

1. Escribe un script que pida al usuario dos valores y muestre un mensaje indicando qué valor es el mayor de ambos, o bien que son iguales.

```
#!/bin/bash
read -p "Introduce el primer número: " numero1
read -p "Introduce el segundo número: " numero2

if [ "$numero1" -gt "$numero2" ]; then
    echo "El número $numero1 es mayor que $numero2."
else
    if [ "$numero1" -eq "$numero2" ]; then
        echo "El número $numero1 es igual a $numero2."
    else
        echo "El número $numero1 es menor que $numero2."
    fi
fi
```



```
#!/bin/bash

read -p "Introduce el primer numero: " numero1
read -p "Introduce el segundo numero: " numero 2

if [ "$numero1" -gt "$numero2" ]; then
    echo "El numero $numero1 es mayor que $numero2"
else if [ "$numero1" -eq "$numero2" ]; then
    echo "El numero $numero1 es igual a $numero2"
else |
    echo "El numero $numero1 es menor que $numero2"
fi
```

2. Escribe un script que pida al usuario un valor mayor que 0 y muestre un mensaje indicando si es par o impar. Si se introduce un valor menor o igual a 0, debe mostrar un mensaje y volver a pedirlo.

```
#!/bin/bash
while true; do
    read -p "Introduce un número mayor que 0: " numero

    if [ "$numero" -le 0 ]; then
        echo "Número no valido, introduce un número mayor que 0."
        continue
    fi

    if [ $((numero%2)) -eq 0 ]; then
        echo "El numero $numero es par."
    Else
        Echo "El numero $numero es impar."
    Fi
Done
```

```

AP8_2.sh (~/) - gedit
Abrir ▾
#!/bin/bash

while true, do
    read -p "Introduce un numero mayor que 0: " numero

    if [ "$numero" -le 0 ]; then
        echo "Numero no valido, introduce un numero mayor que 0"
        continue
    fi
    if [ $((numero%2)) -eq 0 ];then
        echo "El numero $numero es par"
    else
        echo "El numero $numero es impar"
    fi
done|

```

3. Escribe un script que pida al usuario una nota, un valor entre 0 y 10, y muestre un mensaje con la calificación equivalente: insuficiente, suficiente, bien, notable o sobresaliente. Si se introduce un valor menor que 0 o mayor que 10, deberá mostrar un mensaje y volver a pedir la nota.

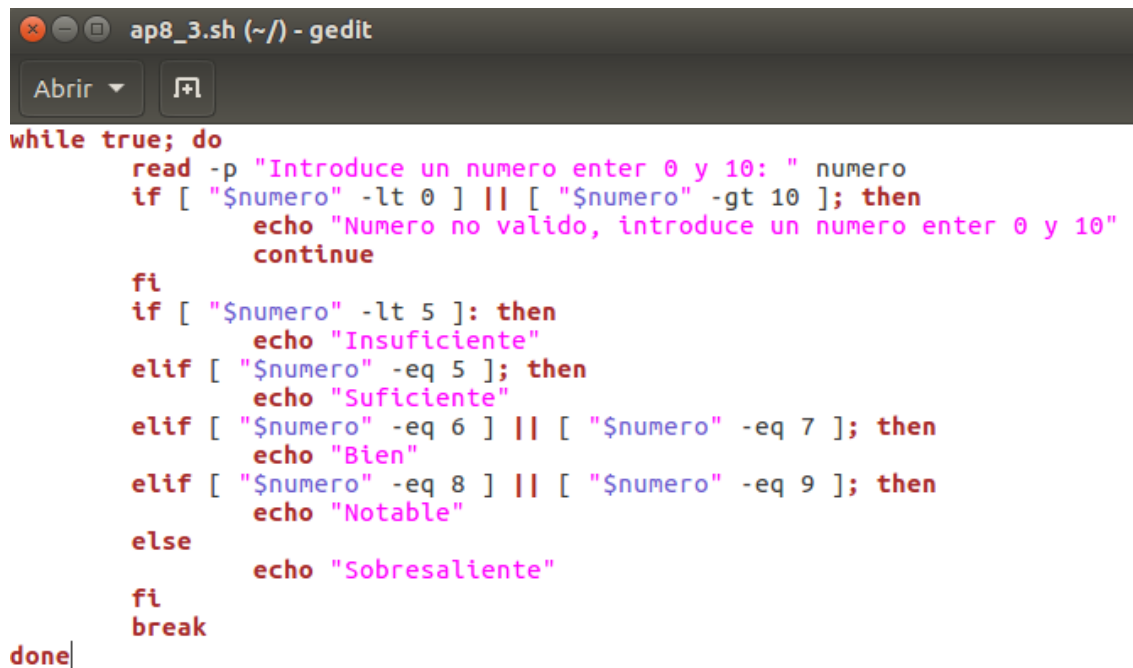
```

#!/bin/bash
While true; do
    read -p "Introduce un número entre 0 y 10: " numero

    if [ "$numero" -lt 0 ] || [ "$numero" -gt 10 ];then
        echo "Numero no valido, introduce un número entre 0 y 10."
        continue
    fi

    if [ "$numero" -lt 5 ]; then
        echo "Insuficiente"
    else if [ "$numero" -eq 5 ]; then
        echo "Suficiente"
    else if [ "$numero" -eq 6 ] || [ "$numero" -eq 7 ]; then
        echo "Bien"
    else if [ "$numero" -eq 8 ] || [ "$numero" -eq 9 ]; then
        echo "Notable"
    else
        echo "Sobresaliente"
    fi
    break
done

```



```

while true; do
    read -p "Introduce un numero enter 0 y 10: " numero
    if [ "$numero" -lt 0 ] || [ "$numero" -gt 10 ]; then
        echo "Numero no valido, introduce un numero enter 0 y 10"
        continue
    fi
    if [ "$numero" -lt 5 ]; then
        echo "Insuficiente"
    elif [ "$numero" -eq 5 ]; then
        echo "Suficiente"
    elif [ "$numero" -eq 6 ] || [ "$numero" -eq 7 ]; then
        echo "Bien"
    elif [ "$numero" -eq 8 ] || [ "$numero" -eq 9 ]; then
        echo "Notable"
    else
        echo "Sobresaliente"
    fi
    break
done

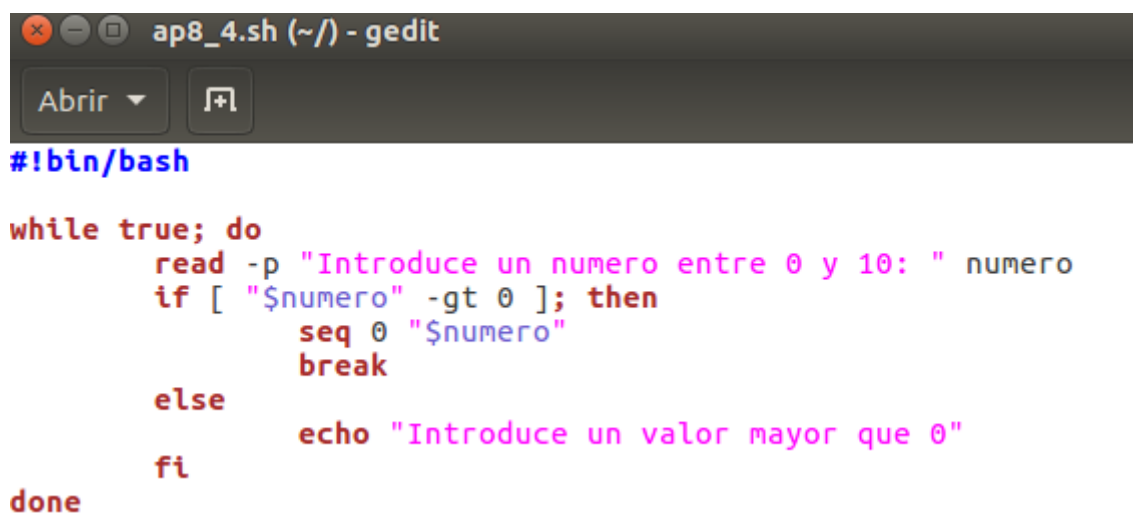
```

4. Escribe un script que pida al usuario un valor mayor que 0 y muestre el listado de números que van desde 0 hasta este valor. Si se introduce un valor menor o igual que 0, debe mostrar un mensaje y volver a pedirlo.

```

#!/bin/bash
While true; do
    read -p "Introduce un número entre 0 y 10: " numero
    if [ "$numero" -gt 0 ]; then
        seq 0 "$numero"
        break
    else
        echo "Introduce un valor mayor que 0"
    fi
done

```



```

#!/bin/bash

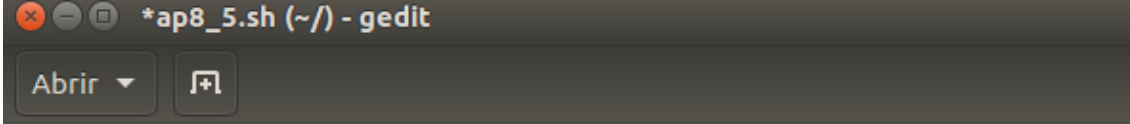
while true; do
    read -p "Introduce un numero entre 0 y 10: " numero
    if [ "$numero" -gt 0 ]; then
        seq 0 "$numero"
        break
    else
        echo "Introduce un valor mayor que 0"
    fi
done

```

5. Escribe un script que pida al usuario valores, hasta que introduzca el 0. Cuando el usuario introduzca el 0, se mostrará un mensaje que muestre la suma total de valores introducidos, así como el valor medio.

```
#!/bin/bash
Suma=0
Contador=0
While true; do
    Read -p "Introduce un numero (0 para terminar): " numero

    if [ "$numero" -eq 0 ]; then
        break
    fi
    suma=$((suma + numero))
    contador=$((contador + 1))
done
if [ "$contador" -eq 0 ]; then
    echo "No se introdujeron valores válidos"
else
    media=$((suma / contador))
    echo "Suma total: $suma"
    echo "Valor medio: $media"
fi
```



```
#!/bin/bash

suma=0
contador=0

while true; do
    read -p "Introduce un numero (0 para terminar): " numero

    if [ "$numero" -eq 0 ]; then
        break
    fi
    suma=$((suma+numero))
    contador=$((contador+1))
done

if [ "$contador" -eq 0 ]; then
    echo "No son valores validos"
else
    media=$((suma/contador))
    echo "Suma total: $suma"
    echo "Valor medio: $media"
fi
```

6. Escribe un script que pida al usuario una cantidad de litros de agua consumidos y muestre el coste total, atendiendo a las siguientes indicaciones:

Los primeros 50 litros cuestan 20 euros.

Los restantes hasta 200 litros cuestan a 20 céntimos el litro.

Los restantes cuestan a 10 céntimos el litro.

El script debe validar que el valor introducido sea mayor que 0 y si no fuera así, mostrará un mensaje y pedirá que se introduzca de nuevo el valor.

```
#!/bin/bash
while true; do
    Read -p "Introduce la cantidad de litros que necesitas: " litros

    if [ "$litros" -le 0 ]; then
        echo "Error: el valor debe ser mayor que 0."
        continue
    fi
    break
done

If [ "$litros" -le 50 ]; then
    Coste=20
Else if [ "$litros" -le 200 ]; then
    extra=$((litros - 50))
    coste_centimos=$((2000 + extra * 20))
else
    extra1=$((200 - 50))
    extra2=$((litros - 200))
    coste_centimos=$((2000 + extra1 * 20 + extra2 * 10))
fi
euros=$((coste_centimos / 100))
echo "Coste total: $euros euros"
```



```
#!/bin/bash

while true, do

    read -p "Introduce la cantidad de litros que necesitas: " litros

    if [ "$litros" -le 0 ]; then
        echo "Error, el valor debe ser mayor que 0"
        continue
    fi
    break
done

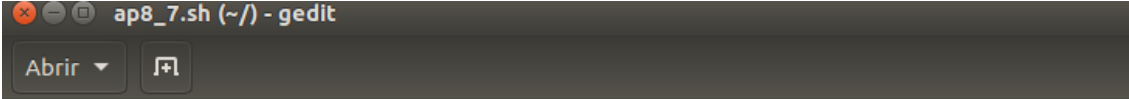
if [ "$litros" -le 50 ]; then
    coste=20
else if [ "$litros" -le 200 ]; then
    extra=$((litros-50))
    coste_centimos=$((2000+extra*20))
else
    extra1=$((200-50))
    extra2=$((litros-200))
    coste_centimos=$((2000+extra1*20+extra2*10))
fi
euros=$((coste_centimos/100))
echo "Coste total: $euros"
```

7. Escribe un script que pida al usuario un valor del día numérico del mes (de 1 a 30) y muestre por pantalla qué día de la semana es, asumiendo que el día 1 es lunes. Para cualquier valor introducido fuera del rango, se mostrará un mensaje y se volverá a pedir.

```
dias_semana=("lunes" "martes" "miércoles" "jueves" "viernes" "sábado" "domingo")
```

```
while true; do
    read -p "Introduce un día numérico del mes (1 a 30): " dia

    if [[ "$dia" -ge 1 && "$dia" -le 30 ]]; then
        dia_semana_index=$(( (dia - 1) % 7 ))
        echo "El día $dia es ${dias_semana[dia_semana_index]}."
        Break
    Else
        echo "Por favor, introduce un día entre 1 y 30."
    Fi
Done´
```



```
ap8_7.sh (~/) - gedit
Abrir ▾ [F1]
#!bin/bash

dias_semana=("lunes" "martes" "miercoles" "jueves" "viernes" "sabado" "domingo")

while true; do
    read -p "Introduce un dia numero del mes (1 a 30): " dia
    if [[ "$dia" -ge 1 && "$dia" -le 30 ]];then
        dia_semana_indice=$(( (dia-1) % 7 ))
        echo "El día $dia es ${dias_semana[dia_semana_indice]}"
        break
    else
        echo "Introduce un dia entre 1 y 30"
    fi
done
```