JavaScript Fundamentals

Introduction to JavaScript

Overview of JavaScript

- JavaScript is a high-level, interpreted programming language primarily used for creating interactive web pages.
- Developed by Netscape, JavaScript was initially called LiveScript and later renamed JavaScript.
- JavaScript plays a crucial role in web development by enabling dynamic content, interactivity, and data manipulation on web pages.

History of JavaScript

- JavaScript emerged in the mid-1990s as a scripting language for web browsers.
- Over time, it evolved into a versatile language used for both client-side and server-side scripting.
- JavaScript frameworks and libraries like React, Angular, and Vue.js have further expanded its capabilities.

Variables, Data Types, and Operators

Variables

- Variables in JavaScript are containers for storing data values.
- They are declared using the var, let, or const keywords.
- Variables can hold various data types including numbers, strings, booleans, arrays, objects, etc.

Data Types

- JavaScript supports primitive data types like numbers, strings, booleans, null, undefined, symbols, and BigInt.
- It also supports complex data types like objects and arrays.

Operators

- JavaScript supports various operators including arithmetic, assignment, comparison, logical, and bitwise operators.
- Operators are used to perform operations on variables and values.

Control Flow

Conditional Statements

- Conditional statements like if, else if, and else allow for branching logic based on specified conditions.
- They control the flow of the program by executing different blocks of code depending on the conditions.

Loops

- Loops like for, while, and do-while are used for repetitive tasks.
- They allow executing a block of code repeatedly until a specified condition is met.

Functions and Scope

Functions

- Functions in JavaScript are reusable blocks of code that perform a specific task.
- They are declared using the function keyword and can accept parameters and return values.
- Function expressions and arrow functions are also commonly used.

Parameters and Return Statements

- Parameters allow passing values to functions.
- Return statements are used to return values from functions.

Scope and Closures

Scope

- Scope refers to the visibility of variables within different parts of the code.
- JavaScript has function scope and block scope.
- Variables declared with var have function scope, while variables declared with let and const have block scope.

Closures

- Closures occur when a function has access to variables from its outer scope even after the outer scope has closed.
- They are a powerful feature of JavaScript and are often used to create private variables and functions.

DOM Manipulation and Events

DOM Structure and Hierarchy

- The Document Object Model (DOM) represents the structure of HTML documents as a hierarchical tree of nodes.
- Each node corresponds to an element, attribute, or text in the HTML document.

Selecting Elements

- JavaScript provides methods like getElementById, getElementsByClassName, querySelector, and querySelectorAll to select elements from the DOM.

Modifying Content

- Once selected, elements can be modified by changing their properties like innerHTML, textContent, value, etc.

Handling Events

- Events are actions or occurrences that happen in the browser.
- JavaScript can respond to events triggered by user actions like clicks, keypresses, mouse movements, etc.