Lesson 1.2 - CSS Fundamentals: The Language of Design

🎯 Lesson Overview

Learning Objectives:

- Understand what CSS is and how it transforms HTML
- Learn basic CSS syntax and selectors
- Master fundamental styling properties (colors, fonts, spacing)
- Understand the Box Model concept
- Create responsive layouts with Flexbox basics
- Build a professional-looking resume from the HTML version

Lesson Duration: 90 minutes

Difficulty Level: Beginner (building on HTML knowledge)

Required Tools: VS Code + Live Server + Browser DevTools

📚 Quick Review - What We Know

🔄 From Lesson 1.1:

- HTML creates **structure and content**
- Tags define **what** things are
- Attributes provide **additional information**
- We can build functional, but plain-looking pages

🎨 Today's Mission:

Transform those plain HTML pages into **beautiful, professional websites**!

🌟 Part A: CSS Fundamentals (30 minutes)

🎯 What is CSS?

CSS (Cascading Style Sheets) is the language that controls how HTML elements **look and are positioned** on a webpage.

🏗️ The Building Analogy:

- **HTML** = The house structure (walls, rooms, doors)
- **CSS** = The decoration (paint, furniture, lighting)
- **JavaScript** = The electricity and plumbing (functionality)

🔤 CSS Syntax - The Basic Rules

Every CSS rule has the same structure:

```css
selector {
    property: value;
    property: value;
}
```

Example:

```css
h1 {
    color: blue;
    font-size: 36px;
}
```

Translation: "Find all h1 elements and make them blue with 36px font size"

🎯 CSS Selectors - How to Target Elements

1. **Element Selector** (targets all elements of that type):

```css
p {
    color: green;
}
```

2. **Class Selector** (targets elements with specific class):

```css
```

```css
.highlight {
    background-color: yellow;
}
```

```html
<p class="highlight">This text will have yellow background</p>
```

3. ID Selector (targets one specific element):

```css
#header {
    text-align: center;
}
```

```html
<h1 id="header">This heading will be centered</h1>
```

4. Multiple Selectors (apply same style to multiple elements):

```css
h1, h2, h3 {
    font-family: Arial, sans-serif;
}
```

🎨 Practical Exercise 1: Your First CSS File (15 minutes)

🎯 Mission: Create a CSS file and apply basic styling

Step 1: Create your workspace

1. Open VS Code
2. Navigate to `02-CSS-Projects` folder
3. Create new folder: `css-basics`
4. Create two files:
    - `index.html`

- `style.css`

Step 2: HTML Foundation

html

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My First CSS Page</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <h1 id="main-title">Welcome to CSS World!</h1>
    <p class="intro">This is my first styled paragraph.</p>
    <p>This is a regular paragraph.</p>
    <h2>About CSS</h2>
    <p class="highlight">CSS makes everything beautiful!</p>
</body>
</html>
```

Step 3: CSS Magic

css

```
/* Basic styling for all paragraphs */
p {
    font-size: 18px;
    line-height: 1.5;
}

/* Styling the main title */
#main-title {
    color: #2E86AB;
    text-align: center;
    font-size: 48px;
}

/* Special styling for intro paragraphs */
.intro {
    color: #666;
    font-style: italic;
}

/* Highlighted text */
.highlight {
    background-color: #F24236;
    color: white;
    padding: 10px;
    border-radius: 5px;
}

/* All headings */
h1, h2 {
    font-family: 'Georgia', serif;
}
```

Step 4: Test and Experiment

- Open the file with Live Server
- Try changing colors, sizes, and other values
- Add new classes and styles

🎨 Essential CSS Properties - Your Design Toolkit

🌈 Colors

Color Values:

```css
/* Named colors */
color: red;
color: blue;

/* Hex colors (most common) */
color: #FF0000; /* Red */
color: #0066CC; /* Blue */

/* RGB colors */
color: rgb(255, 0, 0); /* Red */

/* HSL colors */
color: hsl(0, 100%, 50%); /* Red */
```

Color Properties:

```css
.text-styling {
  color: #333;           /* Text color */
  background-color: #F0F0F0; /* Background color */
  border-color: #CCC;     /* Border color */
}
```

🔤 Typography

Font Properties:

```css
```

```
.typography {
    font-family: 'Arial', sans-serif;
    font-size: 16px;
    font-weight: bold;       /* normal, bold, 100-900 */
    font-style: italic;      /* normal, italic */
    text-align: center;      /* left, center, right, justify */
    text-decoration: underline; /* none, underline, line-through */
    line-height: 1.5;        /* Space between lines */
    letter-spacing: 1px;     /* Space between letters */
}
```

📏 Spacing and Sizing

Size Properties:

```css
.sizing {
    width: 300px;
    height: 200px;
    max-width: 100%;
    min-height: 150px;
}
```

Units in CSS:

- **px** (pixels) - exact size
- **%** (percentage) - relative to parent
- **em** - relative to font size
- **rem** - relative to root font size
- **vh/vw** - viewport height/width

📦 Part B: The Box Model - The Secret of Layout (30 minutes)

🎯 Understanding the Box Model

Every HTML element is a **box** with four layers:

```
┌─────────────────────────────────────┐
│           MARGIN                    │
│  ┌───────────────────────────────┐  │
│  │        BORDER                 │  │
│  │  ┌─────────────────────────┐  │  │
│  │  │      PADDING            │  │  │
│  │  │  ┌───────────────────┐  │  │  │
│  │  │  │   CONTENT         │  │  │  │
│  │  │  └───────────────────┘  │  │  │
│  │  └─────────────────────────┘  │  │
│  └───────────────────────────────┘  │
└─────────────────────────────────────┘
```

Box Model Properties:

```css
.box-example {
  /* Content area */
  width: 200px;
  height: 100px;

  /* Padding - space inside the element */
  padding: 20px;
  padding-top: 10px;
  padding-right: 15px;
  padding-bottom: 10px;
  padding-left: 15px;

  /* Border - the frame around the element */
  border: 2px solid #333;
  border-radius: 8px;

  /* Margin - space outside the element */
  margin: 30px;
  margin-top: 20px;
  margin-bottom: 40px;
}
```

Shorthand Properties:

```css
```

```
/* Padding/Margin shorthand */
padding: 10px;                /* All sides 10px */
padding: 10px 20px;           /* Top/bottom 10px, left/right 20px */
padding: 10px 20px 15px;      /* Top 10px, left/right 20px, bottom 15px */
padding: 10px 20px 15px 25px; /* Top, right, bottom, left (clockwise) */
```

🎨 Practical Exercise 2: Box Model Mastery (15 minutes)

🎯 Mission: Create styled cards using the Box Model

Create a new file: `box-model-practice.html`

```html
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Box Model Practice</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f0f0f0;
            padding: 20px;
        }

        .card {
            background-color: white;
            border: 1px solid #ddd;
            border-radius: 8px;
            padding: 20px;
            margin: 20px 0;
            box-shadow: 0 2px 4px rgba(0,0,0,0.1);
        }

        .card h3 {
            color: #2E86AB;
            margin-top: 0;
            border-bottom: 2px solid #F24236;
            padding-bottom: 10px;
        }

        .highlight-box {
            background-color: #FFF3CD;
            border: 2px solid #FFC107;
            border-radius: 6px;
            padding: 15px;
            margin: 15px 0;
        }

        .button {
            background-color: #28A745;
            color: white;
            padding: 12px 24px;
            border: none;
            border-radius: 5px;
            margin: 10px 5px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <div class="card">
        <h3>Project 1: Personal Website</h3>
        <p>Build your first personal website with HTML and CSS.</p>
        <div class="highlight-box">
            <strong>Tip:</strong> Start with a simple layout and add complexity gradually.
        </div>
        <button class="button">View Project</button>
        <button class="button">Edit Code</button>
    </div>

    <div class="card">
        <h3>Project 2: Business Landing Page</h3>
        <p>Create a professional landing page for a fictional business.</p>
        <div class="highlight-box">
            <strong>Challenge:</strong> Make it responsive for mobile devices.
        </div>
        <button class="button">View Project</button>
        <button class="button">Edit Code</button>
    </div>
</body>
</html>
```

Experiment:

- Change padding values and see the effect
- Modify border styles and colors
- Adjust margins between cards
- Try different border-radius values

📐 Part C: Layout with Flexbox (30 minutes)

🎯 What is Flexbox?

Flexbox is a powerful CSS layout method that makes it easy to:

- Arrange elements in rows or columns

- Center content perfectly
- Distribute space evenly
- Create responsive layouts

🔧 Flexbox Basics

Step 1: Create a Flex Container

```
css

.container {
  display: flex;
}
```

Step 2: Control the Direction

```
css

.container {
  display: flex;
  flex-direction: row;         /* Default: left to right */
  flex-direction: column;      /* Top to bottom */
  flex-direction: row-reverse; /* Right to left */
}
```

Step 3: Justify Content (Main Axis)

```
css

.container {
  display: flex;
  justify-content: flex-start;    /* Default: start */
  justify-content: center;        /* Center */
  justify-content: flex-end;      /* End */
  justify-content: space-between; /* Space between items */
  justify-content: space-around;  /* Space around items */
  justify-content: space-evenly;  /* Even space */
}
```

Step 4: Align Items (Cross Axis)

```
css
```

```
.container {
  display: flex;
  align-items: stretch;     /* Default: stretch to fill */
  align-items: flex-start;  /* Top */
  align-items: center;      /* Middle */
  align-items: flex-end;    /* Bottom */
}
```

🌐 Responsive Design Basics

Mobile-First Approach:

```
css

/* Mobile styles (default) */
.container {
  flex-direction: column;
}

/* Tablet and up */
@media (min-width: 768px) {
  .container {
    flex-direction: row;
  }
}

/* Desktop and up */
@media (min-width: 1024px) {
  .container {
    max-width: 1200px;
    margin: 0 auto;
  }
}
```

🎨 Practical Exercise 3: Flexbox Layout (15 minutes)

🎯 Mission: Create a responsive navigation and hero section

Create: `flexbox-practice.html`

```
html
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Flexbox Layout</title>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
        }

        body {
            font-family: Arial, sans-serif;
        }

        /* Navigation */
        .navbar {
            display: flex;
            justify-content: space-between;
            align-items: center;
            background-color: #2E86AB;
            padding: 1rem 2rem;
            color: white;
        }

        .nav-links {
            display: flex;
            list-style: none;
            gap: 2rem;
        }

        .nav-links a {
            color: white;
            text-decoration: none;
            padding: 0.5rem 1rem;
            border-radius: 4px;
            transition: background-color 0.3s;
        }

        .nav-links a:hover {
            background-color: rgba(255,255,255,0.2);
```

```css
}

/* Hero Section */
.hero {
    display: flex;
    align-items: center;
    justify-content: center;
    min-height: 400px;
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    color: white;
    text-align: center;
}

.hero-content h1 {
    font-size: 3rem;
    margin-bottom: 1rem;
}

.hero-content p {
    font-size: 1.2rem;
    margin-bottom: 2rem;
}

.btn {
    display: inline-block;
    background-color: #F24236;
    color: white;
    padding: 12px 30px;
    text-decoration: none;
    border-radius: 25px;
    transition: transform 0.3s, box-shadow 0.3s;
}

.btn:hover {
    transform: translateY(-2px);
    box-shadow: 0 4px 8px rgba(0,0,0,0.2);
}

/* Features Section */
.features {
    display: flex;
    flex-wrap: wrap;
    gap: 2rem;
    padding: 4rem 2rem;
```

```
            max-width: 1200px;
            margin: 0 auto;
        }

        .feature-card {
            flex: 1;
            min-width: 250px;
            text-align: center;
            padding: 2rem;
            border-radius: 8px;
            box-shadow: 0 2px 10px rgba(0,0,0,0.1);
            transition: transform 0.3s;
        }

        .feature-card:hover {
            transform: translateY(-5px);
        }

        .feature-icon {
            font-size: 3rem;
            margin-bottom: 1rem;
        }

        /* Responsive Design */
        @media (max-width: 768px) {
            .navbar {
                flex-direction: column;
                gap: 1rem;
            }

            .nav-links {
                gap: 1rem;
            }

            .hero-content h1 {
                font-size: 2rem;
            }

            .features {
                flex-direction: column;
            }
        }
    </style>
</head>
```

```html
<body>
    <nav class="navbar">
        <div class="logo">
            <h2>MyWebsite</h2>
        </div>
        <ul class="nav-links">
            <li><a href="#home">Home</a></li>
            <li><a href="#about">About</a></li>
            <li><a href="#services">Services</a></li>
            <li><a href="#contact">Contact</a></li>
        </ul>
    </nav>

    <section class="hero">
        <div class="hero-content">
            <h1>Welcome to My World</h1>
            <p>Creating amazing web experiences with HTML and CSS</p>
            <a href="#" class="btn">Get Started</a>
        </div>
    </section>

    <section class="features">
        <div class="feature-card">
            <div class="feature-icon">🎨</div>
            <h3>Beautiful Design</h3>
            <p>Creating visually stunning websites that capture attention and engage users.</p>
        </div>
        <div class="feature-card">
            <div class="feature-icon">📱</div>
            <h3>Responsive</h3>
            <p>All websites work perfectly on desktop, tablet, and mobile devices.</p>
        </div>
        <div class="feature-card">
            <div class="feature-icon">⚡</div>
            <h3>Fast Loading</h3>
            <p>Optimized code ensures your website loads quickly for the best user experience.</p>
        </div>
    </section>
</body>
</html>
```

🏆 Final Project: Professional Resume Makeover (25 minutes)

🎯 Mission: Transform your HTML resume into a professional, styled document

Take your resume from Lesson 1.1 and apply everything you've learned!

Requirements:

1. **External CSS file** - proper file organization
2. **Color scheme** - professional and consistent
3. **Typography** - readable and hierarchical
4. **Box Model** - proper spacing and layout
5. **Flexbox** - modern layout techniques
6. **Responsive design** - works on mobile and desktop
7. **Hover effects** - interactive elements

Starter Template:

Create: `professional-resume.html` and `resume-style.css`

HTML Structure:

```html
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Your Name - Resume</title>
    <link rel="stylesheet" href="resume-style.css">
    <link href="https://fonts.googleapis.com/css2?family=Inter:wght@300;400;600;700&display=swap" rel="stylesheet"
</head>
<body>
    <div class="container">
        <header class="header">
            <div class="profile-section">
                <div class="profile-info">
                    <h1 class="name">Your Name</h1>
                    <p class="title">Web Developer in Training</p>
                    <div class="contact-info">
                        <span class="contact-item">📧 your.email@example.com</span>
                        <span class="contact-item">📱 (555) 123-4567</span>
                        <span class="contact-item">📍 Your City, Country</span>
                    </div>
                </div>
            </div>
        </header>

        <main class="main-content">
            <section class="section">
                <h2 class="section-title">About Me</h2>
                <div class="section-content">
                    <p>Write a compelling summary about yourself, your goals, and what makes you unique. Keep it concise b
                </div>
            </section>

            <section class="section">
                <h2 class="section-title">Skills</h2>
                <div class="section-content">
                    <div class="skills-grid">
                        <div class="skill-item">
                            <span class="skill-name">HTML</span>
                            <div class="skill-bar">
                                <div class="skill-progress" style="width: 85%"></div>
                            </div>
                        </div>
                <div class="skill-item">
                    <span class="skill-name">CSS</span>
                    <div class="skill-bar">
                        <div class="skill-progress" style="width: 70%"></div>
                    </div>
                </div>
                <div class="skill-item">
                    <span class="skill-name">JavaScript</span>
                    <div class="skill-bar">
                        <div class="skill-progress" style="width: 40%"></div>
                    </div>
                </div>
            </div>
        </div>
    </section>

    <section class="section">
        <h2 class="section-title">Education</h2>
        <div class="section-content">
            <div class="timeline-item">
                <div class="timeline-marker"></div>
                <div class="timeline-content">
                    <h3>High School Name</h3>
                    <p class="timeline-date">2020 - 2024</p>
                    <p>Brief description of your studies and achievements.</p>
                </div>
            </div>
        </div>
    </section>

    <section class="section">
        <h2 class="section-title">Projects</h2>
        <div class="section-content">
            <div class="projects-grid">
                <div class="project-card">
                    <h3>Personal Website</h3>
                    <p>My first HTML and CSS website showcasing my learning journey.</p>
                    <div class="project-tech">
                        <span class="tech-tag">HTML</span>
                        <span class="tech-tag">CSS</span>
                    </div>
                </div>
                <div class="project-card">
                    <h3>Resume Website</h3>
```

```
              <p>A responsive resume built with modern CSS techniques.</p>
              <div class="project-tech">
                <span class="tech-tag">HTML</span>
                <span class="tech-tag">CSS</span>
                <span class="tech-tag">Flexbox</span>
              </div>
            </div>
          </div>
        </div>
      </section>
    </main>
  </div>
</body>
</html>
```

CSS Styling (resume-style.css):

```css
```

```css
/* Reset and base styles */
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

body {
    font-family: 'Inter', sans-serif;
    line-height: 1.6;
    color: #333;
    background-color: #f8f9fa;
}

.container {
    max-width: 900px;
    margin: 0 auto;
    background-color: white;
    box-shadow: 0 0 20px rgba(0,0,0,0.1);
    border-radius: 8px;
    overflow: hidden;
}

/* Header styles */
.header {
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    color: white;
    padding: 3rem 2rem;
    text-align: center;
}

.name {
    font-size: 3rem;
    font-weight: 700;
    margin-bottom: 0.5rem;
}

.title {
    font-size: 1.2rem;
    font-weight: 300;
    margin-bottom: 2rem;
    opacity: 0.9;
}
```

```
.contact-info {
    display: flex;
    justify-content: center;
    gap: 2rem;
    flex-wrap: wrap;
}

.contact-item {
    background-color: rgba(255,255,255,0.2);
    padding: 0.5rem 1rem;
    border-radius: 20px;
    font-size: 0.9rem;
}

/* Main content */
.main-content {
    padding: 2rem;
}

.section {
    margin-bottom: 3rem;
}

.section-title {
    font-size: 1.8rem;
    color: #2c3e50;
    margin-bottom: 1.5rem;
    padding-bottom: 0.5rem;
    border-bottom: 3px solid #3498db;
    position: relative;
}

.section-title::after {
    content: '';
    position: absolute;
    bottom: -3px;
    left: 0;
    width: 50px;
    height: 3px;
    background-color: #e74c3c;
}

.section-content {
    font-size: 1rem;
    line-height: 1.7;
}

/* Skills section */
.skills-grid {
    display: grid;
    gap: 1.5rem;
}

.skill-item {
    display: flex;
    align-items: center;
    gap: 1rem;
}

.skill-name {
    min-width: 100px;
    font-weight: 600;
}

.skill-bar {
    flex: 1;
    height: 8px;
    background-color: #ecf0f1;
    border-radius: 4px;
    overflow: hidden;
}

.skill-progress {
    height: 100%;
    background: linear-gradient(90deg, #3498db, #2980b9);
    transition: width 0.3s ease;
}

/* Timeline styles */
.timeline-item {
    display: flex;
    gap: 1rem;
    position: relative;
}

.timeline-marker {
    width: 12px;
```

```css
    height: 12px;
    background-color: #3498db;
    border-radius: 50%;
    margin-top: 0.5rem;
    flex-shrink: 0;
}

.timeline-content h3 {
    color: #2c3e50;
    margin-bottom: 0.5rem;
}

.timeline-date {
    color: #7f8c8d;
    font-weight: 600;
    margin-bottom: 0.5rem;
}

/* Projects grid */
.projects-grid {
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
    gap: 1.5rem;
}

.project-card {
    background-color: #f8f9fa;
    padding: 1.5rem;
    border-radius: 8px;
    border: 1px solid #e9ecef;
    transition: transform 0.3s ease, box-shadow 0.3s ease;
}

.project-card:hover {
    transform: translateY(-3px);
    box-shadow: 0 5px 15px rgba(0,0,0,0.1);
}

.project-card h3 {
    color: #2c3e50;
    margin-bottom: 0.5rem;
}

.project-tech {
    margin-top: 1rem;
    display: flex;
    gap: 0.5rem;
    flex-wrap: wrap;
}

.tech-tag {
    background-color: #3498db;
    color: white;
    padding: 0.3rem 0.8rem;
    border-radius: 15px;
    font-size: 0.8rem;
    font-weight: 500;
}

/* Responsive design */
@media (max-width: 768px) {
    .container {
        margin: 1rem;
        border-radius: 0;
    }

    .header {
        padding: 2rem 1rem;
    }

    .name {
        font-size: 2rem;
    }

    .contact-info {
        flex-direction: column;
        gap: 0.5rem;
    }

    .main-content {
        padding: 1rem;
    }

    .section-title {
        font-size: 1.5rem;
    }

    .skill-item {
        flex-direction: column;
        align-items: flex-start;
    }

    .skill-name {
        min-width: auto;
    }

    .projects-grid {
        grid-template-columns: 1fr;
    }
}

/* Print styles */
@media print {
    .container {
        box-shadow: none;
        max-width: none;
    }

    .header {
        background: #667eea !important;
        -webkit-print-color-adjust: exact;
    }
}
```

📊 Assessment Criteria

🏆 Excellent (90-100%):

- Clean, professional appearance
- Proper use of all CSS concepts covered
- Responsive design that works on all devices
- Creative use of colors and typography
- Smooth hover effects and transitions
- Well-organized and commented code

⭐ Good (80-89%):

- Good visual appeal with minor issues

- Most CSS concepts applied correctly
- Basic responsive design
- Consistent styling throughout
- Some interactive elements

✅ Satisfactory (70-79%):

- Basic styling applied
- Some responsive elements
- Functional but could be more polished
- Missing some advanced features

🔄 Needs Improvement (<70%):

- Minimal styling applied
- Poor visual hierarchy
- Not responsive
- Code organization issues
- Missing key requirements

📝 Lesson Summary

🎓 What We Learned Today:

- **CSS Fundamentals** - syntax, selectors, and basic properties
- **Box Model** - understanding spacing and layout
- **Color and Typography** - making content visually appealing
- **Flexbox Basics** - modern layout techniques
- **Responsive Design** - mobile-first approach
- **Professional Styling** - creating polished, real-world designs

🚀 Next Steps:

- **Practice daily**