

BUCLAS REPETITIVOS

Iteración

En programación, iteración se refiere al **proceso de ejecutar repetidamente un bloque de código dentro de una estructura repetitiva, como un bucle (for, while, do...while, etc.), hasta que se cumple una condición específica o se recorren todos los elementos de una colección.**

Bucles repetitivos

(también conocidas como estructuras de control de bucle)

En JavaScript los bucles son utilizados para realizar tareas repetitivas con base en una condición.

Las condiciones devuelven true (verdadero) o false(falso) al ser evaluados. El bucle continuará ejecutándose hasta que la condición devuelva false.

While
Do While
For

BUCLE FOR



Sintaxis del bucle FOR

Los bucles For son comúnmente utilizados para contar un cierto número de iteraciones para repetir una sentencia. Puedes utilizar una sentencia **break** para salir del bucle antes que la expresión de condición devuelva false.

```
for (inicialización; condición; incremento) {  
    // Código a ejecutar en cada iteración  
}
```

```
for (inicialización; condición; incremento) {  
    // Código a ejecutar en cada iteración  
}
```

- **Inicialización:** Se ejecuta una vez al principio del bucle. Generalmente se usa para declarar e inicializar una variable de contador.
- **Condición:** Expresión que es evaluada antes de la ejecución de cada iteración. Si se omite, esta expresión es evaluada como verdadera. Si devuelve true, la sentencia del cuerpo del bucle se ejecuta. Si devuelve false, el bucle se detiene.
- **Expresión-final:** Expresión que se ejecuta luego de cada iteración. Usualmente es utilizada para incrementar un contador. Pero también puede ser utilizada para decrementar el contador

Ejemplo de bucle FOR

```
for (let i = 0; i < 5; i++) {  
  console.log(i); // Imprime los números del 0 al 4  
}
```

```
for (inicialización; condición; incremento) {  
  // Código a ejecutar en cada iteración  
}
```

BUCLE WHILE

Sintaxis bucle WHILE

El bucle while empieza por evaluar la condición. Si la condición es verdadera (devuelve true), entonces las sentencias son ejecutadas. Si la condición es falsa (devuelve false), entonces las sentencias no son ejecutadas. Luego el bucle finaliza.

```
while (condición) {  
    // Código a ejecutar mientras la condición sea verdadera  
}
```

Bucle While

```
while (condición) {  
    // Código a ejecutar mientras la condición sea verdadera  
}
```

- **Condición:** Es una expresión booleana (Boolean) que es evaluada antes de cada paso (iteración) por el bucle. Si esta condición evalúa a verdadero (true), las sentencias son ejecutadas. Cuando la condición evalúa a falso (false), la ejecución continúa con la siguiente sentencia luego del bucle while.
- **Sentencia(s):** Una sentencia es código que se ejecuta si la condición devuelve verdadero (true).

Ejemplo de bucle While

```
let i = 0;
while (i < 5) {
  console.log(i); // Imprime los números del 0 al 4
  i++; // Incrementa i en 1
}
```

BUCLE DO WHILE

Sintaxis del bucle DO WHILE

El bucle Do While garantiza que el bloque de código se ejecute al menos una vez, ya que la condición se evalúa después de la ejecución del bloque de código

```
do {  
    // Código a ejecutar  
} while (condición);
```

Bucle DO WHILE

```
do {  
    // Código a ejecutar  
} while (condición);
```

Condición: Se evalúa después de ejecutar el bloque de código. Si es true, el bucle continúa; si es false, el bucle termina.

Los bucles Do While aseguran que el código se ejecute por lo menos una vez, y luego de esta primera ejecución, si la condición dentro del while() es true, continúa con el bucle, de otra forma, se detiene.

Ejemplo del bucle DO WHILE

```
let i = 0;  
do {  
    console.log(i); // Imprime los números del 0 al 4  
    i++;  
} while (i < 5);
```



GRACIAS!

Docente: Verónica Godoy