

Bash Script

¿Qué es un Bash Script?

Un Bash Script es simplemente un archivo de texto que contiene comandos y lógica escritos en el lenguaje Bash, y que se ejecutan en orden como si los estuvieras tipeando en la terminal.

¿Para qué sirve?

Para **automatizar tareas** en sistemas GNU/Linux (y también Windows vía Git Bash o WSL). Desde copiar archivos hasta instalar programas, lanzar backups, procesar datos, monitorear servidores... todo.

Buenas prácticas

- Usá `#!/bin/bash` siempre.
- Comentá tu código
- Validá la entrada del usuario.
- Usá nombres descriptivos de variables.
- Separa lógica compleja en funciones.

Herramientas para probar

- Git Bash (Windows)
- WSL (Windows Subsystem for Linux)
- Terminal de Linux o MacOS
- <https://www.jdoodle.com/test-bash-shell-script-online>

Actividades

Ejercicio 1: Saludar al usuario

Objetivo: Crear un script que pida el nombre del usuario y luego lo salude.

Guía:

- Primero, el script debe pedir al usuario que ingrese su nombre usando el comando `read`.
- Luego, el script debe imprimir un saludo utilizando el comando `echo` y la variable que almacena el nombre ingresado.

2 Ejercicio 2: Verificar si un número es par o impar

Objetivo: Crear un script que reciba un número y diga si es par o impar.

Guía:

- El número debe ser ingresado por el usuario.
- Para determinar si el número es par, se puede usar el operador módulo (%), que da el residuo de la división.
- Si el residuo de la división del número entre 2 es 0, el número es par, de lo contrario es impar.

3 Ejercicio 3: Mostrar el número mayor entre dos números

Objetivo: Crear un script que pida dos números y muestre el mayor.

Guía:

- Se deben pedir dos números al usuario.
- El script debe comparar ambos números usando condicionales (if, elif, else).
- Finalmente, mostrar el número mayor en la pantalla.

4 Ejercicio 4: Mostrar los números del 1 al 10

Objetivo: Crear un script que recorra los números del 1 al 10 y los imprima.

Guía:

- Se utilizará un bucle for para iterar sobre los números del 1 al 10.
- El comando echo se usará para imprimir cada número en cada iteración del bucle.

5 Ejercicio 5: Menú interactivo con case

Objetivo: Crear un menú donde el usuario pueda elegir una opción para mostrar la fecha, los usuarios conectados o salir.

Guía:

- Usar el comando read para permitir al usuario seleccionar una opción del menú.
- Usar la estructura case para manejar las diferentes opciones del menú.
- Cada opción ejecutará un comando diferente, como mostrar la fecha con date o listar los usuarios con who.

Hoja de apuntes con los comandos de bash basicos

Elemento	Ejemplo / Uso
Shebang (primer linea del script)	<code>#!/bin/bash</code>
Comentarios	<code># Esto es un comentario</code>
Variable	<code>nombre="Juan"</code>
Mostrar texto	<code>echo "Hola"</code>
Leer entrada	<code>read -p "Tu nombre: " nombre</code>
If/Else	<code>if ["\$x" -gt 0]; then ... fi</code>
Comparadores	<code>-eq -ne -gt -lt -ge -le</code>
for loop	<code>for i in {1..5}; do ... done</code>
while loop	<code>while [\$i -le 5]; do ... done</code>
case	<code>case \$opcion in 1) ... ;; esac</code>
Crear carpeta	<code>mkdir nombre</code>
Crear archivo	<code>touch archivo.txt</code>
Escribir archivo	<code>echo "Texto" > archivo.txt</code>
Leer archivo	<code>cat archivo.txt</code>
Ejecutar script	<code>bash script.sh o ./script.sh</code>
Hacer ejecutable	<code>chmod +x script.sh</code>