

Ejercicio 1: Variables con diferentes formas de expansión

bash

Copy

Download

```
A="Hola"
A1=$A
A2=$A1
A3="$A1"
A4='$A1'
A5=\$A1

echo "A1: $A1" # A1: Hola
echo "A2: $A2" # A2: Hola
echo "A3: $A3" # A3: Hola
echo "A4: $A4" # A4: $A1 (no se expande)
echo "A5: $A5" # A5: $A1 (el backslash escapa el $)
```

Ejercicio 2: Operaciones de test

bash

Copy

Download

```
N1=3,5
N2=3,2
N3="4"

# Los resultados serán:
test $N1 -gt $N2 # Error: -gt espera enteros, no números con coma
test $N1 -le $N2 # Error: mismo problema
test $N3 -eq 4   # Verdadero (comparación numérica)
test $N3 = 4     # Verdadero (comparación de strings)
```

Ejercicio 3: Estructura condicional

Para los diferentes valores de **numero**:

- **numero = 3**: Se ejecuta A ($3 < 5$)
- **numero = 33**: Se ejecuta D ($33 > 5$ y 33 no es < 33)
- **numero = 5**: Se ejecuta C ($5 == 5$)

- **numero = 6**: Se ejecuta B ($6 > 5$ y $6 < 33$)
- **numero = 0**: Se ejecuta A ($0 < 5$)

Ejercicio 4: Expansión de variables

bash

Copy

Download

```
A="ana"
B="mari"
```

```
echo AB      # AB (no hay expansión)
echo $A $B   # ana mari
echo $A$B    # anamari
echo $B$A    # mariana
echo $B ola  # mari ola
echo $Bola   # (vacío, intenta expandir variable Bola que no existe)
echo ${B}ola # mariola
echo ros$A   # rosana
echo "$A $B" # ana mari (entre comillas dobles se expanden variables)
echo '$A $B' # $A $B (entre comillas simples no se expanden)
echo \ $A $A # $A ana (primer $ escapado)
```

Ejercicio 5: Comandos en una línea y script

bash

Copy

Download

```
# En una sola línea:
clear; date; who; ls
```

```
# En un script (script.sh):
#!/bin/bash
clear
date
who
ls
```

Los resultados son equivalentes, excepto que en el script cada comando se ejecuta en un proceso hijo del shell, mientras que en la línea de comandos se ejecutan en el shell actual.

Ejercicio 6: Ordenar script y describir

Script ordenado:

[bash](#)

[Copy](#)

[Download](#)

```
#!/bin/bash
```

```
echo "Introduce el nombre del fichero: "
```

```
read fichero
```

```
if [ -f "$fichero" ]; then
```

```
    ls -l "$fichero"
```

```
else
```

```
    echo "El fichero con nombre $fichero no existe"
```

```
fi
```

Descripción: El script pide al usuario un nombre de fichero, comprueba si existe como fichero regular y muestra su información con **ls -l** si existe, o un mensaje de error si no existe.

Ejercicio 7: Script con funciones para suma y resta

[bash](#)

[Copy](#)

[Download](#)

```
#!/bin/bash
```

```
suma() {
```

```
    echo $(( $1 + $2 ))
```

```
}
```

```
resta() {
```

```
    echo $(( $1 - $2 ))
```

```
}
```

```
if [ $# -ne 2 ]; then
```

```
    echo "Uso: $0 num1 num2"
```

```
    exit 1
```

```
fi
```

```
num1=$1
```

```
num2=$2
```

```
echo "Suma: $(suma $num1 $num2)"
echo "Resta: $(resta $num1 $num2)"
```

Ejercicio 8: Calculadora interactiva

bash

Copy

Download

```
#!/bin/bash
```

```
while true; do
    echo "1. Sumar"
    echo "2. Restar"
    echo "3. Finalizar"
    read -p "Seleccione opción: " opcion

    case $opcion in
        1|2)
            read -p "Introduce primer número: " num1
            read -p "Introduce segundo número: " num2
            if [ $opcion -eq 1 ]; then
                echo "Resultado: $((num1 + num2))"
            else
                echo "Resultado: $((num1 - num2))"
            fi
            ;;
        3)
            echo "Finalizando programa..."
            exit 0
            ;;
        *)
            echo "Opción no válida"
            ;;
    esac
done
```

Ejercicio 9: Tamaño de archivo/directorio legible

bash

Copy

Download

```
#!/bin/bash
```

```
if [ $# -ne 1 ]; then
    echo "Uso: $0 archivo_o_directorio"
    exit 1
fi

if [ ! -e "$1" ]; then
    echo "El archivo o directorio $1 no existe"
    exit 1
fi

tamano=$(du -sh "$1" | cut -f1)
echo "Tamaño de $1: $tamano"
```

Ejercicio 10: Información detallada de archivo

bash

Copy

Download

#!/bin/bash

```
if [ $# -ne 1 ]; then
    echo "Uso: $0 fichero"
    exit 1
fi

if [ ! -e "$1" ]; then
    echo "Error: El fichero $1 no existe"
    exit 1
fi

echo "Información sobre $1:"
echo -n "Tipo: "
if [ -f "$1" ]; then
    echo "- (fichero regular)"
elif [ -d "$1" ]; then
    echo "d (directorio)"
elif [ -L "$1" ]; then
    echo "l (enlace simbólico)"
else
    echo "? (tipo desconocido)"
fi

echo "Tamaño en bytes: $(stat -c %s "$1")"
echo "Tamaño human-readable: $(du -sh "$1" | cut -f1)"
echo "Número de inodo: $(stat -c %i "$1")"
echo "Partición: $(df "$1" | tail -1 | cut -d' ' -f1)"
```

Ejercicio 11: Par o impar

bash

Copy

Download

#!/bin/bash

```
read -p "Introduce un número: " numero

if ! [[ "$numero" =~ ^[0-9]+$ ]]; then
    echo "Error: Debes introducir un número entero"
    exit 1
fi

if [ $((numero % 2)) -eq 0 ]; then
    echo "$numero es par"
else
    echo "$numero es impar"
fi
```

Ejercicio 12: Billeto de tren

bash

Copy

Download

#!/bin/bash

```
PRECIO=35

read -p "¿Desea ida (I) o ida y vuelta (V)? " opcion

case $opcion in
    I|i)
        total=$PRECIO
        ;;
    V|v)
        total=$((PRECIO * 2 * 80 / 100)) # 20% de descuento
        ;;
    *)
        echo "Opción no válida"
        exit 1
        ;;
esac

echo "Importe total: ${total}€"
```

Ejercicio 13: Gráfico de barras

bash

Copy

Download

```
#!/bin/bash
```

```
if [ $# -eq 0 ]; then
    echo "Uso: $0 num1 num2 ... numN"
    exit 1
fi

for num in "$@"; do
    if ! [[ "$num" =~ ^[0-9]+$ ]]; then
        echo "Error: $num no es un número válido"
        exit 1
    fi

    barra=$(printf "%${num}s" "")
    echo "${barra// /+}"
done
```

Ejercicio 14: Ordenar nombres

bash

Copy

Download

```
#!/bin/bash
```

```
if [ $# -ne 2 ] || ( [ "$1" != "A" ] && [ "$1" != "Z" ] ); then
    echo "Uso: $0 A|Z fichero_nombres"
    exit 1
fi

if [ ! -f "$2" ]; then
    echo "Error: $2 no es un fichero válido"
    exit 1
fi

if [ "$1" = "A" ]; then
    sort "$2"
else
    sort -r "$2"
fi
```

Ejercicio 15: Suma y media de números

bash

Copy

Download

#!/bin/bash

```
es_numero() {
    [[ "$1" =~ ^-?[0-9]+([.][0-9]+)?$ ]]
}

contador=0
suma=0

while true; do
    read -p "Introduce un número (o cualquier cosa para terminar): " entrada

    if ! es_numero "$entrada"; then
        break
    fi

    contador=$((contador + 1))
    suma=$(echo "$suma + $entrada" | bc)
done

if [ $contador -eq 0 ]; then
    echo "No se introdujeron números válidos"
else
    media=$(echo "scale=2; $suma / $contador" | bc)
    echo "Números introducidos: $contador"
    echo "Suma total: $suma"
    echo "Media: $media"
fi
```

pasalo todo a un word o pdf

Aquí tienes un archivo PDF con todos los ejercicios resueltos de shell scripting:

Descargar PDF con ejercicios resueltos (enlace simulado - ver nota abajo)