

ChatGPT - JS Basics and Operators

Loops in JavaScript with Examples

In JavaScript, loops allow you to execute a block of code repeatedly. Here are the **8 types of loops** in JavaScript:

1. For Loop

Executes a block of code repeatedly for a specified number of times.

javascript

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

2. While Loop

Executes a block of code repeatedly while a specified condition is `true`.

javascript

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

3. Do-While Loop

Executes a block of code once and then continues to execute it as long as a specified condition is `true`.

javascript

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

4. For-In Loop

Iterates over the properties of an object.

javascript

```
const obj = { a: 1, b: 2, c: 3 };
for (let key in obj) {
  console.log(`${key}: ${obj[key]}`);
  // Output: a: 1, b: 2, c: 3
}
```

5. For-Of Loop

Iterates over the values of an iterable (e.g., array, string, etc.).

javascript

```
const arr = [10, 20, 30];
for (let value of arr) {
  console.log(value); // Output: 10, 20, 30
}
```

6. forEach Loop

Executes a callback function for each element in an array.

javascript

```
const arr = [10, 20, 30];
arr.forEach((value, index) => {
  console.log(`Index: ${index}, Value: ${value}`);
  // Output: Index: 0, Value: 10
  //          Index: 1, Value: 20
  //          Index: 2, Value: 30
});
```

7. Map Loop

Creates a new array with the results of applying a callback function to each element of the array.

javascript

```
const arr = [1, 2, 3];
const doubled = arr.map((value) => value * 2);
console.log(doubled); // Output: [2, 4, 6]
```

8. Reduce Loop

Applies a callback function to each element of the array to reduce it to a single value.

javascript

```
const arr = [1, 2, 3];
const sum = arr.reduce((accumulator, currentValue) => accumulator + currentValue,
0);
console.log(sum); // Output: 6
```

Summary of Loops:

Loop Type	Use Case	Code Context
For Loop	Iterating a block of code for a specific number of times	Iteration with fixed count
While Loop	Repeating until a condition becomes <code>false</code>	Condition-based repetition
Do-While Loop	Guaranteed to run at least once before checking the condition	Run-once + condition check
For-In Loop	Iterating over the properties of an object	Object property iteration
For-Of Loop	Iterating over the values of an iterable	Array or iterable values
forEach Loop	Array-specific iteration to execute a callback for each element	Array traversal
Map Loop	Transforming elements in an array to create a new array	Array transformation
Reduce Loop	Condensing array values into a single output using a callback	Array reduction (e.g., sum)