
SENTENCIA for

Un bucle **for** se repite hasta que la condición especificada se evalúa como false. Se utiliza generalmente cuando se sabe la cantidad de iteraciones que tendrá el bucle.

Sintaxis:

```
for ([expresionInicial]; [condicion]; [expresionIncremento]) {  
    sentencias...  
}
```

Cuando un bucle **for** se ejecuta, ocurre lo siguiente:

1. La expresión de inicialización **expresionInicial**, si existe, se ejecuta. Esta expresión habitualmente inicializa uno o mas contadores del bucle, pero la sintaxis permite una expresión con cualquier grado de complejidad. Esta expresión puede también declarar variables.
2. Se evalúa la expresión **condicion**. Si el valor de **condicion** es true, se ejecuta la sentencia del bucle. Si el valor de **condicion** es false, el bucle for finaliza. Si la expresión **condicion** es omitida, la condición es asumida como verdadera.
3. Se ejecuta la sentencia. Para ejecutar múltiples sentencias, use un bloque de sentencias ({ ... }) para agruparlas.
4. Se ejecuta la expresión **expresionIncremento**, si hay una, y el control vuelve al paso 2.

```
for(let i = 0;; i++) {  
    i = 10  
    console.log(i)  
    if (i === 2){  
        console.log('el ' + i + ' es mi numero favorito')  
        continue;  
    }  
    if (i >= 6){  
        console.log('detener el loop');  
        break;  
    }  
    console.log('Numero ' + i);  
}
```

SENTENCIA do while

La sentencia do...while se repite hasta que la condición especificada que se evalúa sea false.

Sintaxis:

```
do {  
    sentencias...  
} while (condicion)
```

La sentencia se ejecuta antes de que la *condición* sea evaluada. Para ejecutar múltiples sentencias, se utiliza un bloque de sentencias ({ ... }) para agruparlas. Si *condicion* es true, la sentencia se ejecuta de nuevo. Al final de cada ejecución, la condición es comprobada. Cuando la condición es falsa, la ejecución se detiene y el control pasa a la sentencia siguiente al do...while.

Esta estructura ejecuta al menos una vez la sentencia o sentencias contenidas en do ... while. Es decir, de 1 a n veces.

```
let i = 100;  
do {  
    console.log('numero ' + i);  
    i++;  
}  
while(i < 10);
```

SENTENCIA while

Una sentencia **while** ejecuta sus sentencias mientras la condición sea evaluada como verdadera.

Sintaxis:

```
while (condicion) {  
    sentencia...  
}
```

Si la *condición* cambia a falsa, la sentencia dentro del bucle deja de ejecutarse y el control pasa a la sentencia inmediatamente después del bucle.

La *condición* se evalúa antes de que la sentencia contenida en el bucle sea ejecutada. Si la condición devuelve verdadero, la sentencia se ejecuta y la condición se comprueba de nuevo. Si la condición es evaluada como falsa, se detiene la ejecución y el control pasa a la sentencia siguiente al **while**.

Para ejecutar múltiples sentencias, use un bloque de sentencias ({ ... }) para agruparlas.

Esta estructura ejecuta 0 a n veces el bloque de sentencias.

```
let i = 0;  
while(i < 10) {  
    console.log('numero ' + i);  
    i++;  
}
```

Evite los bucles infinitos. Asegúrese de que la condición en un bucle llegue finalmente a ser falsa; de otra forma, el bucle nunca terminará. Las sentencias en el siguiente bucle while se ejecutan sin fin, porque la condición nunca llega a ser falsa.

```
while (true) {  
    console.log("Hello, world");  
}
```