Comandos varios

1.- Listar todos los directorios de **/etc** que empiecen por **a** y guarda el resultado en un fichero llamado **direca**.

**ls -d /etc/a\* > direca**

2.-Listar todos los ficheros de **/etc** que empiecen por **h** y tengan **5 letras**.

**ls /etc/[ht]???? > ficheros\_ht\_5letras**

3.- Listar ordenado alfabéticamente el directorio actual.

**ls | sort**

4.- Crea un directorio en **/home** llamado **examen**, con ruta relativa.

**mkdir /home/examen**

5.- Crea, en tu directorio de trabajo, el fichero vacío **letras.txt** y cópialo dentro del directorio examen con el nombre **letrass**.

**touch letras.txt**

**cp letras.txt ~/examen/letrass**

6.- Utilizando la orden **cal**, guarda el mes actual en un fichero llamado **mes** y convierte los números en letras.

**cal > mes**

**sed -i 's/1/Uno/g; s/2/Dos/g; s/3/Tres/g; s/4/Cuatro/g; s/5/Cinco/g; s/6/Seis/g; s/7/Siete/g; s/8/Ocho/g; s/9/Nueve/g; s/10/Diez/g; s/11/Once/g; s/12/Doce/g' mes**

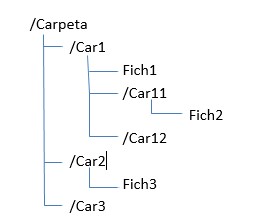
7.- Busca en el directorio del usuario actual todos los ficheros que empiecen por m

**find ~ -type f -name 'm\*'**

9.- Muestra un listado ordenado por tipo de fichero/directorio del directorio actual.

**ls -l --group-directories-first**

10.- Crea la siguiente estructura de directorios y ficheros vacíos en tu directorio de trabajo:



**mkdir -p Carpeta/Car1/Car11**

**touch Carpeta/Car1/Fich1**

**touch Carpeta/Car1/Car11/Fich2**

**touch Carpeta/Car2/Fich3**

**mkdir Carpeta/Car3**

Utilizando rutas relativas: **Suponemos que estamos en /home/alumno/Carpeta**

1. Copia el **Fich3** en el directorio **Car12** con el nombre **Fich4.**

**cp Fich3 Car2/Car12/Fich4**

1. Copia el **Fich2** en el directorio **Car12** con el nombre **Fich5**.

**cp Car1/Car11/Fich2 Car2/Car12/Fich5**

1. Copia los ficheros **Fich4** y **Fich5** en directorio **Car3**.

**cp Car2/Car12/Fich4 Car3/**

**cp Car2/Car12/Fich5 Car3/**

1. Copia el fichero **/etc/fstab** en el Directorio **Car3** con el nombre particiones.

**cp /etc/fstab Car3/particiones**

1. Asegúrate de que te encuentras en tu carpeta principal (pwd).

**Cd ..**

1. Crea 2 carpetas: **ejercicios1, ejercicios2**.

**mkdir ejercicios1 ejercicios2**

1. Accede a **ejercicios1**. Allí, crea un fichero de texto llamado "**file1.dat**" con la frase "**Hola mundo**". Asegúrate de guardar.

**cd ejercicios1**

**echo "Hola mundo" > file1.dat**

18.- Sin moverte del directorio **ejercicios1**, haz una copia del fichero "**file1.dat**" cambiándole el nombre por “**file1\_copia1.dat**”.

**cp file1.dat file1\_copia1.dat**

19.-Accede al directorio **ejercicios2**; asegúrate de existen dos ficheros con el comando "**ls**".

¿Qué ocurre si te equivocas y usas "sl “?

**ls –l**

**Si te equivocas saldrá un error o aparecería para descargar algún paquete con comandos, sale tambien el comando de ayuda para buscar**

20.- Muestra el contenido de cada fichero usando el comando "**cat**".

**cat file1.dat**

**cat file1\_copia1.dat**

21.-Sin moverte del directorio **ejercicios2**, mueve el archivo "**file1.dat**" hasta **ejercicios2/mover**, poniendo como nombre de destino "**file1\_movido.dat**”. Ten en cuenta que, posiblemente tendrás que crear antes la carpeta.

**mkdir -p mover**

**mv file1.dat mover/file1\_movido.dat**

22.- Haz una copia del archivo "**file1\_copia1.dat”** en la misma carpeta, llamada "**file1\_copia2.dat** "

**cp file1\_copia1.dat file1\_copia2.dat**

23.- Aún en **ejercicios2**, elimina de una sola vez los archivos "**file1\_copia1.dat**" y

"**file1\_copia2.dat**". Para ello usa comandos con patrones "\*". Así, solo te quedará la copia de seguridad y la carpeta **mover**. Si te equivocas y borras más de la cuenta, vuelves a empezar.

**rm file1\_copia\*.dat**

24.- Elimina la carpeta "**mover**" con todo su contenido. Al final, solo nos quedará la copia de seguridad.

**rm -r mover**

25.- Listar el contenido del directorio actual ordenado por tamaño de los archivos.

**ls -lS**

26.- Listar el contenido del directorio **/etc** completo invirtiendo el orden usual de mostrar el directorio.

**ls -lR /etc | tac**

27.-Muestra el espacio ocupado por los archivos y directorios del directorio actual.

**du -h**

28.- Indica la orden completa para realizar las siguientes acciones:

1. Crear un fichero llamando **fileTodo** cuyo contenido sea el listado completo del directorio raíz del sistema. Utiliza el símbolo de >

**ls -R / > fileTodo**

1. Visualizar el fichero **fileTodo** paginado, pero pudiendo volver hacia atrás.

**less fileTodo**

1. Muestra las 5 primeras líneas del fichero **fileTodo**.

**head -n 5 fileTodo**

1. Muestra las 5 últimas líneas del fichero **fileTodo**.

**tail -n 5 fileTodo**

1. Muestra las dos últimas líneas de las 6 primeras líneas del fichero **fileTodo**.

**head -n 6 fileTodo | tail -n 2**

1. Muestra las tres primeras líneas de las 6 últimas líneas del fichero **fileTodo**.

**tail -n 6 fileTodo | head -n 3**

1. Visualiza un listado completo en formato largo del directorio **/etc**

**ls -l /etc**

1. Visualizar solo la primera columna del listado anterior.

**ls -l /etc | awk '{print $1}'**

1. Ejecuta la siguiente orden: **ls –l /etc | cut –f1,4,5 -d” “** .¿Que aparece en pantalla?

Este comando muestra una lista con los permisos de los archivos y los nombres de usuario y grupo propietario

1. Crea un fichero llamado file4 con los siguientes datos:

***Pepe 02:30:44***

***Marcos 23:56:33***

***Pepe 10:33:01***

***Marta 05:47:44***

***Pepe 12:22:33***

***José 11:55:00***

**echo -e "Pepe 02:30:44\nMarcos 23:56:33\nPepe 10:33:01\nMarta 05:47:44\nPepe 12:22:33\nJosé 11:55:00" > file4**

* + Visualiza, por pantalla, solo la columna de los nombres

**awk '{print $1}' file4**

* + Ídem, pero con la columna de las horas.

**awk '{print $2}' file4**

* + Visualiza las tres primeras líneas.

**head -n 3 file4**

* + Visualiza las dos últimas líneas.

**tail -n 2 file4**

1. ¿Cuantas líneas tiene el listado del directorio **/dev**?

**ls -l /dev | wc -l**

1. ¿Cuantas líneas tiene el fichero **/etc/fstab**?

**wc -l /etc/fstab**

1. ¿Cuantas palabras hay en el fichero **/etc/group**?

**wc -w /etc/group**

1. Visualiza, por pantalla, las columnas 1,3,4 y 7 del fichero **/etc/passwd**.

**awk -F':' '{print $1, $3, $4, $7}' /etc/passwd**

1. Busca la palabra **root** en el fichero **/etc/passwd**

**grep "root" /etc/passwd**

1. Busca las líneas que comiencen por la palabra **root** en el fichero **/etc/passwd**

**grep "^root" /etc/passwd**

1. Crea el fichero **file5** con los siguientes datos:

***José 11:55:00***

***Marcos 23:56:33***

***Marta 05:47:44***

***Pepe 02:30:44***

***Pepe 10:33:01***

***Pepe 12:22:33***

**echo -e "José 11:55:00\nMarcos 23:56:33\nMarta 05:47:44\nPepe 02:30:44\nPepe 10:33:01\nPepe 12:22:33" > file5**

* + Busca el numero 4

**grep "4" file5**

* + Listar solo la hora.

**awk '{print $2}' file5**

* + Listar solo los minutos

**awk -F: '{print $2}' file5**

* + Listar solo los segundos

**awk -F: '{print $3}' file5**

* + Buscar todas las líneas que comiencen por m

**grep "^m" file5**

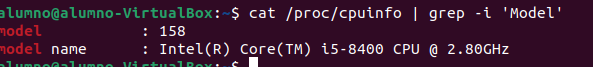
* + Mostrar todas las líneas numeradas que comiencen por p ^

**grep -n "^p" file5**

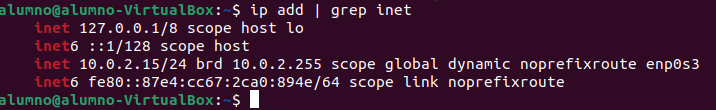
* + Buscar todas las líneas que finalicen en 3

**grep "3$" file5**

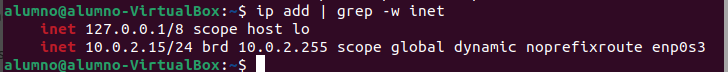
1. Indica que muestra la ejecución de las siguientes líneas:
   * **cat /proc/cpuinfo | grep –i ‘Model’**



* + **ip add | grep inet**



* + **ip add | grep –w inet**



1. Verifica que existe el usuario **pulse** en el fichero **/etc/passwd**

**grep "^pulse:" /etc/passwd**

1. Averigua que archivos tienen permisos rwxrwxrwx en el directorio **/etc/**

**find /etc/ -type f -exec ls -l {} + | grep "rwxrwxrwx"**

1. Buscar todos los archivos enlaces blandos o simbólicos del directorio **/dev**

**find /dev -type l**

1. Indica que realizar la siguiente orden: **cut –f1 –d” “ file4 | sort | uniq –ic**

cut -f1 -d" " file4: Corta el archivo file4 utilizando como delimitador el espacio (" "), seleccionando el primer campo de cada línea.

sort: Ordena alfabéticamente las líneas resultantes.

uniq -ic: Filtra líneas duplicadas y muestra el recuento de cada línea única

29.- Indica las ordenes ejecutadas en cada caso del comando **tar**:

1. Crea tres ficheros llamados **file1, file2 y file3**. Cada fichero debe tener un mínimo de 3 líneas de texto. Puedes utilizar el comando nano. Con dichos ficheros debemos crear un paquete llamado ficheros.tar.

**tar -cf ficheros.tar file1 file2 file3**

1. El paquete ficheros.tar es demasiado pesado para enviarlo por correo. Crea un paquete comprimido con el nombre ficheros.tar.gz o ficheros.tgz

**gzip ficheros.tar**

1. Antes de enviarlo por correo debemos comprobar su contenido. Visualiza el contenido del paquete comprimido que has generado, pero sin descomprimir ni desempaquetar.

**tar -ztvf ficheros.tar.gz**

1. El receptor del paquete lo ha descomprimido y desempaquetado.

**tar -xzf ficheros.tar.gz**

30.- Utilizando el comando **locate**. Si realizas algún cambio, deberás utilizar updatedb para actualizar los valores.

1. Busca el archivo passwd.

**locate passwd**

1. Busca el archivo makefile ignorando mayúsculas.

**locate -i makefile**

1. Muestra información de la base de datos de locate.

**locate --statistics**

1. Mostrar la versión de locate.

**locate --version**

31.- Utilizando el comando **whereis**

1. Averigua la ruta de acceso al comando passwd

**whereis passwd**

1. Indica en que páginas del manual se encuentra la información del comando passwd

**man -k passwd**

1. Indica las páginas del manual del comando mount

**man -k mount**

32.- Utilizando el comando **which**

1. Mostrar el directorio donde se encuentra el comando mount

**which mount**

1. Mostrar todas las coincidencias del comando which

which -a which

34.- Utilizando el comando **sort**

1. Muestra por pantalla un listado ordenado del contenido completo del directorio actual.

**ls | sort**

1. Muestra por pantalla un listado ordenado de modo inverso del contenido completo del directorio actual.

**ls | sort -r**

1. Muestra por pantalla un listado ordenado por el campo 3 del archivo /etc/group.

**sort -t':' -k3 /etc/group**

1. Crea un archivo llamado **sortfich** (hay un espacio de separación) con los siguientes datos:

Azul 24 Venus 1970 Ana

Rojo 35 Neptuno 1122 Javier

Blanco 73 Marte 1543 Manuel

Amarillo 135 Tierra 1234 Raúl Verde 17 Júpiter 1968 Sebastián

**echo -e "Azul 24 Venus 1970 Ana\nRojo 35 Neptuno 1122 Javier\nBlanco 73 Marte 1543 Manuel\nAmarillo 135 Tierra 1234 Raúl\nVerde 17 Júpiter 1968 Sebastián" > sortfich**

Ordena:

* 1. por colores el fichero.

**sort -k1 sortfich**

* 1. por el segundo campo del fichero.

**sort -k2n sortfich**

* 1. por planetas el fichero y guarda el resultado en un fichero llamado fichero.ord.

**sort -k3 sortfich > fichero.ord**

* 1. por colores de forma inversa y muestra por pantalla **solo** los colores.

**sort -rk1 sortfich | awk '{print $1}' | uniq**