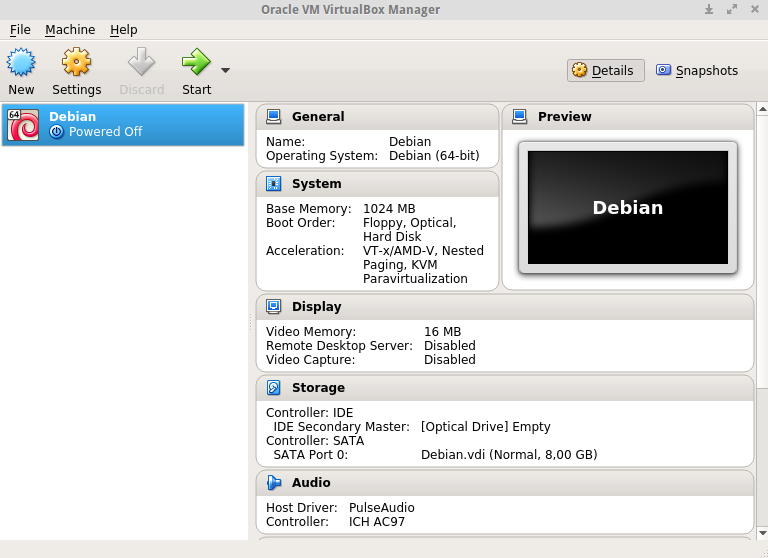
# Unidad 5. Primeros pasos con la línea de comandos. Comandos básicos y expresiones regulares

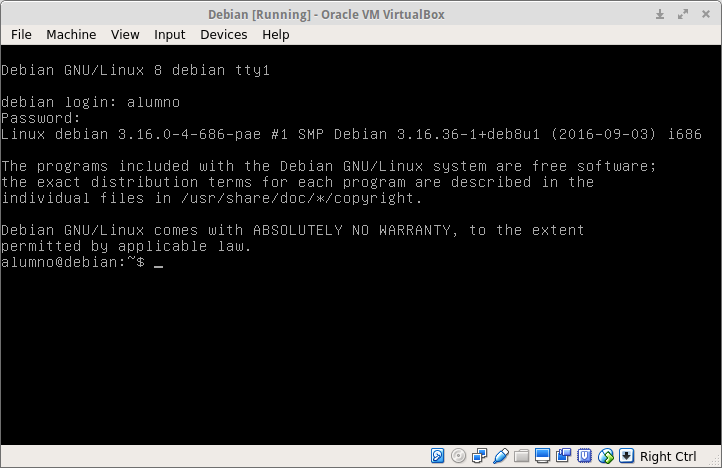
En esta unidad empezaremos a trabajar con la línea de comandos. Para ello utilizaremos la distribución Debian, instalada en una máquina virtual. Para iniciar la máquina virtual, primero abrimos el software VirtualBox:



Luego seleccionamos la máquina virtual “Debian” y hacemos click en “Start”, Esto iniciará la máquina virtual y nos pedirá las credenciales:

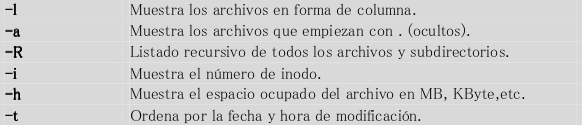
* Usuario: alumno
* Contraseña: alumno

Una vez ingresadas las credenciales, podemos empezar a utilizar la línea de comandos

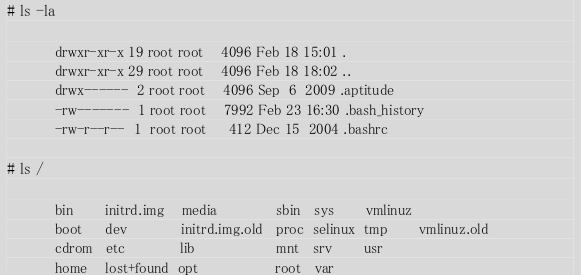


## Comandos básicos

Uno de los comandos más básicos que empezaremos a utilizar será el ***ls*** que nos permite poder ver los contenidos de los directorios.



Ejemplos:



Para movernos en distintos directorio/subdirectorios usamos el comando ***cd***.

Ejemplos:



Para ir a un directorio anterior:

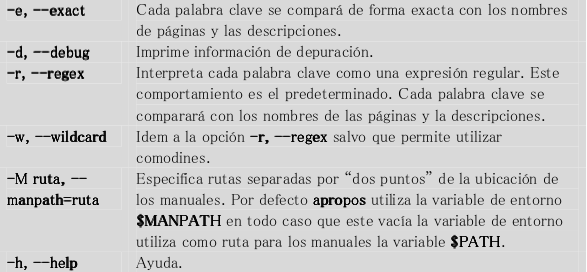


Para ir a nuestro directorio home, utilizamos lo siguiente:



## Obtener información de los comandos

Gracias a una serie de comandos podemos obtener información de los archivos de configuración, y de los comandos que ejecutamos, también tendremos otros comandos que nos permite realizar búsquedas de manuales. El comando que utilizaremos es ***apropos***. Cada manual contiene una pequeña descripción, este comando nos permite buscar dentro de esas descripciones. Las opciones que podemos usar con este comando son las siguientes:

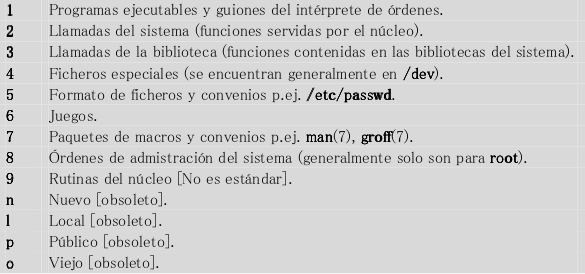


Ejemplos:

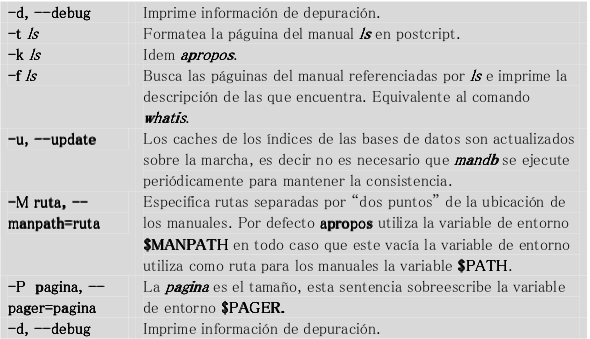


### Comando *man*

Con el comando *man* podemos ver los manuales de los programas y los archivos de configuración. La siguiente tabla muestra los números de sección del manual y los tipos de páginas que contienen.



Con el comando *man*, podemos usar las siguientes opciones:



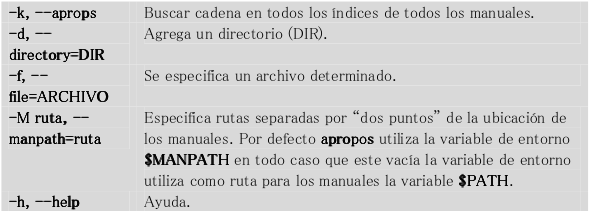
Ejemplos:



Dentro del archivo ***/etc/manpath.config*** contiene los directorios donde se busca la información de los archivos man.

### Comando *info*

Este comando supera la información del man, y nos permite navegar por medio de los enlaces como si fuera una página web. Las opciones que podemos usar con el comando *info* son las siguientes:



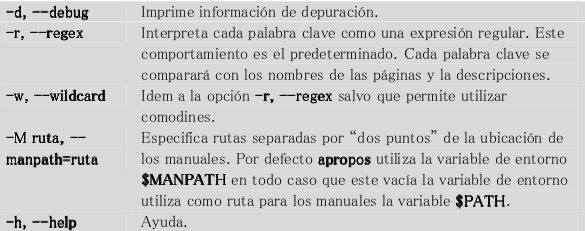
Ejemplos:



Dentro del directorio /usr/share/info se encuentra los archivo de información comprimida.

### Comando *whatis*

El comando *whatis* nos da una pequeña descripción de un comando y las secciones de man que podemos consultar. Las opciones que podemos usar con el comando *whatis* son las siguientes:

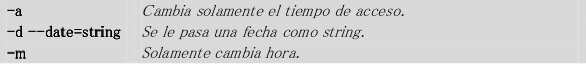


Ejemplos:



### Creación de un archivo vació o cambio de fecha y hora: *touch*

Este comando nos sirve para dos motivos: Si un archivo no existe este comando lo creará con el nombre especificado y con la fecha del momento. Por otro lado, si el archivo existe, le cambiará la fecha y hora. Las opciones que podemos utilizar con el comando *touch* son:

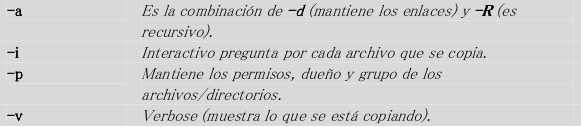


Ejemplos:



### Copia de arhivos: *cp*

El comando *cp* nos permite copiar un archivo o directorio completo. Las opciones que podemos usar con este comando son:



Ejemplo:



Podemos usar los comodines (\* y ?). El \* nos permite un conjunto de caracteres alfanuméricos o un valor alfanumérico.



### Mostrar contenido de archivos: *more*, *less* y *cat*

Tanto el comando *more* y *less* nos muestra el contenido del archivo y solo podemos bajar de a una línea por vez (con la tecla enter) o avanzar por página (con la tecla space). El comando *less* debe ser instalado como paquete apt, usando el comando *apt-get install less*. Además, el comando *less* nos permite realizar búsquedas dentro del archivo. También con las teclas (pgdown) podemos bajar a la página siguiente y con la tecla (pgup) podemos volver a la página anterior.

Ejemplos:

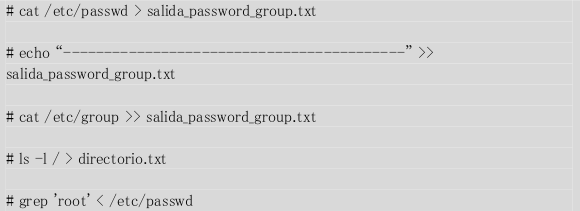


El comando *cat,* cumple la misma función que los comandos *more* o *less*, con la diferencia que el contenido completo se muestra en la pantalla.

### Redirección: >, >>, <

A través de la redirección, podemos tomar la salida de un programa y enviarla automáticamente a un archivo. Este proceso lo maneja la propia shell, en un lugar del programa. La redirección se divide en tres clases: salida a un archivo, añadir al final de un archivo, o envió de un archivo como entrada. Para recoger la salida de un programa en un archivo, finalice la línea del comando con el símbolo mayor que (>) y el nombre del archivo en el cual quiere guardar la salida redirigida.

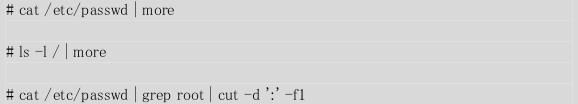
Ejemplos:



### Utilización de pipes

Los pipes son un mecanismo por el cual la salida de un programa se puede enviar como entrada de otros programas. Los programas individuales se pueden encadenar juntos para convertirse en unas herramientas extremadamente potentes.

Ejemplos:



Si quisiera concatenar comandos de forma que no tenga que esperar el resultado de un comando para ejecutar el siguiente, lo que puede hacer es escribirlos todos separándolos con el símbolo “punto y coma” (;) o con &&.

Ejemplo:



Al utilizar (;) si da error el primer comando sigue con el proximo comando.

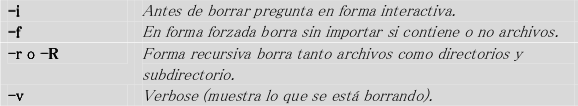


Al utilizar (&&) si da error el primer comando no sigue con el próximo comando.



### Eliminar archivos o directorios: *rm*

El comando *rm* nos permite borrar tanto archivos como directorios completo. Las opciones que podemos utilizar son:



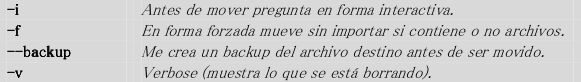
Ejemplo:



Si como root (es decir super usuario) ejecutamos el comando ***rm - rf /*** nos borrara casi todo el sistema operativo, haciendo que el sistema deje de funcionar.

### Mover archivos/directorios o renombrarlos: *mv*

El comando mv nos permite tanto mover como renombrar archivos o directorios. Las opciones que podemos utilizar son:

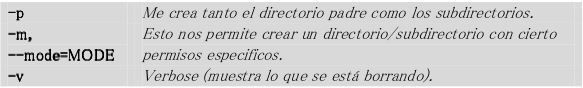


Ejemplos:

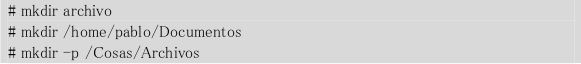


### Creación de un directorio: *mkdir*

Utilizamos el comando *mkdir* cuando necesitamos crear un nuevo directorio. Las opciones que podemos utilizar son:



Ejemplos:



En el siguiente ejemplo, se crean los directorios “Archivos”, “Documentos”, “Fotos” e “Imagenes” dentro del directorio “Datos\_Generales”



En el siguiente ejemplo se crean los directorios “Datos1” y “Datos2” dentro del directorio “Datos”. Además, dentro de los directorios “Datos1” y “Datos2”, se crea un directorio “Datos\_Generales”



### Remover un directorio: *rmdir*

Utilizamos *rmdir* para eliminar directorios. La única opción que podemos utilizar con *rmdir* es -p, que eliminará también la carpeta superior (parent).



### Borrar la pantalla

Para borrar la pantalla, utilizaremos el comando *clear.* En algunos casos es posible utilizar la combinación de teclas *ctrl+l*

### Salir de la terminal

Para salir de la terminal utilizamos el comando *logout.*

## Práctica

1. Iniciar la máquina virtual e iniciar sesión como alumno.
2. Listar todos los archivos, incluyendo los ocultos.
3. Identificar aquellos archivos que sean ocultos.
4. Crear un directorio llamado padre. Dentro del mismo, crear un directorio llamado hijo.
5. Entrar al directorio hijo.
6. Una vez dentro del directorio hijo, crear un archivo vacío llamado archivo1.
7. Escribir el texto “Hola Mundo!” dentro del archivo archivo1.
8. Volver al directorio superior.
9. Crear un archivo llamado archivo2.
10. Copiar el archivo archivo1 dentro del directorio padre.
11. Copiar el contenido de archivo1 en archivo2
12. Mostrar el contenido de ambos archivos (debería mostrarse lo mismo).
13. Eliminar el directorio hijo.

### Respuestas

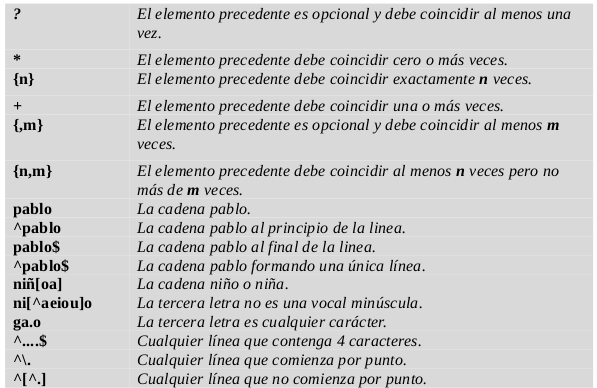
1. Utilization las credenciales usuario: alumno y password: alumno
2. ls -ls
3. Los archivos ocultos son aquellos que están identificados con un .
4. mkdir -p padre/hijo
5. cd padre/hijo
6. touch archivo1
7. echo “Hola mundo!” >> archivo1
8. cd ..
9. touch archivo2
10. cp hijo/archivo1 .
11. cat archivo1 >> archivo2
    1. cat archivo1
    2. cat archivo2
12. rm -rf hijo

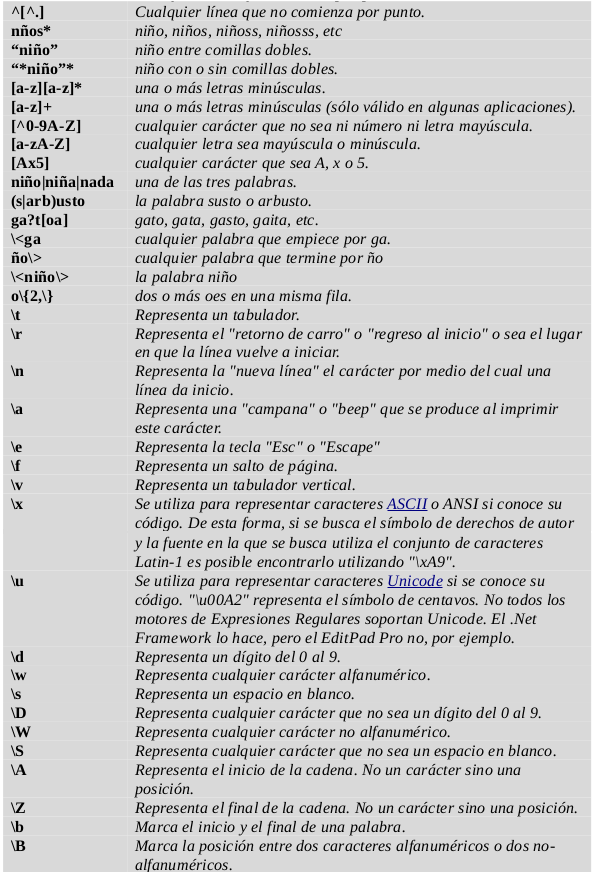
## Expresiones regulares

Las expresiones regulares son generalmente cadenas de caracteres, que se utilizan para representar patrones de texto. Cada caracter en una expresión regular puede representar tanto un caracter específico como un conjunto de caracteres.

Estas expresiones son muy utilizadas, ya que permiten, por ejemplo, representar patrones para realizar búsquedas, validar datos que ingrese el usuario, etc. En este curso, utilizaremos las expresiones regulares para realizar búsquedas en archivos, eliminar archivos que comiencen con un determinado patrón, y algunos usos más.

A continuación se muestra un cuadro con las expresiones regulares, y qué representan las mismas:

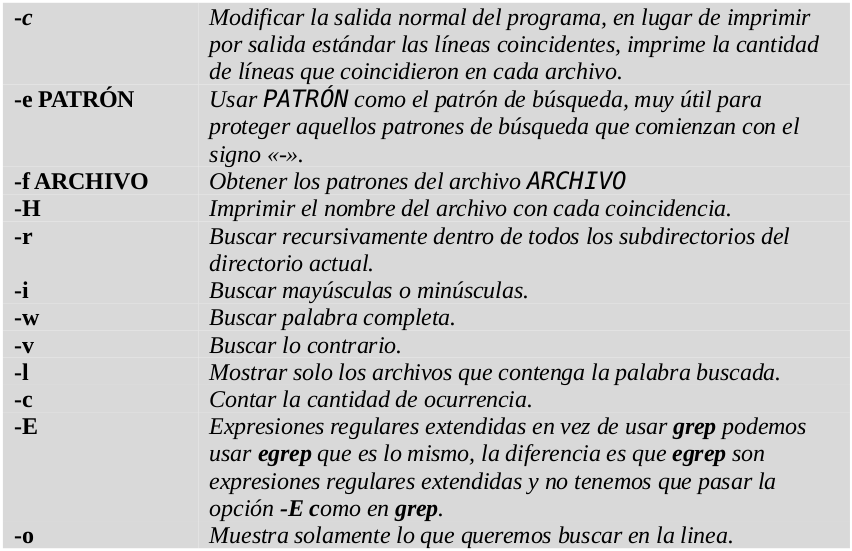




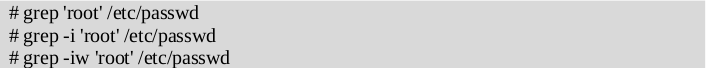
### Búsqueda en archivos utilizando expresiones regulares

El comando **grep** toma una expresión regular de la línea de comandos, lee la entrada estándar o una lista de archivos, e imprime las líneas que contengan coincidencias para la expresión regular.

Las opciones disponibles del comando *grep* son:



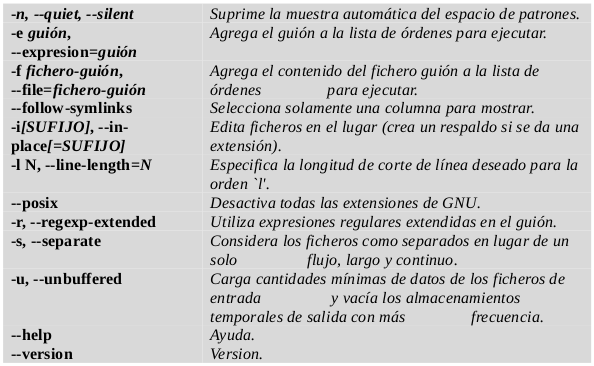
Algunos ejemplos de uso del comando *grep* son:



### Modificar archivos utilizando el comando *sed*

El comando *sed* permite, de una manera simple, modificar un archivo, sustituyendo líneas y reemplazándolas por otras. Para poder indicar qué líneas sustituír, se utilizan, también, expresiones regulares.

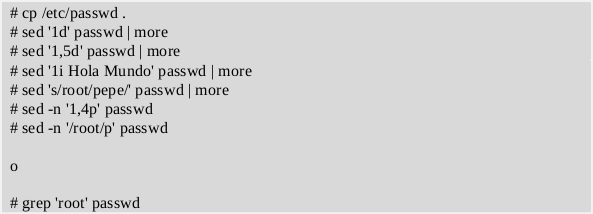
Las opciones que se pueden utilizar con el comando *grep* son:



Además de opciones, el comando *grep* tiene distintos subcomandos que se pueden utilizar:



Algunos ejemplos de uso del comando *sed* son:



### Ejemplo. Comando *grep* para obtener mails de un archivo.

Para poder buscar mails dentro de un archivo, primero debemos construir una expresión regular que represente los mails. Para eso haremos lo siguiente:

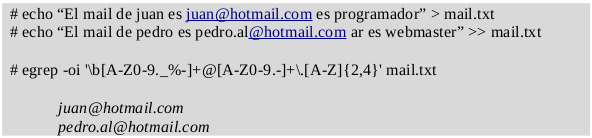
1- Buscamos un comienzo de palabra **\b** que contenga uno o más caracteres **+** dentro del conjunto **A a Z, 0 a 9,., \_, %** y **-**, que son los caracteres admitidos para un nombre de usuario. Si unimos todas estas condiciones, la expresión regular que obtenemos es: **\b[A-Z0-9.\_%-]+**

2- En una dirección de email, luego del nombre de usuario vendrá el caracter @, por lo que nuestra expresión regular tomará la siguiente forma: **\b[A-Z0-9.\_%-]+@**

3- Luego del caracter @, tendremos el dominio. El dominio se forma por uno o más caracteres de la A a Z, 0 a 9,. y -. Por lo tanto nuestra expresión ahora será **\b[A-Z0-9.\_%-]+@[A-Z0-9.-]+**

4- Por último, luego del dominio tendremos un punto (.), ,seguido por 2, 3 o 4 caracteres de la A a la Z. Esto último, además, debe ser el final de una palabra. La expresión final será: **\b[A-Z0-9.\_%-]+@[A-Z0-9.-]+\.[A-Z]{2,4}**

Para el ejemplo, crearemos un archivo llamado mail.txt y le agregaremos dos líneas. En cada una de ellas habrá una dirección de mail.



Tanto el comando *sed* como el comando *grep*, pueden utilizarse sobre la salida de otro comando. Para poder hacer eso, debemos utilizar el caracter | y redirigir la salida del otro comando hacia *grep* o *sed*. Por ejemplo, podríamos utilizar el comando echo para imprimir un número, y esa salida reemplazarla por un string distinto.

