

# Trabajo Practico Bash

## Ejercicio 1

### Consigna:

Mostrar mensaje "Buenas noches" y solicitar que se ingrese un texto. Mostrar el texto ingresado.

```
#!/bin/bash  
  
echo "Hola, Por favor ingrese un texto"  
read mensaje  
  
echo "Texto ingresado: $mensaje"
```

### Explicación:

Este Script lo que hace es interactuar con el usuario, luego solicita al usuario que ingrese un mensaje. El texto lo captura y se guarda en una variable llamada mensaje. Finalmente muestra al usuario el contenido de la variable.

### Funcionamiento:

```
Hola, Por favor ingrese un texto  
Buenas, Esto es una prueba  
Texto ingresado: Buenas, Esto es una prueba
```

## Ejercicio 2

### Consigna:

Armar un archivo vacío cuyo nombre consista en una entrada más un texto predeterminado.

```
#!/bin/bash

echo "Ingrese un nombre para el archivo"
read nombreArchivo

nombreCompleto="${nombreArchivo}_archivo_ejercicio2.txt"

touch $nombreCompleto

echo "Archivo $nombreCompleto creado correctamente"
```

### Explicación:

Este script solicita al usuario que ingrese un nombre para el archivo. A continuación, se genera una nueva variable para guardar el nombre asignado por el usuario más la extensión. Luego, con el comando `touch` más la variable `$nombreCompleto`, se crea el archivo. Finalmente, se muestra un mensaje con `echo` informando que el archivo se creó correctamente.

### Funcionamiento:

```
Ingrese un nombre para el archivo
archivoPrueba
Archivo archivoPrueba_archivo_ejercicio2.txt creado correctamente
```

## Ejercicio 3

### Consigna:

Calcular días transcurridos entre dos fechas

```
#!/bin/bash

fechaD_1=0
fechaM_1=0
fechaA_1=0

fechaD_2=0
fechaM_2=0
fechaA_2=0

diasFecha1=0
diasFecha2=0
diasDiferencia=0

echo "Ingrese la primera fecha (Dias - Mes - Año)"
read fechaD_1 fechaM_1 fechaA_1

echo "Ingrese la segunda fecha (Dias - Mes - Año)"
read fechaD_2 fechaM_2 fechaA_2

diasFecha1=$(( fechaA_1 * 365 + fechaM_1 * 30 + fechaD_1 ))
diasFecha2=$(( fechaA_2 * 365 + fechaM_2 * 30 + fechaD_2 ))

if [[ $diasFecha1 -gt $diasFecha2 ]]; then
diasDiferencia=$((diasFecha1 - diasFecha2))
else
diasDiferencia=$((diasFecha2 - diasFecha1))
fi

echo "La diferencia de dias es $diasDiferencia"
```

## Explicación:

Este Script lo que hace es solicitarle al usuario que ingrese (Dia Mes Año) de dos fechas distintas . Luego el año se multiplica por 365 (días totales de un año aproximadamente), los meses no es exacto pero puse que los multiplique por 30 y se suman los días. Finalmente se guarda en dos variables y se compara con el operador mayor que (-gt) para que de diferencia positiva.

## Funcionamiento:

```
Ingrese la primera fecha (Dias - Mes - Año)
1 1 2023
Ingrese la segunda fecha (Dias - Mes - Año)
1 1 2024
La diferencia de dias es 365
```

## Ejercicio 4

### consigna:

Contar cantidad de letras en una palabra

```
#!/bin/bash

echo "Ingrese una palabra"
read palabra

cantidadPalabras=${#palabra}

echo "La cantidad de letras en la palabra es: $cantidadPalabras"
```

### Explicación:

Este Script lo que hace es solicitarle al usuario que ingrese una palabra para luego a través del comando `${#}` calcula la longitud de la palabra y guardarla en otra variable. Finalizando con un echo que se le muestra al usuario la cantidad de letras que la componen.

### Funcionamiento:

```
Ingrese una palabra
hola
La cantidad de letras en la palabra es: 4
```

## Ejercicio 5

## consigna

Definir si un número es o no primo

```
#!/bin/bash

echo "Ingrese un numero"
read numero

if [[ $numero -lt 2 ]]; then
echo "El numero no es primo"
exit 0
else
for ((i=2; i <=numero/2; i++ )); do
    if [[ $(numero % i) -eq 0 ]]; then
        echo "El numero no es primo"
        exit 0
    fi
done
fi

echo "El numero es primo"
```

## Explicación:

Este Script solicita un numero al usuario. Luego verifica si ese numero es un numero es menor a 2, si es así le devolverá una respuesta diciendo que no es un numero primo, caso contrario se utiliza un bucle `for` para comprobar si ese numero tiene divisores entre 2 y su mitad.

## Funcionamiento:

Caso 1

```
Ingrese un numero
3
El numero es primo
```

## Caso 2

```
Ingrese un numero
8
El numero no es primo
```

## Ejercicio 6

### consigna:

Definir si un número es par o impar

```
#!/bin/bash

numero=0

echo "Ingrese un numero"
read numero

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

### Explicación:

Este Script Solicita al usuario que ingrese un numero. Luego se verifica si ese numero es par o no a través de una condicional, utilizando el operador `%`. Si en caso de que el resultado de esa operación es igual `(-eq)` a 0, entonces es par, caso contrario será impar.

### Funcionamiento:

```
Ingrese un numero
6
El numero 6 es par
```

## Ejercicio 7

### consigna:

Convertir una frase de mayúsculas a minúsculas y verificar si esta ingresado todo en minúsculas.

```
#!/bin/bash

echo "Ingrese una frase"
read frase

verificar_frase=${frase,,}

if [[ "$frase" == "$verificar_frase" ]]; then
echo "Esta frase ya esta en minuscula"
else
echo "$verificar_frase"
fi
```

### Explicación:

Este Script solicita al usuario que ingrese una frase. Luego se crea una nueva variable `$verificar_frase` en la que se almacena la frase que el usuario ingreso, pero en minúscula usando el operador `{,,}`. No importa si la frase ya esta en minúscula. Después se compara con la frase original . Si la frase esta en minúscula, se devuelve una respuesta diciendo que la frase ya se encuentra en minúscula, caso contrario se devuelve la frase en minúscula.

### Funcionamiento:

Caso 1:

```
Ingrese una frase
hola
Esta frase ya esta en minuscula
```

Caso 2:

```
Ingrese una frase
HOLA
hola
```

## Ejercicio 8

### consigna:

Informar entre dos numero cuál es el mayor y cuál es el menor.

```
#!/bin/bash

echo "Ingrese el primer numero"
read num1

echo "Ingrese el segundo numero"
read num2

if [[ $num1 -gt $num2 ]]; then
    echo "El numero $num1 es mayor"
elif [[ $num2 -gt $num1 ]]; then
    echo "El numero $num2 es mayor"
else
    echo "Ambos numeros son iguales"
fi
```

### Explicación:

Este script solicita al usuario que ingrese dos números y guardarlos en una variable. Luego con una condición, compara ambos números usando el



operador `-gt` (Mayor que). En caso de que ninguna de las dos comparaciones sea verdadera, devuelve que son iguales.

## Funcionamiento:

```
Ingrese el primer numero
8
Ingrese el segundo numero
4
El numero 8 es mayor
```

## Ejercicio 9

### consigna:

Informar el resultado de la suma, resta, división, multiplicación y potencia entre dos números.

```
#!/bin/bash

echo "ingresar el primer numero"
read num1

echo "Ingrear el segundo numero"
read num2

suma=$((num1+ num2))
resta=$((num1 - num2))
multiplicacion=$((num1 * num2))
division=$((num1 / num2))
potencia=$((num1 ** num2))

echo "Suma de ambos numeros: $suma"
echo "Resta de ambos numeros: $resta"
echo "Dividir de ambos numeros: $division"
echo "Multiplicacion de ambos numeros: $multiplicacion"
echo "Potencia de ambos numeros: $potencia"
```

## Explicación:

Este script solicita al usuario que ingrese dos números y guardarlos en variables. Luego, se crean variables con los nombres que indican el operador que se utilizó en dicha operación, almacenando el resultado. Finalizando con varias respuestas indicando los resultados con diferentes operaciones que se hicieron con ambos números.

## Funcionamiento:

```
ingresar el primer numero
10
Ingrear el segundo numero
5
Suma de ambos numeros: 15
Resta de ambos numeros: 5
Dividir de ambos numeros: 2
Multiplicacion de ambos numeros: 50
Potnecia de ambos numeros: 100000
```

## Ejercicio 10

### consigna:

Calcular el promedio entre 5 números.

```
#!/bin/bash

suma=0
numero=0
promedio=0
for i in {1..5}
do
echo "Ingrese un numero"
read numero
suma=$((suma + numero))
done

promedio=$((suma / 5))

echo "El promedio de los 5 numeros es $promedio"
```

### Explicación

En este script se genera un bucle `for` en el que itera del 1 al 5, solicitando al usuario que ingrese un numero. Ese numero se guarda en la variable y se suma con un acumulador llamado suma. Una vez finalizado el bucle, se divide el acumulador (suma) por la cantidad de números (5) y se guarda el valor en una variable. Finalizando con una respuesta al usuario mostrando el promedio.

## Funcionamiento

```
Ingrese un numero
6
Ingrese un numero
7
Ingrese un numero
6
Ingrese un numero
6
Ingrese un numero
9
El promedio de los 5 numeros es 6
```

## Ejercicio 11

### consigna:

Indicar cuál es la palabra de mayor longitud entre 5 palabras.

### Importante

En este ejercicio quería hacerlo con `arrays` para practicar como funcionan en `bash`, pero también lo hice sin utilizarlos por si no se acepta.

### Primer código

```
#!/bin/bash

for i in {1..5}
do
echo "Ingrese la palabra $i"
read palabras
listaPalabras+=($palabras)
done

mayorLongitud=0

for palabra in ${listaPalabras[@]};
do
longitud=${#palabra}

    if [[ $longitud -gt $mayorLongitud ]]; then
        mayorLongitud=$longitud
        mayorPalabra=$palabra
    fi
done

echo "La palabra de mayor longitud es: $mayorPalabra (${#mayorPalabra})"
```

**Explicación:** Este script lo que hace es ejecutar un bucle `for` que itere del 1 al 5 , dentro del bucle, se le solicita al usuario que ingrese una palabra para luego guardar esa palabra en un array. Después, se genera otro bucle que recorra todos los elementos del array `${listaPalabras[@]}` . Dentro de ese bucle, se guarda en una variable la palabra leída pero la cantidad de letras que lo componen, seguido de eso se compara esa variable `$longitud` si es mayor que (`-gt`) `$mayorLongitud` , si se cumple la condición la última variable mencionada toma el valor de la variable `$longitud` . Finalizando con un mensaje al usuario diciendo cual es la palabra con mayor longitud.

**Funcionamiento:**

```

agustin@agustin-VirtualBox:~/Escritorio/tp-bash$ ./ejercicio11.sh
Ingrese la palabra 1
a
Ingrese la palabra 2
a
Ingrese la palabra 3
hola
Ingrese la palabra 4
a
Ingrese la palabra 5
a
La palabra de mayor longitud es: hola (4)
agustin@agustin-VirtualBox:~/Escritorio/tp-bash$

```

## Segundo código

```

#!/bin/bash

echo "Ingrese 5 palabras"
read palabra1 palabra2 palabra3 palabra4 palabra5

mayorLongitud=$palabra1

if [[ ${#palabra2} -gt ${#mayorLongitud} ]]; then
    mayorLongitud=$palabra2
fi
if [[ ${#palabra3} -gt ${#mayorLongitud} ]]; then
    mayorLongitud=$palabra3
fi
if [[ ${#palabra4} -gt ${#mayorLongitud} ]]; then
    mayorLongitud=$palabra4
fi
if [[ ${#palabra5} -gt ${#mayorLongitud} ]]; then
    mayorLongitud=$palabra5
fi

echo "La palabra con mas longitud es $mayorLongitud ${#mayorLongitud}"

```

## Explicación:

En este script se le solicita al usuario que agregue las 5 palabras seguidas. Luego, se crea otra variable `$mayorLongitud` que toma el valor de una de las palabras ingresadas. Después pasa por una serie de condicionales en donde

se lo compara con las demás palabras ingresadas. Si una de las palabras tiene mayor longitud, la variable `$mayorLongitud` guarda la nueva palabra. Finalizando con una respuesta al usuario indicando la palabra con mayor longitud.

## Funcionamiento:

```
Ingrese 5 palabras
hola buenas tardes como estas
La palabra con mas longitud es buenas 6
```

## Ejercicio 12

### consigna:

Según una tabla de puntuación (I/B/MB/E), mostrar calificación según nota ingresada.

```
#!/bin/bash

echo "Ingrese su nota"
read nota

if [[ $nota -ge 0 && $nota -le 10 ]]; then
    if [[ $nota -ge 9 ]]; then
        echo "Su nota: $nota es excelente (E)"
    elif [[ $nota -ge 7 ]]; then
        echo "Su nota: $nota es Muy bueno (MB) "
    elif [[ $nota -eq 6 ]]; then
        echo "Su nota: $nota es Bueno (B)"
    else
        echo "Su nota: $nota es insuficiente (I)"
    fi
else
    echo "Nota incorrecta, Por favor vuelve a ingresarla"
fi
```

### Explicación:

Este Script solicita al usuario que ingrese su nota. Luego antes de mostrarle su calificación verifica si el usuario ingreso un numero mayor que cero y menor o igual o diez, Si cumple con ese requisito se crea otras condiciones.

## Funcionamiento:

Caso 1:

```
Ingrese su nota
8
Su nota: 8 es Muy bueno (MB)
```

Caso 2:

```
Ingrese su nota
11
Nota incorrecta, Por favor vuelve a ingresarla
```

## Ejercicio 13

### consigna:

Ingresar usuario y contraseña (solo letras), el usuario debe aceptarse sin importar mayúsculas o minúsculas. La contraseña debe ser exacta.

```
#!/bin/bash

user="user"
pass="password"

echo "Ingrese el nombre de usuario"
read userIngresado

echo "Ingrese la contraseña"
read passIngresado

if [[ ${userIngresado,,} == ${user,,} && ${passIngresado,,} == ${pass,,} ]]; then
echo "Se ingreso correctamente, Bienvenido $user"
else
echo "Credenciales incorrectas"
fi
```

### Explicación:



Este Script solicita al usuario que ingrese su usuario y su contraseña. Luego, se verifica en una condición en la cual ambas variables se comparan en minúscula utilizando el operador {,,} para ignorar mayúsculas como minúsculas. Si cumple con ambas condiciones, usando && (Y), Permitirá el acceso.

## Funcionamiento:

### Caso 1

```
Ingrese el nombre de usuario
USER
Ingrese la contraseña
PASSword
Se ingreso correctamente, Bienvenido user
```

### Caso 2:

```
Ingrese el nombre de usuario
user
Ingrese la contraseña
1234
Credenciales incorrectas, vuelva a intentarlo
```