Trabajo Práctico N°4

**Herramientas de líneas de comandos**

**Objetivos:** Desarrollar habilidades para manipular y gestionar archivos y directorios en GNU/Linux utilizando comandos básicos de línea de comandos.

Descarga el archivo almacen.zip del aula virtual y guarda su contenido en el directorio home del usuario con el que iniciaste sesión. Recuerda utilizar el comando **man <comando>**, o bien, **<comando>**

**-- help**, para obtener ayuda de los comandos.

1. Analiza la información que presenta el prompt del sistema.
   1. ¿Qué elementos está mostrando?

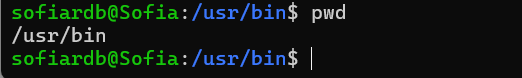


Los elementos que muestra son:

* **Usuario:sofiardb** → el nombre de la cuenta de usuario actual.
* **Equipo:Sofia** → el nombre del host (nombre del sistema o PC).
* **~ o ruta** → el directorio actual en el que se encuentra el usuario (~ es el home).
* **$** → símbolo que indica que es un usuario normal (si fuera root sería #).
  1. Utiliza el comando correspondiente, visualiza el directorio en el que estás actualmente.



* 1. Accede al directorio /usr/bin, y utiliza el mismo comando del apartado anterior para verificar que el directorio actual cambió. ¿Cambió la información del prompt?



Si cambia ya que ahora muestra la ruta al directorio de trabajo.

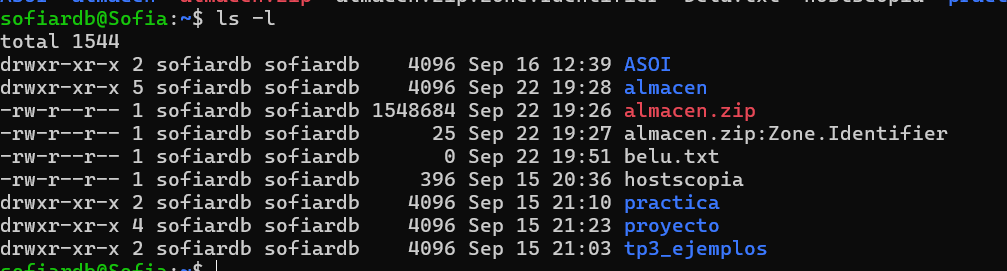
* 1. Vuelve al directorio home del usuario con el comando y las opciones vistas en el práctico anterior.



1. Posiciónate en el directorio home.
   1. Visualiza el contenido del directorio, pero que sólo aparezcan los nombres de los archivos.

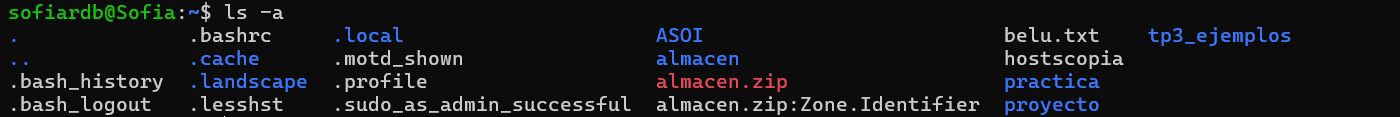


* 1. Visualiza nuevamente, pero mostrando información detallada del contenido. Identifica qué significa la información de cada columna.

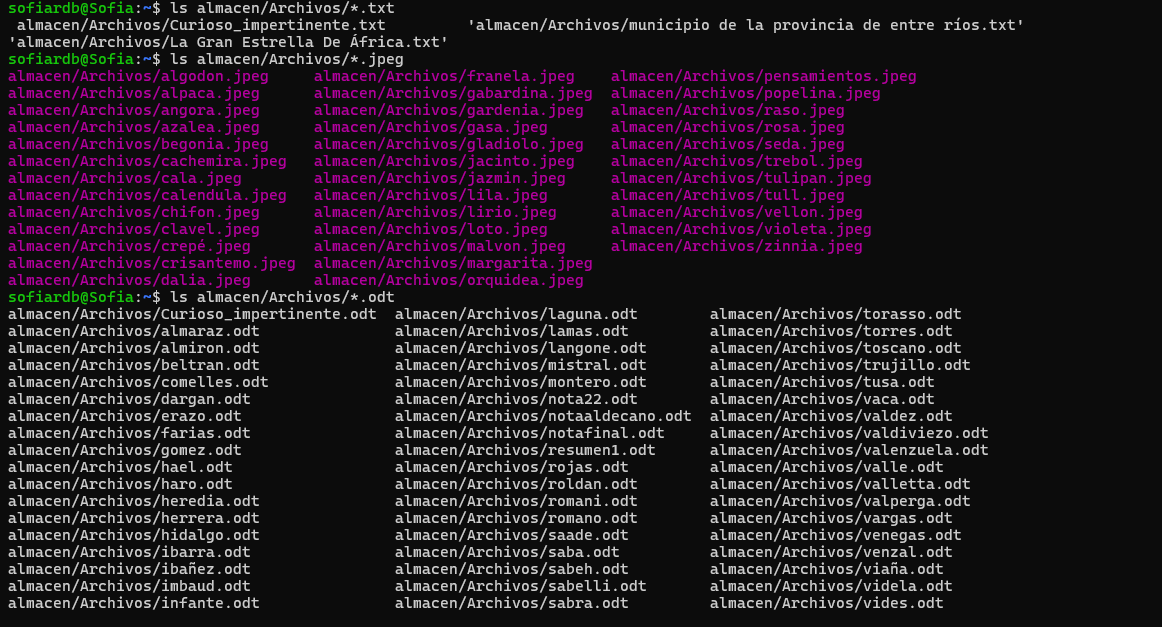


| **Columna** | **Qué muestra** |
| --- | --- |
| 1 | Tipo y permisos del archivo |
| 2 | Número de enlaces |
| 3 | Usuario propietario |
| 4 | Grupo propietario |
| 5 | Tamaño en bytes |
| 6 | Fecha y hora/año de modificación |
| 7 | Nombre del archivo/directorio |

* 1. Visualiza los archivos ocultos del directorio.



* 1. Desde el directorio en el que estás, visualiza el contenido del directorio /almacen/Archivos, mostrando únicamente los archivos con:
     1. Extensión .txt.
     2. Extensión .jpeg.
     3. Extensión .odt.



1. Crea una **secuencia** de comandos que:
   1. Cree un directorio llamado practica\_comandos.
   2. Cambie a ese directorio.
   3. Copie en el directorio los archivos .txt del directorio /almacen/Archivos.
   4. Liste el contenido del directorio.



1. Ejecuta una **secuencia** de comandos que:
   1. Mueva el archivo LaOdisea.txt al directorio /Documentos del home del usuario.
   2. Renombre el archivo como La\_Odisea.txt.
   3. Guarde una copia de seguridad del archivo original en ese directorio.
   4. Guarde una lista del contenido del directorio destino en un archivo llamado Lista.txt.



1. Investiga los comandos sleep y echo. ¿Para qué sirven?

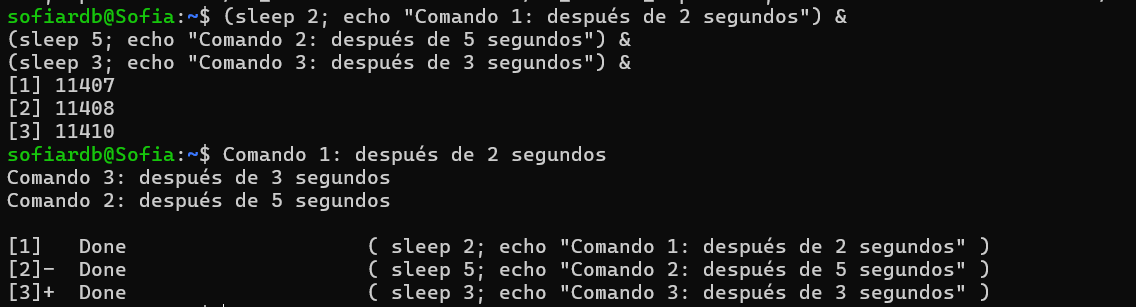
🔹 Comando echo

Función: muestra un texto o el valor de una variable en la terminal. Es muy usado en scripts para imprimir mensajes.

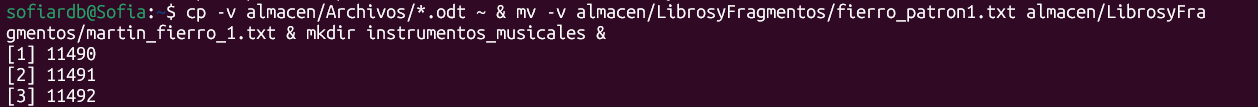
🔹 Comando sleep

Función: pausa la ejecución durante un tiempo determinado en segundos.

* 1. Ejecuta tres comandos en **paralelo**, donde cada uno espere un tiempo diferente antes de imprimir un mensaje.



1. Ejecuta en **paralelo** las siguientes tareas:
   1. Copia los archivos .odt de su ubicación original a la raíz del directorio home.
   2. Renombre el archivo fierro\_patron1.txt como martin\_fierro\_1.txt.
   3. Cree en el home un directorio llamado instrumentos\_musicales.





* 1. Investiga con qué comando se puede monitorear la ejecución de comandos.

El comando más usado en Linux para monitorear la ejecución de un comando en tiempo real es: watch. Te permite ejecutar otro comando de forma repetitiva y ver cómo cambia su salida en tiempo real.

Sintaxis:

watch [opciones] comando

Ejemplo básico:

Monitorear el contenido de un directorio: watch ls -l

Esto ejecutará ls -l cada 2 segundos (valor por defecto) y actualizará la pantalla.

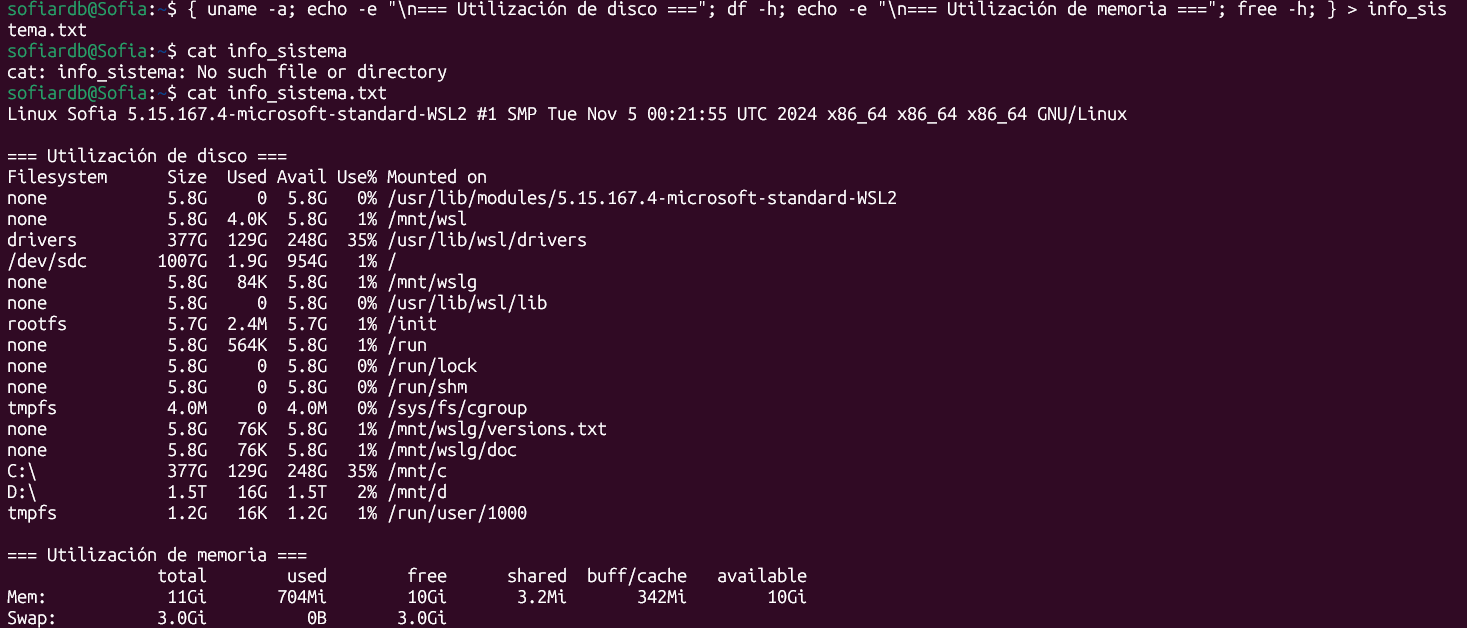
1. En el directorio instrumentos\_musicales, crea la siguiente estructura de directorios, utilizando una sola línea de comandos:





1. Obtén información la siguiente información, y guárdala en un mismo archivo de texto:
   1. Información detallada del sistema.
   2. Utilización de disco.
   3. Utilización de memoria.

Encuentra una manera de guardar esta información en una sola línea de comandos.



**9.** Averigua cómo saber cuántos discos físicos tiene instalado un sistema operativo GNU/Linux y qué tamaños tiene, con algún comando alternativo a los que vimos en clase. Guarda esa información en un archivo.

Para ver los discos físicos instalados y sus tamaños, podés usar comandos alternativos como:

1. lsblk

Muestra los dispositivos de bloques (discos y particiones) en forma de árbol:

lsblk -o NAME,SIZE,TYPE,MODEL

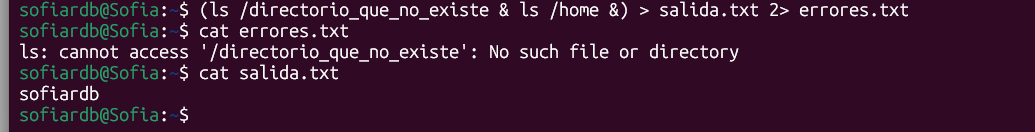
NAME → nombre del dispositivo (ej: sda, sdb).

SIZE → tamaño del disco o partición.

TYPE → si es disk (físico) o part (partición).

MODEL → modelo del disco.

1. Intenta abrir un directorio que no existe y uno que sí existe, ejecutando comandos en paralelo. ¿Es posible redirigir la salida de error de esta línea de comando a un archivo, y la salida estándar a otro, utilizando una vez cada operador de redirección? ¿Esto sería una expresión regular o una característica de la sintaxis del shell?



Esto NO es una expresión regular, sino una característica propia de la sintaxis del shell.

Las expresiones regulares (RegEx) se usan para buscar patrones en texto, por ejemplo con grep.

En este caso, estamos manejando flujos de entrada/salida, algo que forma parte del diseño del shell Bash y otros intérpretes.

1. Busca con el comando grep, en el archivo /almacen/LibrosyFragmentos/curioso\_patron1, todas las líneas que contengan el texto “güenza”. Redirige esa salida al comando sort, y la salida de este último, a un archivo de texto.

