

# **SHELL SCRIPTING: CONTROL DE FLUJO Y PROGRAMACIÓN MODULAR**

# ÍNDICE

- Condicionales y bucles
- Programación modular (bibliotecas)
- Buenas prácticas

# CONDICIONALES

- if [comparación];then

tareas

fi

- case \$var in

valor)

tareas

::

\*)

tareas default

::

esac

# CONDICIONALES

- Comparando cadenas
  - = : igual
  - != : distinto
- Comparando enteros
  - gt mayor
  - lt menor
  - ge mayor o igual
  - le menor o igual
  - eq igual
  - ne distinto

# BUCKLES

- for i in {1..1000} / for i in `seq 1 100`
- for ((i=1;i<=1000;i++))
- while [ \$i -lt 1000 ];do

tareas

((i++))

done

# PROGRAMACIÓN MODULAR

- Creación de bibliotecas
- Reutilización de código
  - Estabilidad y eficiencia
- Cargar biblioteca:
  - source /ruta/biblioteca.sh

# BUENAS PRÁCTICAS

- Modularización y reutilización de código
- Documentación y comentarios detallados
- Validación y pruebas automatizadas
- Seguridad en scripts: evitar datos y contraseñas en texto claro