JavaScript

Day 1

Introduction

What is JavaScript?

JavaScript is a scripting language initially created to add interactivity to web pages. It has evolved into a powerful language capable of building complex applications. JavaScript runs inside browsers and can be embedded in HTML.

Why is it called JavaScript

- Originally named LiveScript.
- Renamed to JavaScript to ride on Java's popularity during its early days.
- Despite the name JavaScript is fundamentally different from Java.

>> JavaScript × Web Development

Where JavaScript Runs:

- In the browser (client-side)
- On the server (via platforms like Node.js)
- On devices with JS engines

Popular JavaScript Engines:

- V8 – Chrome, Opera, Edge.
- SpiderMonkey – Firefox.
- Chakra – Internet Explorer.
- JavaScriptCore, Nitro, SquirrelFish.

What Can In-Browser JavaScript Do?

- Add and modify HTML/CSS
- React to user actions
- Handle keyboard and mouse events
- Display alerts and popups
- Get/set cookies
- Use local storage for persistence.

Getting Started with JavaScript

- Install Visual Studio Code
- Install Node.js from https://nodejs.org/
- Create a .js file (e.g., app.js)

- Open terminal and run: node app.js

DAY -1 CONTD ...

Embedding JavaScript in HTML

Inline Script:

```
<script>

alert("Hello, world!");

</script>
```

External Script:

```
<script src="HelloWorld.js"></script>
```

Advantages:

- Better code organization Reusability
- Easier debugging

>>Code Structure:

JavaScript is case-sensitive and structured. Use proper indentation and line breaks.

- Comments in JavaScript
- Single-line comment: // comment
- Multi-line comment:

Example:

```
/* This is a multi-line comment */
```

- Semicolons in JavaScript

- Optional, but recommended for clarity and error prevention.

Understanding JavaScript Variable Declarations Variables store data values. Use: var, let, const. **var**

Function-scoped, older standard. Can be re-declared and updated. Hoisted to the top of its scope.

Example:

```
var x = 10;

var x =20; //Allowed
```

DAY -1 CONTD ...

let

Introduced in ES6, block-scoped. Can be updated but not re-declared. Hoisted but not initialized.

Example:

let y = 5; y = 15;

let y = 20; //Error

const

Block-scoped and immutable. Must be initialized at declaration. Object contents can be modified.

Example:

const z = 100;

z = 200; // Error const arr = [1, 2, 3];

arr.push(4); // Allowed