

T9.2. TABLA Operadores condicionales

Operator	Description
! EXPRESSION	The EXPRESSION is false.
-n STRING	The length of STRING is greater than zero.
-z STRING	The length of STRING is zero (ie it is empty).
STRING1 = STRING2	STRING1 is equal to STRING2
STRING1 != STRING2	STRING1 is not equal to STRING2
INTEGER1 -eq INTEGER2	INTEGER1 is numerically equal to INTEGER2
INTEGER1 -gt INTEGER2	INTEGER1 is numerically greater than INTEGER2
INTEGER1 -lt INTEGER2	INTEGER1 is numerically less than INTEGER2
-d FILE	FILE exists and is a directory.
-e FILE	FILE exists.
-r FILE	FILE exists and the read permission is granted.
-s FILE	FILE exists and its size is greater than zero (ie. it is not empty).
-w FILE	FILE exists and the write permission is granted.
-x FILE	FILE exists and the execute permission is granted.

T9.3. Ejemplo Bucles while

Para crear un bucle basta con indicar la palabra reservada **while** seguido de la condición (al igual que con los ifs) entre corchetes **[]** tras la condición se debe finalizar con un **;** y un **do**, para cerrar el bucle se debe indicar con la palabra reservada **done**, por lo que las casuísticas a desarrollar dentro del bucle deben aparecer entre el **do** y el **done**.

Aquí un ejemplo de bucle while con un contador:

```
#!/bin/bash

CONTADOR=0

while [ $CONTADOR -lt 10 ]; do

    echo El contador es $CONTADOR

    CONTADOR=$((CONTADOR+1))

done
```

T9.4. Ejemplo Bucles for

Para crear un bucle for con indicar la palabra reservada **for** seguido de la variable que va a guardar cada iteración del bucle, seguido a su vez de un **in** y de los elementos sobre los que va a iterar, éstos pueden ser variables, número, ficheros, etc. Un **do** para empezar el bucle y un **done** para cerrarlo, por lo que las casuísticas a desarrollar dentro del bucle deben aparecer entre el do y el done.

Aquí un ejemplo de bucle for que avanza por los números del 1 al 5:

```
#!/bin/bash

for i in 1 2 3 4 5

do

    echo "Número $i"

done
```

Para indicar una secuencia de número y que se recorran de 1 en 1:

```
#!/bin/bash
for i in {1..5}
do
    echo "Número $i"
done
```

Para indicar una secuencia de números y que se recorran de 2 en 2 (en **negrita** el lugar donde se indica el avance de 2 en 2):

```
#!/bin/bash

for i in {0..10..2}

do

    echo "Número $i"

done
```

Si queremos recorrer variables o ficheros:

```
for VARIABLE in file1 file2 file3

do

    command1 on $VARIABLE
```

```
command2
```

```
commandN
```

```
done
```

Para recorrer elementos de un directorio **(MUY ÚTIL)**:

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

```
for file in /etc/*
```

```
do
```

```
    if [ "${file}" == "/etc/passwd" ]; then
```

```
        echo "Puedo hacer cosas sobre un fichero concreto"
```

```
    fi
```

```
done
```