BUDE
GLOBAL ENTERPRISE

INTRODUCTION TO WEB DEVELOPMENT

COMPLETE GUIDE TO MODERN WEB TECHNOLOGIES
FROM BASICS TO ADVANCED CONCEPTS

ARAVIND GOVINDHASAMY

AGP

👔 Software Developer

💻 3 years in .NET, Python, IoT & RFID

🌟 Backend Development & Automation Specialist

🔗 github.com/aravind-govindhasamy

🌐 aravind-govindhasamy.github.io

WHAT IS WEB DEVELOPMENT?

The process of building and maintaining websites covering:

- 🎨 **Frontend** - What users see and interact with

- ⚙️ **Backend** - Server, database, and application logic

- 🧮 **Full Stack** - Combination of frontend and backend

Evolution: From static HTML to dynamic web applications
**Trend:** Moving towards JAMstack and serverless

FRONTEND DEVELOPMENT

Everything users see and interact with directly

🎯 **Goal:** Create engaging, responsive user experiences

🛠️ CORE TECHNOLOGIES

The Building Blocks

- 📄 **HTML** - Structure and content
- 🎨 **CSS** - Styling and layout
- ⚡ **JavaScript** - Interactivity and logic

HTML — THE SKELETON

🏗️ HTML STRUCTURE

Basic HTML Document Structure

```html
<!DOCTYPE html>
<html>
<head>
    <title>My Page</title>
</head>
<body>
    <h1>Hello World!</h1>
    <p>This is a paragraph.</p>
</body>
</html>
```

- 📄 DOCTYPE declares HTML version
- 🏷️ Tags define elements and structure
- 📃 Semantic HTML improves accessibility

🎨 CSS BASICS

Styling with CSS

```css
/* Select element by tag */
h1 {
    color: blue;
    font-size: 2em;
}

/* Select by class */
.highlight {
    background-color: yellow;
}

/* Select by ID */
#main-content {
    width: 80%;
    margin: 0 auto;
}
```

💡 CSS controls presentation: colors, layout, fonts, animations

⚡ JAVASCRIPT BASICS

Adding Interactivity

```javascript
// Variables
let message = 'Hello';
const PI = 3.14;

// Functions
function greet(name) {
    return `${message}, ${name}!`;
}

// Event handling
document.querySelector('button').addEventListener('click', () => {
    alert('Button clicked!');
});
```

- 🔄 Manipulate DOM elements
- 🎯 Handle user interactions
- 📡 Make API calls

MODERN WEB FRAMEWORKS

⚛ POPULAR FRAMEWORKS

- ⭐ **React** - Component-based UI library by Facebook
- 🟢 **Vue** - Progressive framework, easy learning curve
- 🅰 **Angular** - Full-featured framework by Google
- ⚡ **Svelte** - Compiler-based, no virtual DOM

BACKEND — THE BRAIN

🖥️ BACKEND TECHNOLOGIES

- 🌀 **Python** - Django, Flask, FastAPI
- 🪐 **Java** - Spring Boot, enterprise applications
- 💎 **Ruby** - Ruby on Rails, rapid development
- 🟧 **Node.js** - JavaScript on server

💾 DATABASE TYPES

- 📇 **SQL** - Structured, relational (MySQL, PostgreSQL)
- 📄 **NoSQL** - Flexible, document-based (MongoDB)
- 🔑 **Key-Value** - Fast caching (Redis)

🔧 ESSENTIAL TOOLS

- 📦 **Git** - Version control system
- 🐳 **Docker** - Containerization
- ☁️ **Cloud** - AWS, Azure, Google Cloud
- 🔄 **CI/CD** - Automated testing and deployment

WEB DEVELOPMENT BEST PRACTICES

✅ KEY PRINCIPLES

- 📱 **Responsive Design** - Works on all devices
- ⚡ **Performance** - Fast loading times
- 🔒 **Security** - Protect user data
- ♿ **Accessibility** - Everyone can use

💼 WEB DEVELOPMENT CAREERS

- 🎨 **Frontend Developer** - UI/UX implementation
- ⚙️ **Backend Developer** - Server, APIs, databases
- 🚀 **Full Stack Developer** - End-to-end development
- 📱 **Mobile Developer** - React Native, Flutter

🛣️ HOW TO LEARN WEB DEVELOPMENT

- 1️⃣ Master HTML, CSS, JavaScript fundamentals
- 2️⃣ Learn a frontend framework (React/Vue)
- 3️⃣ Understand backend basics (Node.js/Python)
- 4️⃣ Practice with projects and build portfolio
- 5️⃣ Learn DevOps and deployment

LEARNING RESOURCES

- 🌐 **MDN Web Docs** - Official documentation
- 🎓 **freeCodeCamp** - Free interactive courses
- 📺 **YouTube** - Tutorials and walkthroughs
- 💻 **GitHub** - Open source projects

QUESTIONS? 💬

Let's discuss web development, career paths, or specific technologies!
All questions are welcome!

THANK YOU! 🎉

🐍 Ready to code in Python?
Start building something today!

Practice • Build Projects • Join Community

🏢 **Bude Global Enterprise - India**

Empowering developers worldwide