JavaScript Study Material

Introduction to JavaScript

JavaScript is a high-level, interpreted programming language primarily used for web development. It allows developers to create dynamic and interactive web pages. JavaScript can manipulate HTML and CSS, respond to user inputs, and communicate with servers.

Key Features of JavaScript

- **Lightweight and Fast**: Runs directly in the browser without requiring compilation.
- **Interpreted Language**: Executed line by line by the JavaScript engine.
- **Dynamic Typing**: No need to define variable types explicitly.
- **Object-Oriented**: Uses prototype-based object-oriented programming.
- **Event-Driven**: Responds to user actions like clicks and keypresses.
- **Cross-Platform**: Works on multiple browsers and devices.

Basic Syntax

1. Variables and Data Types

JavaScript has three ways to declare variables:

```
var name = "John"; // Function-scoped
let age = 25; // Block-scoped
const PI = 3.14; // Block-scoped and immutable
```

Data Types

- **Primitive Types**: Number, String, Boolean, Undefined, Null, Symbol, BigInt
- **Reference Types**: Objects, Arrays, Functions

2. Operators

Arithmetic Operators

```
let sum = 5 + 3;  // Addition
let difference = 5 - 3; // Subtraction
let product = 5 * 3; // Multiplication
let quotient = 5 / 3; // Division
let remainder = 5 % 3; // Modulus
```

Comparison Operators

```
5 == "5"; // true (loose equality)
5 === "5"; // false (strict equality)
5 !== 3; // true
5 > 3; // true
```

Logical Operators

```
true && false; // false (AND)
true || false; // true (OR)
!true; // false (NOT)
```

Control Structures

1. Conditional Statements

```
let age = 18;
if (age >= 18) {
```

```
    console.log("Eligible to vote");
} else {
    console.log("Not eligible to vote");
}
```

2. Loops

For Loop

```
for (let i = 0; i < 5; i++) {
    console.log("Iteration: " + i);
}
```

While Loop

```
let count = 0;
while (count < 5) {
    console.log(count);
    count++;
}
```

Functions in JavaScript

1. Function Declaration

```
function greet(name) {
    return "Hello, " + name;
}
console.log(greet("Alice"));
```

2. Arrow Functions

```
const square = (x) => x * x;
console.log(square(5));
```

Objects and Arrays

1. Objects

```
let person = {
    name: "John",
    age: 30,
    greet: function() {
        console.log("Hello " + this.name);
    }
};
console.log(person.name); // Accessing property
person.greet();
```

2. Arrays

```
let fruits = ["Apple", "Banana", "Cherry"];
console.log(fruits[1]); // Accessing element
fruits.push("Orange"); // Adding element
```

DOM Manipulation

JavaScript interacts with HTML through the Document Object Model (DOM).

```
let element = document×getElementById("demo"); // Selecting element
element×innerHTML = "Hello, World!"; // Modifying content
```

Events Handling

```
document.getElementById("btn").addEventListener("click", function() {
    alert("Button Clicked!");
```

```
});
```

Conclusion

JavaScript is a powerful language essential for modern web development. Understanding its core concepts—variables, loops, functions, objects, and event handling—lays a strong foundation for building interactive applications.