Desde el directorio **dir04** muestre el listado del directorio **dir03** utilizando ruta absoluta como ruta relativa

```
ls ../../dir03
```

Ruta relativa

```
ls $HOME/dir03
```

Ruta absoluta

- i) Asumiendo que nos encontramos en nuestro directorio de conexión, queremos cambiarnos al directorio **dir05**

```
cd dir02/dir05
```

- j) Estando en **dir05**, muestre como podemos regresar a nuestro directorio de conexión

```
cd $HOME
```

```
cd ../../
```

```
cd
```

- k) Liste todos los directorios y archivos que empiecen con "g" que estén en el directorio de conexión

```
ls $HOME/d*
```

- l) El comando **ls** con parámetro **-l** muestra un listado largo de los archivos y directorios del directorio actual. Muestra el tamaño del archivo, fecha y hora de creación, el tipo de archivo (archivo, directorio, enlace), los atributos del propietario, grupo y el resto, nombre del propietario, nombre del grupo, etc.

```
ls -l
```

- m) El comando **ls** con el parámetro **-a** nos muestra todos (*all*) los archivos y directorios de nuestro directorio actual. Los directorios que normalmente no se muestran con **ls**, tienen un carácter "." al inicio del nombre lo que los convierte en archivos "ocultos"

```
ls -a
```

- n) La siguiente secuencia de comandos listan el contenido del directorio actual mostrando solo un espacio en blanco de varios espacios en blanco consecutivos

```
ls -l | tr -s " "
```

4. Manejo de archivos

- a) Mostrar el contenido de un archivos

```
cat arch1
```

- b) Mostrar el contenido de dos archivos que se encuentran en el mismo directorio con un único comando

```
cat arch1 arch2
```

- c) Muestre el contenido de arch1 cada 10 líneas, de dos maneras diferentes

```
cat arch1 | more -10
```

```
more -10 arch1
```

- d) Muestre las diez primeras líneas de **arch1**

```
head -10 arch1
```

- e) Muestre las diez últimas líneas de **arch1**

```
tail -10 arch1
```

5. Nos ubicamos en el dir03

```
cd $HOME/dir03
```

- a) Copiamos todos los archivos del directorio **/bin** que empiecen con **p** en el directorio actual

```
cp /bin/p* .
```

 El carácter "." indica que el destino es el directorio actual

- b) ¿Quién es el propietario y a que grupo pertenecen los archivos que empiezan con **p** en el directorio **/bin**?

```
ls -l /bin/p*
```

 Observamos que el propietario y el grupo es **root**

- c) ¿Quién es el propietario y a que grupo pertenecen los archivos que empiezan con **p** en el directorio **dir03**?

```
ls -l p*
```

 Observamos que el propietario y el grupo es el usuario

6. Borre los archivos que empiezan con "d"

```
rm d*
```

7. Enlaces o links

- a) Cree un enlace al archivo **arch1** llamándole **arch11**

```
ln arch1 arch11
```

b) Indique el comando que muestra los enlaces o *links* que tiene **arch1**

```
ls -l arch1
```

c) ¿Qué sucede si borramos arch1?

Al crear un enlace de **arch1** en **arch11**, prácticamente se ha creado un segundo archivo con el contenido del primero (a esto se conoce como "enlace duro"). Si borramos **arch1**, **arch11** permanece y con él, el contenido que tuvo **arch1** y que ahora está en **arch11**

8. Copie el contenido completo de dir01 en dir03

```
cp /$home/dir01 /$home/dir03
```

9. Muestre el número de líneas que tiene arch2

```
wc -l arch2
```

10. Muestre el número de líneas que tiene arch2 y arch3

```
wc -l arch2 arch3
```



Esta obra está bajo una [Licencia Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/)