



JavaScript



Módulo 5



1. Estructuras de control iterativas: `for` / `for in` / `for of`
2. Estructuras de control iterativas: `while` / `do-while`
3. Interrumpir, continuar o abandonar un bucle.



Estructura de control: for.

Recorre un bloque de código varias veces

Inicializa la variable de control del bucle.

Expresión lógica que determina si han de ejecutar mientras sea verdadera.

```
3  for (inicialización; Condición ; Incremento) {  
4      // Bloque que será ejecutado varias veces.  
5  }
```

Sentencias a ejecutar en cada iteración del bucle.

Incrementa o decrementa la variable de control del bucle.

Hagamos algunos ejemplos.



Ejercicios.

1. Crear un bucle for que cuente en forma descendente del 10 al 0.
2. Crear un bucle for que imprima las potencias de 2 del exponente 0 al 10.
3. Crear un factorial con un bucle for.



Estructura de control: for in.

La sentencia JavaScript for in recorre las propiedades de un objeto:

Variable que tomará los valores de los posibles índices de array.

Nombre del arreglos del que vamos a extraer elemento a elemento.

```
5 for (indice in arreglo) {  
6   //Bloque que será ejecutado.  
7 }
```

Sentencias a ejecutar en cada iteración del bucle.

Hagamos un ejemplo.



Ejercicios.

1. Crear un arreglo con los días de la semana e imprimir toda la semana con un for in.
2. Crear un arreglo de 5 números e imprimirlos todos multiplicados por 3 con un for in.



Estructura de control: for of.

La sentencia JavaScript for of recorre los valores de un objeto iterable, tales como Arrays, Objects, Strings, etc...

Variable que tomará los valores de cada elemento contenido en el Array.

Nombre del arreglo del que vamos a extraer elemento a elemento.

```
8  for (elemento of arreglo) {  
9      //Bloque de código que será ejecutado.  
10 }
```

Sentencias a ejecutar en cada iteración del bucle.

Hagamos un ejemplo.



Ejercicios.

1. Crear un arreglo con 5 nombres de personas diferentes y mostrarlos en la ventana del navegador usando for of.



Estructura de control: while.

La sentencia JavaScript while ejecuta un bloque de código siempre que se cumpla una condición específica.

Condición específica que provocará que termine el bucle.

```
2 while (condición) {  
3     // Bloque de código que será ejecutado.  
4 }
```

Sentencias a ejecutar en cada iteración del bucle.

Hagamos un ejemplo.



Ejercicios.

1. Crear un ciclo while que haga la suma de los números del 1 al 100.
2. Imprimir los elementos de un arreglo con el bucle while.



Estructura de control: do-while.



El bucle do while es una variante del bucle while.

Este ciclo ejecutará el bloque de código una vez, antes de verificar si la condición es verdadera, luego repetirá el ciclo mientras la condición sea verdadera.

Tip: podemos usarlo para filtrar valores numéricos.



Estructura de control: do-while.

Sentencias a ejecutar al
menos una vez antes.

```
3  do {  
4      //Bloque de código que será ejecutado.  
5  }while (condición);  
6
```

Condición específica que provocará que termine
el bucle, será evaluada después de ejecutar una
vez el bloque.



Ejercicios.

1. Crear un filtro de datos entre [1,5]
2. Crear un filtro que sólo acepte los siguientes valores:
1,2,3,7,8,9.



Interrumpir, continuar o abandonar un bucle.

Break es la instrucción que salta de un bucle.

Continue es la instrucción que salta una iteración en el bucle.

Hagamos un ejemplo de cada uno.



Ejercicios.

1. Hacer un bucle while que imprima los números del 1 al 10 excepto el 4 y el 9.