

1. Pedir un número por pantalla y crear tantas carpetas como el usuario quiera, llamadas

Datos1

Datos2

Datos3

...

```
GNU nano 7.2 scrip
#!/bin/bash

echo dime un numero
read num

echo dime el nombre de la carpeta
read carpeta

for ((i=1;i<num;i++))
do
    mkdir $carpeta$i
done
```

2. Pedir nombre del usuario por pantalla
 - Que el nombre se copie en un fichero de texto llamado nombre.txt
 - Si el fichero de texto existe, que elimine su contenido antes de añadir al usuario

```
GNU nano 7.2
#!/bin/bash

echo dime el nombre de usuario
read nombre

echo $nombre > nombre.txt
```

3. Pedir nombre del usuario por pantalla
 - Que el nombre se añada a un fichero de texto
 - Mostrar el contenido del fichero

```
GNU nano 7.2 scr
#!/bin/bash

echo dime el nombre de usuario
read nombre
clear
echo $nombre >> nombre.txt
cat nombre.txt
```

4. Pedir un número por pantalla
 - Añadir desde el número 1 hasta el número introducido por pantalla a un fichero de texto

```
GNU nano 7.2
#!/bin/bash

echo dime el numero
read numero

for ((i=1;i<=numero;i++))
do
    echo $i >> fichero.txt
done
```

5. Pedir el nombre de un fichero, comprobar si el fichero existe o no y mostrar el resultado por pantalla.

```
GNU nano 7.2 script23.sh *
#!/bin/bash

echo dime el nombre de un fichero
read nombre

[ -f $nombre ] && echo existe && cat $nombre || echo no existe
```

6. Pedir el nombre de un directorio por pantalla. Si existe sacar un mensaje por pantalla, y si no existe, crearlo.

```
GNU nano 7.2 script23.sh *
#!/bin/bash

echo dime el nombre de un directorio
read nombre

[ -d $nombre ] && echo existe || mkdir $nombre
```

7. Recorrer el contenido del directorio actual mostrando por pantalla solo los directorios.

```
#!/bin/bash

for i in *
do
    [ -d $i ] && echo $i
done
```

8. Recorrer el contenido del directorio actual contando cuántos directorios hay y cuántos archivos. Mostrar el resultado final por pantalla.

```
GNU nano 7.2 script23.sh *
#!/bin/bash

fichero=0
directorio=0

for i in *
do
    [ -d $i ] && let directorio++
    [ -f $i ] && let fichero++
done

echo hay $fichero ficheros y $directorio directorios
```

9. Imprimir el abecedario completo en una sola línea

```
GNU nano 7.2
#!/bin/bash

for i in {a..z}
do
    echo -n $i
done
```

10. Pedir un nombre de usuario por pantalla. Si el usuario existe mostrar información sobre él, y si no existe que no aparezca nada.

```
andres@andres-Vir
GNU nano 7.2
#!/bin/bash

echo nombre
read nombre

id $nombre 2>/dev/null
```

11. Almacenar un valor en una variable. Pedir un número por pantalla hasta que el usuario acierte el número.

```
GNU nano 7.2
#!/bin/bash

num1=55
num=0

while [ $num -ne $num1 ]
do
    echo dime el numero
    read num
done
echo acertado!!!!!!
```

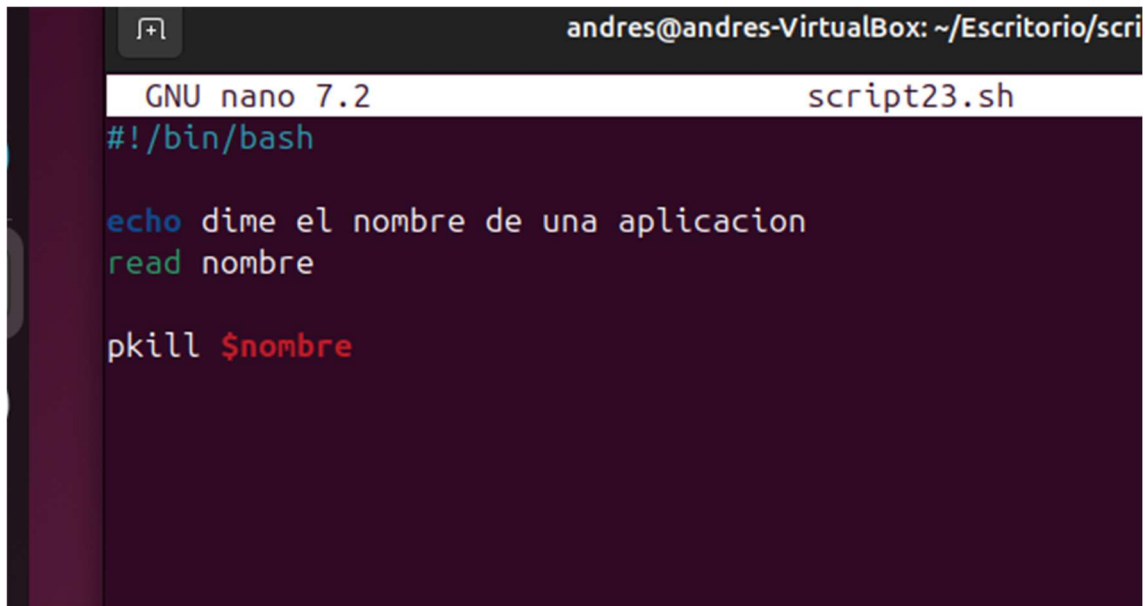
12. Solicitar el nombre de una aplicación activa por pantalla y mostrar los PID de todos sus procesos.

```
andres@andres-VirtualBox: ~/Escritor
GNU nano 7.2 script23.sh
#!/bin/bash

echo dime el nombre de una aplicacion
read nombre

pidof $nombre
```

13. Repetir el ejercicio anterior, pero en lugar de mostrar los procesos, cerrarlos todos.



```
andres@andres-VirtualBox: ~/Escritorio/scri
GNU nano 7.2 script23.sh
#!/bin/bash

echo dime el nombre de una aplicacion
read nombre

pkill $nombre
```

14. Repetir el ejercicio 12 pero en lugar de mostrar los PID por pantalla, almacenarlos todos en un fichero de texto.



```
andres@andres-VirtualBox: ~/Escritorio/script
GNU nano 7.2 script23.sh
#!/bin/bash

echo dime el nombre de una aplicacion
read nombre

pidof $nombre > PIDS.txt
```

15. Solicitar un número por pantalla. Si este número es un 1 mostrar el primer día de la semana en letra, si es un 2 el segundo y así consecutivamente.

```
GNU nano 7.2 script23.s
#!/bin/bash

echo dime un numero
read numero

case $numero in
    1)
        echo lunes;;
    2)
        echo martes;;
    3)
        echo miercoles;;
    4)
        echo jueves;;
    5)
        echo viernes;;
    6)
        echo sabado;;
    7)
        echo domingo;;
    *)
        echo Numero no valido;;
esac
```