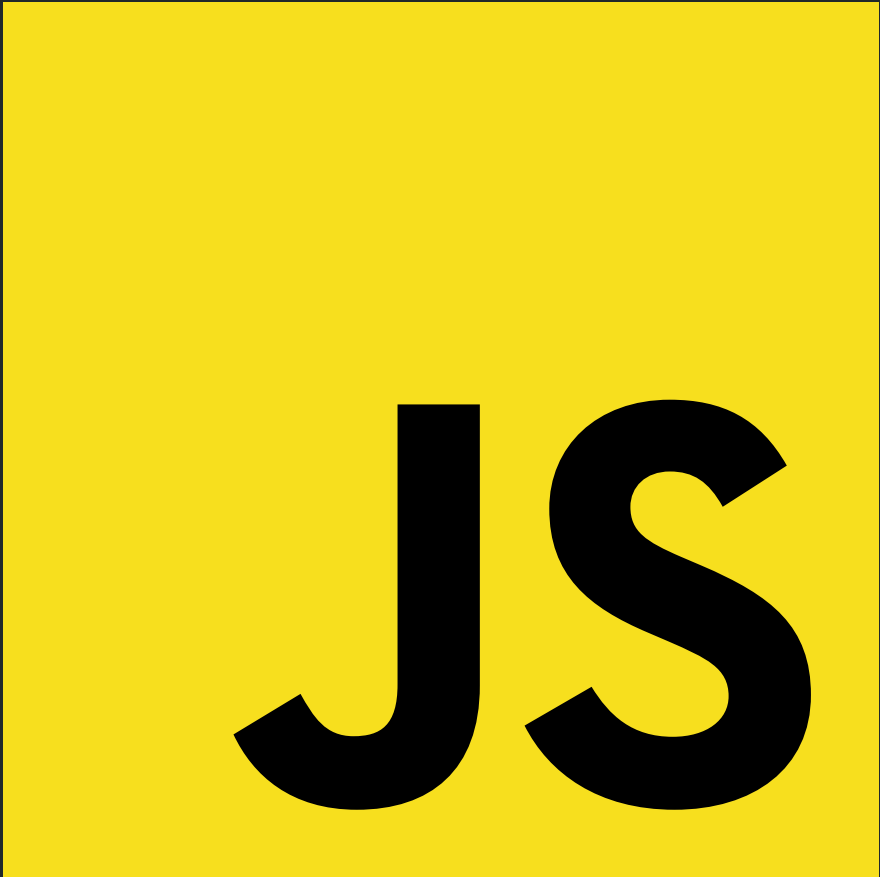




## Types Of **Loops** ...

- while
- do-while
- for
- forEach()
- for...in
- for...of



JS



The **while statement** generates a **loop** that gets executed over a particular **block of statement** (code) as long as the condition is **true**. Every time **before** executing the block of code the condition gets checked.



```
let i = 0;  
while (i < 5) {  
    console.log(i);  
    i++;  
} // 0 1 2 3 4
```

The **do-while loop** is similar to the while loop, but it executes the block of code **at least once**, even if the condition is **false**.



```
let i = 5;  
do {  
    console.log(i);  
    i++;  
}  
while (i < 5); // Output: 5
```

In case of **for loop** the declaration & initialization of looping variable, condition checking & increment or decrements of the looping variable can be done in the **same line**.



```
// Syntax...
```

```
for ([initialization]; [condition]; [final - expression]) {  
  Block of code  
}
```

```
// Example
```

```
for (let i = 0; i < 5; i++) {  
  console.log(i);  
} // 0 1 2 3 4
```

The **forEach() method** is used to iterate over the elements of an array.



```
const numArray = [1, 2, 3];  
numArray.forEach(value => {  
  console.log(value);  
}); // 1, 2, 3
```

The **for...in loop** is used to iterate over the properties of an object.



```
// Syntax
for (variableName in object) {
  Block of code
}
// Example
const obj = { a: 1, b: 2, c: 3 };
for (let prop in obj) {
  console.log(`${prop}: ${obj[prop]}`);
} // Output: a: 1 b: 2 c: 3
```

The **for...of loop** is used to iterate over the values of an iterable object, such as an **array** or a **string**.



```
// Syntax
```

```
for (variable of iterable) {  
    Block of code  
}
```

```
// Example
```

```
const arr = [1, 2, 3];  
for (let value of arr) {  
    console.log(value);  
} // 1 2 3
```