

For

Un bucle for tiene la siguiente estructura:

```
for (inicialización; condición de salida; expresión final) {  
    //Código a repetir  
}
```

En el apartado inicialización se suele declarar una variable, aunque también se puede utilizar una ya existente, depende de las condiciones del bucle.

A la variable que hemos creado se le asigna un valor inicial (normalmente 0), que se modificará con la expresión final. El bucle se repetirá hasta que la condición de salida devuelva un valor falso.

Este tipo de bucles se utiliza principalmente para recorrer matrices, cadenas o estructuras de las que sabemos su longitud, o para **repetir un proceso del que sabemos exactamente cuantas repeticiones se tienen que producir**.

While

La estructura de un bucle while es:

```
while (condición de salida) {  
    //Código a repetir  
    //Expresión final (opcional)  
}
```

En el apartado condición de salida hay que escribir una expresión que devuelva verdadero o falso, el bucle se repetirá hasta que la condición devuelva el valor falso.

La expresión final se usa para incrementar o modificar alguna de las variables de la condición inicial, dependiendo de la configuración del bucle, será necesaria o no.

Este bucle se recorre siempre que la condición inicial devuelva verdadero, por lo que **puede llegar a no recorrerse**.

Do ... While

La estructura de un bucle do ... while es:

```
do{  
    //Código a repetir  
    //Expresión final (opcional)  
} while (condición de salida)
```

Funciona igual que el bucle while, salvo que este tipo de bucles **se recorre como mínimo una vez**, ya que se comprueba la condición de salida al final del bucle en vez de al principio como en el caso anterior.