



## Taller Práctico de Sistemas de Archivos (IFCT114PO)


### Ejercicio 1: Creación de estructuras jerárquicas

- **Objetivo:** Aprender a crear rutas de directorios complejas de un solo golpe.
- **Paso a paso:** Ejecuta `mkdir -p Proyectos/2026/Laboratorio`
- **Explicación:** El parámetro `-p` (parents) crea toda la ruta necesaria si los directorios intermedios no existen. 


### Ejercicio 2: Creación de archivos con contenido multilínea

- **Objetivo:** Usar el "Here Document" para generar archivos de configuración.
- **Paso a paso:** `bash cat << EOF > configuracion.txt Usuario: Admin Puerto: 8080 Estado: Activo EOF`
- **Explicación:** `<< EOF` permite escribir varias líneas seguidas hasta que el sistema encuentra la palabra de cierre "EOF". 


### Ejercicio 3: Listado de archivos ocultos y detalles

- **Objetivo:** Identificar archivos de configuración ocultos en el sistema.
- **Paso a paso:** Ejecuta `ls -la /etc | less`
- **Explicación:** `-a` muestra archivos ocultos (los que empiezan por punto) y `-l` muestra permisos y dueño. Usamos `less` para navegar por la lista sin que se escape de la pantalla. 

### Ejercicio 4: Redirección de salida y errores

- **Objetivo:** Separar los datos correctos de los mensajes de error.
- **Paso a paso:** Ejecuta `ls -l /root /home > exitos.txt 2> errores.txt`
- **Explicación:** El sistema intentará listar `/root` (dará error si no eres admin) y `/home`. Los datos irán a `exitos.txt` y el error al archivo de errores. 

### Ejercicio 5: Uso del carácter de escape para nombres con espacios

- **Objetivo:** Gestionar archivos cuyos nombres no siguen los convenios recomendados.
- **Paso a paso:** Ejecuta `mkdir Mis\ Documentos\ Privados`
- **Explicación:** La barra invertida `\` anula el significado especial del espacio, permitiendo que el sistema entienda que es un solo nombre y no tres carpetas distintas. 

### Ejercicio 6: Concatenación de comandos con separador

- **Objetivo:** Ejecutar una secuencia de tareas de forma automatizada.
- **Paso a paso:** Ejecuta `touch prueba.txt ; mv prueba.txt final.txt ; ls -l final.txt`

- **Explicación:** El punto y coma ; permite lanzar varios comandos uno tras otro sin esperar a que el usuario pulse Enter entre ellos. ☹️

#### Ejercicio 7: Acumulación de datos en un log

- **Objetivo:** Registrar eventos sin borrar lo que ya existía en el archivo.
- **Paso a paso:** Ejecuta `date >> historial.log` (hazlo varias veces).
- **Explicación:** El símbolo >> añade la salida al final del archivo. Si usaras solo >, borrarías lo anterior cada vez. 📊

#### Ejercicio 8: Identificación del tipo de archivo (independiente de la extensión)

- **Objetivo:** Comprobar que en Linux la extensión no define el archivo.
- **Paso a paso:** Ejecuta `touch imagen.jpg` y luego `file imagen.jpg`
- **Explicación:** El comando `file` analiza el contenido real. Verás que te dice "empty", demostrando que aunque se llame .jpg, el sistema sabe que no es una imagen. 🧐

#### Ejercicio 9: Copia recursiva de seguridad

- **Objetivo:** Clonar una estructura de directorios completa.
- **Paso a paso:** Ejecuta `cp -r Proyectos Proyectos_Backup`
- **Explicación:** La opción -r (recursive) es obligatoria para copiar directorios; de lo contrario, el comando daría error al encontrar carpetas dentro. 🙌

#### Ejercicio 10: Limpieza selectiva de temporales

- **Objetivo:** Eliminar archivos con precaución.
- **Paso a paso:** Ejecuta `rm -i final.txt`
- **Explicación:** El parámetro -i (interactive) hace que el sistema te pregunte "¿está seguro?" antes de borrar. Es la mejor práctica para evitar desastres en la terminal. 🧹