

Ejercicios introductorios en el terminal

1. Ejercicio de Listar Contenido: Utiliza el comando "ls" para listar el contenido del directorio actual.
2. Ejercicio de Directorio Actual: Usa el comando "pwd" para mostrar el directorio actual en el que te encuentras.
3. Ejercicio de Copiar un Archivo: Crea un archivo de texto llamado "archivo.txt" en el directorio actual. Copia el archivo "archivo.txt" y nómbralo como "copia.txt".
4. Ejercicio de Eliminar Archivos: Crea dos archivos de texto llamados "archivo1.txt" y "archivo2.txt" en el directorio actual. Utiliza el comando "rm" para eliminar el archivo "archivo1.txt".
5. Ejercicio de Mover Archivos: Mueve el archivo "archivo2.txt" a un nuevo directorio llamado "Documentos".
6. Ejercicio de Edición de Archivos con nano: Crea un archivo de texto llamado "notas.txt" usando el comando "nano". Escribe algunas líneas de texto en el archivo y guárdalo.
7. Ejercicio de Edición de Archivos con vim: Crea un archivo de texto llamado "texto.txt" usando el comando "vim". Escribe algunas líneas de texto en el archivo y guárdalo.
8. Ejercicio de visualización parcial de archivos: Crea un archivo de texto con más de 20 líneas. Usa el comando "head" para mostrar las 5 primeras líneas del archivo. Usa el comando "tail" para mostrar las últimas 5 líneas del archivo.
9. Ejercicio de Creación de Directorios: Crea un nuevo directorio llamado "Proyecto".
10. Ejercicio de Creación de Subdirectorios: Navega dentro del directorio "Proyecto". Crea dos subdirectorios llamados "src" y "docs" dentro del directorio "Proyecto".
11. Ejercicio de Creación de Múltiples Archivos: Crea tres archivos vacíos llamados "file1.txt", "file2.txt", y "file3.txt" en el directorio actual utilizando el comando "touch" en una sola línea.
12. Ejercicio de Creación de Archivos Vacíos: Crea un archivo vacío llamado "nuevo.txt" en el directorio "Proyecto/src".
13. Ejercicio de Copiar un Directorio: Copia todo el contenido del directorio "Proyecto/src" al directorio "Proyecto/docs".

14. Ejercicio de Renombrar Archivos: Renombra el archivo "nuevo.txt" a "renombrado.txt" en el directorio "Proyecto/src".
15. Ejercicio de Mover un Directorio: Mueve el directorio "Proyecto/docs" a un nuevo directorio llamado "ProyectosAntiguos".
16. Ejercicio de Listar Contenido con Detalles: Utiliza el comando "ls" con la opción para listar el contenido del directorio actual en formato largo.
17. Ejercicio de Concatenación de Archivos: Crea dos archivos de texto llamados "parte1.txt" y "parte2.txt". Escribe algunas líneas de texto en cada uno. Usa el comando cat para concatenar el contenido de ambos archivos y redirige la salida a un nuevo archivo llamado "completo.txt".
18. Ejercicio de Eliminación Recursiva: Crea un directorio llamado "Eliminar" y dentro de él, crea algunos archivos y subdirectorios. Utiliza el comando "rm" con la opción para eliminar el directorio "Eliminar" y todo su contenido de forma recursiva.
19. Ejercicio de Copiar y Renombrar con mv: Crea un archivo de texto llamado "original.txt" en el directorio actual. Copia y renombra el archivo "original.txt" como "copia-original.txt" usando el comando "mv".
20. Ejercicio de Creación de Directorios Anidados: Crea un nuevo directorio llamado "Proyecto/NuevaCarpeta/Pruebas".
21. Ejercicio de Edición de Archivos con nano y vim: Abre el archivo "texto.txt" creado anteriormente con "nano" y realiza cambios en el contenido. Abre el archivo "notas.txt" creado anteriormente con "vim" y realiza cambios en el contenido.
22. Crear un archivo llamado "registro.log" y añade al menos 10 líneas de contenido. Algunas de esas líneas deberán contener la palabra "error".
Ejemplo:

```
Unset
hola buenos dias
esto es error
una prueba
de error
para ver
como funcionan
los operadores
y la
redirección error
error
muchos errores
```

23. Utiliza un pipe para contar cuántas veces aparece la palabra "error" en un archivo de registro llamado "registro.log".
24. Redirige la salida del comando "ls" a un archivo llamado "lista_archivos.txt".
25. Ejecuta el comando "echo 'Este es un mensaje'" y redirige la salida a un archivo llamado "mensaje.txt".
26. Utiliza el operador > para crear un archivo llamado "salida_comando.txt" que contenga la salida del comando "ls -l".
27. Utiliza el operador >> para agregar la salida del comando "date" al final de un archivo llamado "registro_fecha.txt". Prueba a ejecutarlo varias veces y ve observando el contenido del archivo de registro.
28. Ejecuta el comando "rm archivo_inexistente.txt" y, utilizando &&, muestra un mensaje que diga "El archivo ha sido eliminado" solo si el comando anterior tiene éxito.
29. Ejecuta el comando "ls directorio_inexistente" y, utilizando ||, muestra un mensaje que diga "El directorio no existe" solo si el comando anterior falla.

Ejercicios extra (más dificultad)

1. Ejercicio de Búsqueda de Texto en Archivos: Crea un archivo de texto llamado "data.txt" con varias líneas de contenido. Utiliza el comando "grep" "palabra" para buscar una palabra específica dentro del archivo "data.txt".
2. Crea un archivo llamado "informacion.txt" que contenga múltiples líneas, algunas con números. Usa un comando que filtre y muestre solo las líneas que contienen números, redirigiendo la salida a un archivo llamado "lineas_con_numeros.txt".
3. Ejercicio de Redirección y Pipes: Crea un archivo llamado "historial.log" con varias entradas de texto. Utiliza un pipe para contar cuántas líneas contiene el archivo y redirige este conteo a un archivo llamado "conteo.txt".
4. Ejercicio de Comparación de Archivos: Crea dos archivos llamados "archivoA.txt" y "archivoB.txt". Escribe contenido diferente en ambos archivos. Usa el comando "diff" para comparar los archivos y redirige las diferencias a un archivo llamado "diferencias.txt".
5. Ejercicio de Scripts Básicos: Escribe un pequeño script en bash que realice lo siguiente: Cree un directorio llamado "Backup". Mueve todos los archivos .txt del directorio actual a "Backup". Cree un archivo de registro llamado "backup.log" que registre la fecha y hora en que se realizó la copia.

6. Ejercicio de Combinación de Comandos con Pipes y Redirección: Crea un archivo llamado "log_sistema.log" con varias líneas que simulen entradas de un log de sistema. Filtra las líneas que contienen la palabra "ERROR", cuenta cuántas veces aparece la palabra y redirige el resultado a un archivo llamado "reporte_errores.txt".
7. Ejercicio de Redirección Condicional: Intenta eliminar un archivo llamado "importante.txt". Si el archivo existe y se elimina correctamente, redirige un mensaje de éxito a un archivo llamado "resultado.txt". Si no existe, redirige un mensaje de error a "resultado.txt".