

Session 2

2. JavaScript and Programming Fundamentals

2.1 Loops

Loops help you run the same block of code multiple times.

For Loop

```
for (let i = 0; i < 3; i++) {  
  console.log(i);  
}
```

Output:

```
0  
1  
2
```

💡 Breakdown:

- Start with `i = 0`
- Run loop while `i < 3`
- After each loop, increase `i` by 1

2.2 Functions

Functions let you reuse logic. Instead of writing the same code over and over, wrap it inside a function.

```
function greet(name) {  
  return "Hello, " + name;  
}
```

```
console.log(greet("Ani"));
```

Output:

```
Hello, Ani
```

You can pass different names:

```
greet("Sara") // Hello, Sara  
greet("Mike") // Hello, Mike
```

👉 You can also use arrow functions:

```
const add = (a, b) => a + b;  
console.log(add(3, 4));
```

Output:

```
7
```



Functions are like mini-machines. You provide input (parameters), and they produce output (return value).

2.3 Arrays

An array is just a list. You access items using their position, called an index. Indexes start from 0.

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

Output:

```
apple
```

2.4 Objects

Objects are a way to group related data. Think of it like a set of labeled boxes.

- Each box has a name (the key)
- Inside the box is a value (the value)

```
let user = {  
  name: "Ani",  
  age: 22,  
  isAdmin: false  
};  
  
console.log(user.name);
```

Output:

```
Ani
```

2.5 Callback Functions

A **callback** is a function passed to another function to be run later.

```
function greet(name, callback) {  
  console.log("Hello " + name);  
  callback();  
}  
  
function sayBye() {  
  console.log("Bye!");  
}  
  
greet("Ani", sayBye);
```

Output:

```
Hello Ani  
Bye!
```

SSSSO Vidyakshina 2025

