

1. Realiza el script `ejer09.sh` al que se le puedan pasar varios ficheros regulares como argumento, y diga para cada uno si es ejecutable o no. Ejemplo:

```
$ ejer09.sh fich1 fich2 fich3
fich1 es ejecutable
fich2 es ejecutable
fich3 no es ejecutable

#!/bin/bash

test -f $1;

if [ $? -eq 0 ];
then
    test -x $1;
    if [ $? -eq 0 ];
    then
        echo "$1 es ejecutable"
    else
        echo "$1 no es ejecutable"
    fi
fi

test -f $2;

if [ $? -eq 0 ];
then
    test -x $2;
    if [ $? -eq 0 ];
    then
        echo "$2 es ejecutable"
    else
        echo "$2 no es ejecutable"
    fi
fi

test -f $3

if [ $? -eq 0 ];
then
    test -x $3;
    if [ $? -eq 0 ];
    then
        echo "$3 es ejecutable"
    else
        echo "$3 no es ejecutable"
    fi
fi
```

2. Crea un script llamado `ejer10.sh` al que se le pase como argumento una de las siguientes bebidas: `cafe`, `te` o `chocolate` (aunque se admite el uso de tildes, es más conveniente no usarlas). Se mostrarán los siguientes mensajes:

- Si se escoge `cafe`: Su café estará listo en un momento.
- Si se escoge `te`: En seguida le traigo su té.
- Si se escoge `chocolate`: Buena elección, ya verá como le encanta.
- Si se escoge otra cosa: Lo siento, pero no servimos esta bebida.

```
#!/bin/bash

case $1 in
    cafe)
        echo "Su café estará listo en un momento."
        ;;
    te)
        echo "En seguida le traigo su té."
        ;;
    chocolate)
        echo "Buena elección, ya verá como le encanta."
        ;;
    *)
        echo "Lo siento, pero no servimos esta bebida."
        ;;
esac
```

3. Modifica el script anterior (llamándolo `ejer11.sh`) para que, en caso de elegir `té`, pregunte si quiere el té con leche. Según se responda `sí` o `no`, mostrará el mensaje apropiado.

```
#!/bin/bash

case $1 in
    cafe)
        echo "Su café estará listo en un momento."
        ;;
    te)
        echo "En seguida le traigo su té. Pero..."
        read -p "¿Lo quiere con leche?[SI/NO]: " resp
        if [ $resp == "si" -o $resp == "SI" ];
        then
```

```
        echo "Añadiendo leche..."
    else
        echo "Té sin leche."
    fi
    ;;
chocolate)
    echo "Buena elección, ya verá como le encanta."
    ;;
*)
    echo "Lo siento, pero no servimos esta bebida."
    ;;
esac
```

4. Crea un script llamado `ejer12.sh` que solicite la longitud del lado de un cuadrado y calcule su área.

```
#!/bin/bash
read -p "
INDIQUE LA LONGITUD DEL CUADRADO:
" respc
echo "EL ÁREA DEL CUADRADO ES:"
expr $respc \* $respc
```

5. Crea un script llamado `ejer13.sh` que solicite la base y la altura de un triángulo y calcule su área.

```
#!/bin/bash
read -p "
INDIQUE LA ALTURA DEL TRIÁNGULO:
" altt
read -p "
INDIQUE LA BASE DEL TRIÁNGULO:
" bast
echo "EL ÁREA DEL TRIÁNGULO ES:"
expr $bast \* $altt / 2
```

6. Crea un script llamado `ejer14.sh` que permita visualizar el contenido de un fichero que se le pase como argumento. Debe comprobarse previamente si se tiene permiso de lectura sobre el mismo (si no tiene permiso, se indicará y terminará el script).

```
#!/bin/bash
test -f $1;
if [ $? -eq 0 ];
then
    test -r $1;
    if [ $? -eq 0 ];
    then
        echo "$1 tiene permisos de lectura"
    elif [ $? -eq 1 ];
    then
        echo "$1 no tiene permisos de lectura"
    fi
fi
```

7. Realiza el mismo ejercicio (llamándolo `ejer15.sh`), pero en este caso solicitando que se introduzca por el teclado el nombre del fichero. Se deberá comprobar que el nombre introducido corresponde a un fichero.

```
#!/bin/bash
read -p "INTRODUCE EL NOMBRE DEL FICHERO QUE SE DESEA IDENTIFICAR: " fic
test -f $fic;
if [ $? -eq 0 ];
then
    echo "El fichero $fic existe"
    test -r $fic;
    if [ $? -eq 0 ];
    then
        echo "$fic tiene permisos de lectura"
    elif [ $? -eq 1 ];
    then
        echo "$fic no tiene permisos de lectura"
    fi
else
    echo "El fichero $fic no existe"
fi
```

8. Crea un script llamado `ejer16.sh` que solicite un nombre de fichero y una palabra, y que, tras comprobar si el fichero existe y tiene permiso de lectura, diga si la palabra está en el fichero o no.

```
#!/bin/bash
read -p "NOMBRE DEL FICHERO: " fic
read -p "PALABRA SELECCIONADA: " pal
test -f $fic;
if [ $? -eq 0 ]; then
    test -r $fic;
    if [ $? -eq 0 ]; then
        echo "El fichero $fic tiene permisos de lectura"
        grep -iw $pal $fic
    elif [ $? -eq 1 ]; then
        echo "El fichero $fic no tiene permisos de lectura"
    fi
fi
```

9. Diseña un script llamado `ejer17.sh` en el que, dado un argumento (un solo carácter), nos diga si es un número, una letra u otra cosa. Ejemplos de utilización:

```
$ ejer17.sh 4      $ ejer17.sh Z      $ ejer17.sh +
4 es un número    Z es una letra    + no es número ni letra
```

(Nota: uso de case)

```
#!/bin/bash
Case $1 in
    [a-zA-Z])echo "Es una letra"
    ;;
    [0-9])echo "Es un número"
    ;;
    *)echo "Es otra cosa"
    ;;
esac
```

10. Crea un script llamado `ejer18.sh` al que se le pase un número como argumento. Usando el bucle `while` deberá mostrar su tabla de multiplicar. La salida deberá ser algo parecido a:

```
alumno@server:~/scripts$ ejer18.sh 5
La tabla de multiplicar del número 5 es:
-----
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50
```

```
#!/bin/bash

number=10

while [ $number -gt -1 ]; do

    echo "$1 * $number = "

    expr $1 \* $number

    number=$(( $number - 1 ))

done
```

11. Repite el ejercicio anterior llamando al script `ejer19.sh` pero usando el bucle `until`.

```
#!/bin/bash

i=10
until [ $i -lt 0 ]
do
    echo "$1 * $i = "

    expr $1 \* $i

    ((i--))
done
```

12. Escribe un script llamado `ejer20.sh` que reciba como argumento el *login* de un usuario. Deberá comprobarse si existe en el sistema y, en caso afirmativo, hacer una copia de seguridad del contenido del directorio `home` de ese usuario:
- La copia de seguridad se hará con el comando `tar` y el fichero resultante se debe dejar en el directorio `/tmp`. El nombre será: *login.tar*
 - Se supone que el directorio `home` de todos los usuarios es `/home/login`.

```
Login=$1
Ruta=/tmp/
If [ id -u $login > /dev/null 2 > &1;
Then
    Echo "El usuario $login existe, haciendo copia de seguridad"
    Tar -cf $login.tar $ruta
    Echo "Creando copia de seguridad"
    Mv $login.tar $ruta
Else
    Echo "No existe $login"
fi
```