

# JavaScript Programming Fundamentals

## 1. Basic Syntax and Structure

- Comments
- Statements and Semicolons
- Case Sensitivity
- Whitespace and Formatting
- Script Tag for HTML Integration

## 2. Variables and Data Types

- var, let, and const
- Primitive Types: Number, String, Boolean, null, undefined, Symbol
- Complex Types: Object, Array, Function
- Type Conversions (implicit and explicit)

## 3. Operators

- Arithmetic Operators (+, -, \*, /, %)
- Assignment Operators (=, +=, -=)
- Comparison Operators (==, ===, !=, !==, >, <, >=, <=)
- Logical Operators (&&, ||, !)
- Ternary Operator (condition ? trueExpr : falseExpr)
- Bitwise Operators (&, |, ^, ~, <<, >>)

## 4. Control Flow

- Conditional Statements (if, else, else if, switch)

- Loops:
  - for, while, do...while
  - for...in and for...of

## 5. Functions

- Function Declaration vs. Function Expression
- Parameters and Arguments
- Return Values
- Default Parameters
- Arrow Functions (=>)
- Callback Functions
- Immediately Invoked Function Expressions (IIFE)

## 6. Scope and Closures

- Global vs Local Scope
- Block Scope (with let and const)
- Function Scope (with var)
- Hoisting
- Closures (capturing variables in a function's environment)

## 7. Arrays

- Array Declaration and Initialization
- Common Array Methods: push(), pop(), shift(), unshift(), map(), filter(), reduce(), slice(), splice()
- Looping over Arrays

## 8. Objects

- Object Literals and Initialization
- Accessing Properties (dot notation vs bracket notation)
- Adding, Modifying, and Deleting Properties
- Object Methods and this
- Object Destructuring
- Prototype and Prototypal Inheritance

## 9. Strings

- String Declaration ("", "", `` for Template Literals)
- String Methods (length, indexOf(), substring(), replace(), split(), toUpperCase(), toLowerCase())
- String Interpolation with Template Literals
- String Concatenation

## 10. Numbers and Math

- Arithmetic operations
- Working with floating points
- Math Object (Math.round(), Math.floor(), Math.random(), Math.max(), Math.min(), Math.pow())

## 11. Error Handling

- try...catch...finally blocks
- throw Statements
- Error Objects

## 12. Event Handling (for Web Development)

- Adding Event Listeners (addEventListener())
- Common DOM Events (click, keydown, keyup, mouseover, change)

- Event Object (event object, preventDefault(), stopPropagation())

### 13. Document Object Model (DOM) Manipulation

- Selecting Elements (getElementById(), querySelector(), getElementsByClassName())
- Creating and Inserting Elements
- Modifying HTML Content (innerHTML, textContent)
- Modifying Styles and Attributes (style, setAttribute())
- Traversing the DOM (parent, child, sibling nodes)

### 14. Asynchronous Programming

- Callbacks
- Promises:
  - resolve(), reject(), .then(), .catch(), .finally()
- async/await
- Fetch API (fetch(), .then(), .catch())
- Event Loop and JavaScript Concurrency Model

### 15. Modules (ES6 and Beyond)

- Import and Export of modules
- Default vs Named Exports
- Using import and export statements

### 16. Classes (ES6)

- Class Declaration and Instantiation
- Constructor Functions
- Methods

- Inheritance (extends, super)
- Static Methods and Properties

## 17. JSON (JavaScript Object Notation)

- Parsing JSON (JSON.parse())
- Stringifying Objects (JSON.stringify())

## 18. Regular Expressions (RegEx)

- Syntax of Regular Expressions
- Methods: test(), match(), replace()

## 19. Higher-Order Functions

- Functions that accept other functions as arguments or return functions
- Common Higher-Order Functions in Arrays: map(), filter(), reduce()

## 20. Browser APIs

- LocalStorage and SessionStorage
- Geolocation API
- Web Storage API
- Fetch API and AJAX

## 21. Debugging Techniques

- Using console.log(), console.error(), console.table()
- Breakpoints in Developer Tools
- debugger Statement

## 22. Version Control Integration (Optional but Important)

- Using Git and GitHub with JavaScript projects
- Basic git commands (commit, push, pull, merge, branch)