Web Development BASICS: HTML, CSS & JavaScript

Table of Contents

1. HTML Basics

What is HTML?

HTML (HyperText Markup Language) is the standard language for creating web pages. It structures the content on the web.

HTML Structure

Basic structure:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>
</body>
</html>
```

Elements & Tags

Elements: building blocks.

Tags: <tag>content</tag>. Example: <p>Hello World</p>

Attributes

Attributes provide extra info. Example: <a href="https://example.com">Link</a>

Common Tags

<h1>-<h6>, <p>, <div>, <span>, <a>, <img>, <ul>, <ol>, <li>, <table>, <form>

Lists & Links

Lists: <ul>, <ol>, <li>. Links: <a href="URL">text</a>

Images & Multimedia

<img src="image.jpg" alt="Description">

Forms

<form>, <input>, <label>, <select>, <textarea>, <button>

Best Practices

- Use semantic tags.
- Keep it clean & indented.
- Validate your HTML.

Resources

MDN HTML: https://developer.mozilla.org/en-US/docs/Web/HTML

W3Schools HTML: https://www.w3schools.com/html/

2. CSS Basics

What is CSS?

CSS (Cascading Style Sheets) styles your HTML content. It controls layout, colors, fonts, spacing, and responsiveness.

How CSS Works

Web Development BASICS: HTML, CSS & JavaScript

Three ways: Inline, Internal (<style>), External (<link>). External is best for maintenance.

Syntax & Selectors

Syntax: selector { property: value; }

Example: p { color: red; }

Selectors: element, class(.), ID(#), descendant.

Colors, Units, Fonts

Colors: names, hex, rgb.

Units: px, %, em, rem.

Fonts: font-family, font-size, font-weight.

Box Model

Content + Padding + Border + Margin.

.box { width, padding, border, margin }

Positioning & Display

Position: static, relative, absolute, fixed, sticky.

Display: block, inline, inline-block, flex, grid, none.

Flexbox & Grid

Flexbox: for 1D layouts.

Grid: for 2D layouts.

.container { display: flex; }

.container { display: grid; }

Responsive Design

Media queries: @media (max-width: 768px) { ... }.

Use relative units and flexible layouts.

Best Practices

- Keep styles in external files.

- Use classes.

- Use a CSS reset.

- Validate CSS.

Resources

MDN CSS: https://developer.mozilla.org/en-US/docs/Web/CSS

W3Schools CSS: https://www.w3schools.com/css/

3. JavaScript Basics

What is JavaScript?

JavaScript makes web pages interactive. It runs in the browser and manipulates HTML/CSS.

How JavaScript Works

Runs in the browser's JS engine. Uses the DOM to interact with page elements.

Linking JS to HTML

Inline: <button onclick="alert('Hi')">Click</button>

Internal: <script>code</script>

External: <script src="file.js"></script>

Variables & Data Types

let, const, var.

Types: String, Number, Boolean, Array, Object, Null, Undefined.

Web Development BASICS: HTML, CSS & JavaScript

Operators

Arithmetic (+ - * /), Assignment (= +=), Comparison (== === !=), Logical (&& || !).

Functions

function greet() { ... }

Arrow: const add = (a,b) => a+b;

Conditionals

if (condition) {...} else {...}

switch (value) { case 'x': break; default: }

Loops

for, while, forEach.

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

Events

document.getElementById('btn').addEventListener('click', function() { ... });

Best Practices

- Use let/const.
- Keep code DRY.
- Use comments.
- Test in console.

Resources

MDN JS: https://developer.mozilla.org/en-US/docs/Web/JavaScript

W3Schools JS: https://www.w3schools.com/js/