

Beginner Portfolio Website - Guide

Learning Objectives

By the end of this class, students will be able to:

- Structure a webpage using semantic HTML5
- Style a website with CSS including layouts, colors, and animations
- Add interactivity using JavaScript
- Understand how HTML, CSS, and JavaScript work together
- Create a responsive portfolio website

Class Structure

Part 1: Introduction

What We're Building: A complete portfolio website with multiple sections including home, about, skills, projects, and contact form.

Key Concepts to Introduce:

- HTML provides the structure (the bones)
- CSS provides the styling (the skin)
- JavaScript provides the interactivity (the brain)

Part 2: HTML Structure

Concepts to Cover:

1. Document Structure

html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>My Portfolio</title>
</head>
<body>
</body>
</html>
```

2. Semantic HTML Elements

- <header> - Top section with navigation
- <nav> - Navigation menu

- <section> - Content sections
- <footer> - Bottom section

3. Key HTML Elements Used:

- Headings: <h1>, <h2>, <h3>
- Paragraphs: <p>
- Lists: ,
- Links:
- Buttons: <button>
- Forms: <form>, <input>, <textarea>, <label>
- Divs: <div> for grouping content

Part 3: CSS Styling

Concepts to Cover:

1. CSS Reset

CSS

```
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
```

Explain why we reset default browser styles.

2. Selectors

- Element selectors: body, header, h1
- Class selectors: .hero, .btn, .skill-card
- ID selectors: #home, #about

3. Layout Techniques

- **Flexbox** for navigation and about section

CSS

```
display: flex;
justify-content: space-between;
align-items: center;
```

- **CSS Grid** for skills and projects

CSS

```
display: grid;
grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
gap: 2rem;
```

4. Colors and Gradients

CSS

```
background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
```

5. Transitions and Hover Effects

CSS

```
transition: transform 0.3s;
```

CSS

```
.btn:hover {  
  transform: translateY(-2px);  
}
```

6. Responsive Design

- Using max-width for containers
- Using auto-fit in grid for automatic responsiveness

Part 4: JavaScript Interactivity

Concepts to Cover:

1. Selecting Elements

javascript

```
document.querySelector('#contact')  
document.querySelectorAll('nav a')  
document.getElementById('name')
```

2. Event Listeners

javascript

```
element.addEventListener('click', function() {  
  // code here  
});
```

3. Functions

- Named functions: `function toggleTheme() { }`
- Arrow functions: `() => { }`

4. Key Features Explained: A. Smooth Scrolling Navigation

javascript

```
document.querySelectorAll('nav a').forEach(link => {
```

```

link.addEventListener('click', function(e) {
  e.preventDefault();
  const target = document.querySelector(this.getAttribute('href'));
  target.scrollIntoView({ behavior: 'smooth' });
});
});

```

- Prevents default link behavior
- Gets target section
- Scrolls smoothly to it

B. Theme Toggle

javascript

```

function toggleTheme() {
  document.body.classList.toggle('dark-theme');
}

```

- Adds/removes CSS class
- Changes entire site appearance

C. Form Handling

javascript

```

function handleSubmit(e) {
  e.preventDefault();
  const name = document.getElementById('name').value;
  alert(`Thank you, ${name}!`);
  e.target.reset();
}

```

- Prevents form submission
- Gets form values
- Shows confirmation
- Resets form

D. Scroll Animations

javascript

```

window.addEventListener('scroll', () => {
  // Check if element is in viewport
  // Apply animation

```

});

Part 5: Bringing It All Together (20 minutes)

How the Three Technologies Work Together:

1. **HTML** creates the structure
 - Navigation menu with links
 - Form with input fields
 - Sections with content
2. **CSS** makes it beautiful
 - Colors and gradients
 - Layout and spacing
 - Animations and effects
3. **JavaScript** makes it interactive
 - Clicking navigation scrolls smoothly
 - Theme toggle changes colors
 - Form validates and shows messages
 - Animations trigger on scroll

Student Exercises

Exercise 1: Customize Content (Beginner)

- Change the name from "John Doe" to their own name
- Update the about section with their information
- Change skill names and descriptions

Exercise 2: Modify Styles (Intermediate)

- Change the gradient colors
- Modify font sizes and spacing
- Try different hover effects
- Change the dark theme colors

Exercise 3: Add Features (Advanced)

- Add a new project card
- Create a new skill card
- Add social media links in the footer
- Modify the form to include a phone number field