



Taller Práctico: Comandos de Sistema y Utilidades (IFCT114PO)


Ejercicio 11: Creación de archivos con marcas de tiempo específicas

- **Objetivo:** Utilizar `touch` para manipular fechas de acceso.
- **Paso a paso:** Ejecuta `touch -t 202501010000 archivo_viejo.txt`
- **Explicación:** El comando `touch` no solo crea archivos vacíos; con `-t` puedes "viajar en el tiempo" y asignarle al archivo una fecha de creación personalizada (AAAAMMDDhhmm). 


Ejercicio 12: Visualización de variables de entorno

- **Objetivo:** Usar `echo` para extraer información del sistema.
- **Paso a paso:** Ejecuta `echo "El usuario actual es: $USER y mi ruta es: $HOME"`
- **Explicación:** El comando `echo` sirve para imprimir texto, pero al usar el símbolo `$` podemos mostrar el contenido de las variables de entorno del sistema. 


Ejercicio 13: Localización de ejecutables

- **Objetivo:** Saber exactamente qué versión de un programa estamos usando.
- **Paso a paso:** Ejecuta `which python3` o `which ls`
- **Explicación:** `which` busca en las rutas del sistema (PATH) y nos devuelve la ruta completa del binario que se ejecuta cuando escribimos el nombre del comando. 

Ejercicio 14: Formateo personalizado de la fecha

- **Objetivo:** Obtener la fecha en un formato específico para scripts.
- **Paso a paso:** Ejecuta `date +"Hoy es: %d/%m/%Y y son las: %H:%M"`
- **Explicación:** El comando `date` permite usar "formateadores" (empezando por `+`) para mostrar solo la información que nos interesa de forma legible. 

Ejercicio 15: Cambio de contraseña de usuario

- **Objetivo:** Entender la gestión de seguridad básica.
- **Paso a paso:** Ejecuta `passwd` (y sigue las instrucciones, o pulsa `Ctrl+C` para cancelar).
- **Explicación:** `passwd` es la utilidad estándar para cambiar la contraseña del usuario actual. Si se usa como `root` (`sudo passwd usuario`), permite cambiar la de otros. 

Ejercicio 16: Creación de múltiples archivos en serie

- **Objetivo:** Ahorrar tiempo usando expansiones con `touch`.
- **Paso a paso:** Ejecuta `touch documento_{1..5}.docx`

- **Explicación:** Al usar llaves {} con un rango, touch crea automáticamente múltiples archivos (documento_1, documento_2, etc.) en un solo paso. 📄

✎ Ejercicio 17: Impresión de texto sin salto de línea

- **Objetivo:** Controlar el comportamiento de echo en scripts.
- **Paso a paso:** Ejecuta `echo -n "Cargando sistema... " ; sleep 2 ; echo "OK"`
- **Explicación:** La opción -n evita que echo baje a la siguiente línea, permitiendo que el siguiente comando escriba justo a continuación. 💡

✎ Ejercicio 18: Cálculo de fechas futuras o pasadas

- **Objetivo:** Usar la inteligencia del comando date.
- **Paso a paso:** Ejecuta `date -d "next friday"` o `date -d "last monday"`
- **Explicación:** La opción -d (display) permite pasarle frases en lenguaje natural (en inglés) para que el comando calcule la fecha correspondiente automáticamente. 🌟

✎ Ejercicio 19: Búsqueda de la ubicación de comandos del sistema

- **Objetivo:** Diferenciar entre comandos internos y programas instalados.
- **Paso a paso:** Ejecuta `which passwd`
- **Explicación:** Nos indica que passwd se encuentra en /usr/bin/passwd. Esto confirma que es una utilidad instalada en el sistema de archivos jerárquico. 📁

✎ Ejercicio 20: Combinación de echo y redirección para archivos de sistema

- **Objetivo:** Crear archivos de texto rápido sin editores.
- **Paso a paso:** Ejecuta `echo "nameserver 8.8.8.8" > dns.conf`
- **Explicación:** Combinamos echo con la redirección > para crear directamente un archivo con una línea de configuración técnica. 🔧