

WEB DEVELOPMENT GUIDE

INTRODUCTION

Welcome to the world of web development! This guide is designed to provide an overview of what we'll be focusing on during the upcoming 5-week course.

Front-end development includes technologies such as HTML (Hypertext markup language), CSS (Cascading Style Sheet) and JavaScript.

HTML sets the structure, CSS styles the appearance, and JavaScript adds functionality to web content. Together, these three technologies form the core of modern web development, allowing developers to create engaging and interactive websites and web applications for both desktop and mobile users.

Day 1: Advanced HTML

- **Semantic HTML:** We'll learn about semantic HTML elements like `<header>`, `<nav>`, `<section>`, `<article>`, `<footer>`, and how they enhance the structure and accessibility of web content.
- **Forms:** Explore advanced form elements such as radio buttons, checkboxes, select menus, and text areas. Create a complex form with validation.
- **Tables:** Understand how to create tables in HTML and style them using CSS for better presentation. Learn about table headers, rows, and columns.
- **HTML Media:** Embed multimedia content like videos and audio using HTML5 `<video>` and `<audio>` elements. Learn to set attributes for media elements.

Day 2: CSS Styling Techniques

- **CSS Flexbox:** Dive into CSS flexbox, a powerful layout model for creating responsive and flexible web designs. Create a layout with flexbox for our PlayStation website.
- **CSS Grid:** Explore CSS grid layout, another layout model for designing grid-based layouts. Learn how to define grid containers and grid items.
- **Responsive Design:** Understand the principles of responsive web design. Use media queries to adapt your webpage's layout and styles for different screen sizes and devices.
- **Advanced CSS Selectors:** Explore advanced CSS selectors like `:hover`, `:nth-child`, and `:not`. Use them to target specific elements for styling.

Day 3: Introduction to Bootstrap

- **What is Bootstrap:** Introduce Bootstrap, a popular CSS framework, and its benefits for web development.
- **Setting Up Bootstrap:** Learn how to include Bootstrap in your project using CDN links or by downloading and linking the Bootstrap CSS and JavaScript files.
- **Bootstrap Grid System:** Explore Bootstrap's grid system for creating responsive layouts. Create a responsive webpage layout using Bootstrap's grid classes.

Day 4: Bootstrap Components

- **Bootstrap Components:** Discover common Bootstrap components like navigation bars, buttons, forms, and cards. Learn how to use these components to enhance your webpage's design and functionality.
- **Customizing Bootstrap:** Understand how to customize Bootstrap styles to match your project's branding and design requirements. Modify Bootstrap variables or use custom CSS.
- **Bootstrap Resources:** Explore Bootstrap documentation and resources to help you leverage Bootstrap's features effectively.

Day 5: Building the ReqZone Movie Website (Part 1)

- **Project Overview:** Begin by defining the structure and features of the ReqZone Movie website. Decide on the main sections like the homepage, movie listings, and movie details pages.
- **HTML Structure:** Create the HTML structure for the website, including the main layout, navigation bar, and placeholders for movie listings.
- **CSS Styling (Bootstrap):** Apply Bootstrap classes to style the website's layout, navigation bar, and basic elements. Customize the color scheme and fonts as needed.
- **Homepage:** Design and populate the homepage with introductory content, such as a welcome message and featured movies.
- **Movie Listings:** Create a movie listings page with a grid layout using Bootstrap's grid system. Add placeholders for movie posters and titles.

Day 6: Building the ReqZone Movie Website (Part 2)

- **Movie Details Page:** Create a movie details page template with placeholders for movie information like title, genre, release date, and a brief description.
- **Responsive Design:** Ensure that your website is responsive by testing it on different screen sizes and adjusting the layout as needed using Bootstrap classes.

- **Testing and Debugging:** Test your website thoroughly to identify and fix any issues. Use browser developer tools for debugging and ensuring a smooth user experience.
- **Final Touches:** Add any additional features or styling enhancements you'd like to include. Make sure that your website is user-friendly and visually appealing.
- **Documentation:** Consider documenting the code and functionality for future reference.

Day 7: JavaScript Fundamentals (Part 1)

- **Introduction to JavaScript:** Start with an overview of JavaScript, its history, and its role in web development.
- **Variables and Data Types:** Dive into JavaScript variables, data types (e.g., numbers, strings, booleans), and how to declare and use them.
- **Operators:** Learn about JavaScript operators (e.g., arithmetic, comparison, logical) and how they work.
- **Conditional Statements:** Explore conditional statements, including **if**, **else if**, and **else**. Learn how to make decisions in your code.
- **Functions:** Introduce JavaScript functions, how to define them, and how to use them.

Day 8: JavaScript Fundamentals (Part 2)

- **Loops:** Explore JavaScript loops, including **for**, **while**, and **do...while**. Learn how to iterate through data.
- **Arrays:** Understand JavaScript arrays, how to create them, access elements, and perform common array operations.
- **Objects:** Learn about JavaScript objects, how to create them, and how to access and manipulate object properties.
- **DOM Introduction:** Introduce the Document Object Model (DOM) and its role in web development.
- **Event Handling Basics:** Get started with event handling in JavaScript by attaching event listeners to DOM elements.

Day 9: DOM Manipulation (Part 1)

- **DOM Basics:** Review the basics of the DOM, including how it represents the structure of a webpage as a tree of objects.
- **Selecting DOM Elements:** Learn how to select DOM elements using methods like **getElementById**, **getElementsByClassName**, and **querySelector**.
- **Modifying DOM Elements:** Practice changing the content and attributes of DOM elements dynamically using JavaScript.

- **Event Handling:** Explore event handling in JavaScript by adding event listeners to elements and responding to user interactions.

Day 10: Advanced DOM Manipulation (Part 2)

- **Creating DOM Elements:** Learn how to create new DOM elements dynamically using JavaScript and add them to the webpage.
- **Traversal and Manipulation:** Dive deeper into traversing the DOM tree and manipulating elements, such as adding, removing, and moving elements.
- **Event Delegation:** Explore the concept of event delegation, a powerful technique for efficiently handling events on multiple elements.
- **Project:** Apply your DOM manipulation skills to enhance your "ReqZone Movie" website. You can add features like dynamic content loading, interactive forms, or pop-up modal dialogs using DOM manipulation.