



**Intern Name:** Fatima Wajid

**Submitted to:** Ali Hyder

**Date:** 17<sup>th</sup> July 2025

**Internship Role:** Web Development

## **Loops (for, while, do-while) & Arrays in JavaScript**

### Objective

To understand:

- How to **repeat code efficiently** using loops (for, while, do-while)
- How to **store and manipulate multiple values** using arrays
- Learn array methods for handling data

### JavaScript Loops

Loops allow you to run a block of code **multiple times** until a condition is met.

#### 1. for loop

```
for (let i = 1; i <= 5; i++) {  
  console.log("Count:", i);  
}
```

## 2. while loop

```
let i = 1;
while (i <= 5) {
  console.log("While Count:", i);
  i++;
}
```

## 3. do...while loop

```
let i = 1;
do {
  console.log("Do-While Count:", i);
  i++;
} while (i <= 5);
```

Use Cases:

- Use for when the number of iterations is known.
- Use while when the condition might change during runtime.
- Use do...while to **ensure the code runs at least once**.

## JavaScript Arrays

Arrays store **ordered collections** of data (indexed starting from 0).

### ✓ Creating Arrays

```
let fruits = ["apple", "banana", "cherry"];
console.log(fruits[0]); // apple
```

### Array Methods

| Method  | Description           | Example              |
|---------|-----------------------|----------------------|
| push()  | Adds element to end   | fruits.push("mango") |
| pop()   | Removes last element  | fruits.pop()         |
| shift() | Removes first element | fruits.shift()       |

| Method     | Description                    | Example                  |
|------------|--------------------------------|--------------------------|
| unshift()  | Adds element at beginning      | fruits.unshift("kiwi")   |
| length     | Returns number of items        | fruits.length            |
| includes() | Checks if array contains value | fruits.includes("apple") |
| indexOf()  | Gets index of element          | fruits.indexOf("banana") |
| join()     | Combines array into a string   | fruits.join(", ")        |

### Looping through Arrays

```
let colors = ["red", "blue", "green"];
```

```
for (let i = 0; i < colors.length; i++) {  
  console.log(colors[i]);  
}
```

Or using for...of loop:

```
for (let color of colors) {  
  console.log(color);  
}
```

### Mini Task (Day 4)

**Create an array of marks** and calculate the **average** using a loop:

```
let marks = [80, 90, 75, 88, 92];  
let sum = 0;
```

```
for (let mark of marks) {  
  sum += mark;  
}
```

```
let average = sum / marks.length;  
console.log("Average marks:", average);
```

## Conclusion

Loops and arrays are **fundamental tools** for managing data and automating repetitive tasks in JavaScript. Mastering these helps write efficient and scalable code, especially in real-world applications like dashboards, forms, and games.