

# EJERCICIOS DE CONTROL DE FLUJOS

---

## EJERCICIO 1 - IF

Crea un script en Bash que reciba como parámetro el nombre de un archivo y verifique si existe en el directorio actual. Si existe, imprime un mensaje que incluya el nombre del archivo y su tamaño. Si no existe, imprime un mensaje de error.

```
#!/bin/bash

# Verificamos que se haya proporcionado un parámetro
if [ $# -eq 0 ]
then
echo "Error: Debes proporcionar el nombre de un archivo como parámetro."
exit 1
fi

# Verificamos si el archivo existe en el directorio actual
if [ -e $1 ]
then
# Obtenemos el tamaño del archivo
tamano=$(du -h $1 | cut -f1)

# Imprimimos un mensaje que incluye el nombre del archivo y su tamaño
echo "El archivo $1 existe en el directorio actual y tiene un tamaño de $tamano."
else
echo "Error: El archivo $1 no existe en el directorio actual."
exit 1
fi
```

## EJERCICIO 2 - FOR Y ARRAYS

Crea un script en Bash que imprima en la pantalla un mensaje que incluya el nombre del script, el número de argumentos que se proporcionan y los valores de cada uno de los argumentos.

```
#!/bin/bash

# Imprimimos un mensaje que incluye el nombre del script
echo "El nombre del script es: $0"

# Imprimimos un mensaje que indica el número de argumentos proporcionados
echo "Se han proporcionado $# argumentos:"

# Imprimimos los valores de cada uno de los argumentos
echo "Los valores de los argumentos son: $@"
```

```
# Imprimimos los valores de cada uno de los argumentos utilizando un bucle
"for"
echo "Los valores de los argumentos son:"

for argumento in "$@"
do
echo "- $argumento"
done
```

## EJERCICIO 3 - IF

Crea un script en Bash que imprima en la pantalla un mensaje que incluya el nombre del script, el número de argumentos que se proporcionan y los valores de cada uno de los argumentos.

```
#!/bin/bash

# Solicitamos al usuario su edad
echo "Ingresa tu edad:"
read edad

# Verificamos si el usuario es mayor o menor de edad
if [ $edad -ge 18 ]
then
    echo "Eres mayor de edad"
else
    echo "Eres menor de edad"
fi
```

## EJERCICIO 4 - IF

Crea un script en Bash que imprima en la pantalla un mensaje que incluya el nombre del script, el número de argumentos que se proporcionan y los valores de cada uno de los argumentos.

```
#!/bin/bash

# Definimos la contraseña
contraseña="contraseña123"

# Solicitamos al usuario la contraseña
echo "Ingresa la contraseña:"
read contraseña_ingresada

# Verificamos si la contraseña ingresada es correcta
if [ $contraseña_ingresada == $contraseña ]
then
    echo "Contraseña correcta"
else
    echo "Contraseña incorrecta"
fi
```

## EJERCICIO 5 - CASE

Crea un script en Bash que solicite al usuario un número del 1 al 7 y que imprima en pantalla el día de la semana correspondiente.

```
#!/bin/bash

echo "Ingresa un número del 1 al 7:"
read numero

case $numero in
    1)
        echo "Lunes";;
    2)
        echo "Martes";;
    3)
        echo "Miércoles";;
    4)
        echo "Jueves";;
    5)
        echo "Viernes";;
    6)
        echo "Sábado";;
    7)
        echo "Domingo";;
    *)
        echo "Número no válido";;
esac
```

## EJERCICIO 6 - CASE

Crea un script en Bash que solicite al usuario una letra y que imprima en pantalla si la letra ingresada es vocal o consonante.

```
#!/bin/bash

echo "Ingresa una letra:"
read letra

case $letra in
    [AEIOUaeiou])
        echo "La letra ingresada es una vocal";;
    [BCDFGHJKLMNPQRSTVWXYZbcd fghjklmnpqrstvwxyz])
        echo "La letra ingresada es una consonante";;
    *)
        echo "No se reconoce la letra ingresada";;
esac
```

## EJERCICIO 6 - WHILE

Crea un script en Bash que solicite al usuario una letra y que imprima en pantalla si la letra ingresada es vocal o consonante.

```
#!/bin/bash

# Definimos la contraseña correcta
password="contraseña123"

# Solicitamos al usuario que ingrese una contraseña
echo "Ingresa tu contraseña:"
read input_password

# Utilizamos un bucle while para seguir solicitando la contraseña hasta que
sea correcta
while [ "$input_password" != "$password" ]
do
    echo "Contraseña incorrecta. Inténtalo de nuevo:"
    read input_password
done

echo "Contraseña correcta. Bienvenido."
```

## EJERCICIO 7 - UNTIL

Crea un script en Bash que solicite al usuario un número mayor o igual a 10 y menor o igual a 20. Si el número ingresado no cumple con esta condición, el script debe solicitar al usuario que ingrese otro número hasta que se cumpla la condición. Cuando se ingrese un número válido, el script debe imprimir en la pantalla el mensaje "El número ingresado es válido".

```
#!/bin/bash

# Solicitamos al usuario un número mayor o igual a 10 y menor o igual a 20
read -p "Ingresa un número mayor o igual a 10 y menor o igual a 20: " numero

# Validamos que el número ingresado cumpla con la condición
until ((numero >= 10 && numero <= 20))
do
    read -p "El número ingresado no cumple con la condición. Ingresa otro número: " numero
done

# Imprimimos en pantalla el mensaje de que el número ingresado es válido
echo "El número ingresado es válido"
```