

DAY-3 Hands-On Activities – Keerthivasan R

Problem 1

Assessment Goal: Check ability to apply styling and understand selectors and box model.

Hands-on Tasks:

1. Create an external CSS file and link it to HTML
2. Apply different text styles to headings and paragraphs
3. Add background color and borders to a section
4. Demonstrate margin and padding on a div
5. Style the form created on Day 2

Expected Outcome:

A visually styled webpage using external CSS and proper selectors.

CODE : index.html

```
<!DOCTYPE html>
<!-- Declares the document type and tells the browser this is HTML5 -->

<html lang="en">
<!-- Root element of the webpage.
     lang="en" helps with accessibility and SEO -->

<head>
    <meta charset="UTF-8">
    <!-- Sets character encoding to UTF-8
        so special characters display correctly -->

    <title>Styled Webpage Example</title>
    <!-- Title shown in the browser tab -->

    <!-- Link external CSS file to separate styling from structure -->
    <link rel="stylesheet" href="style.css">
</head>

<body>
    <!-- Contains all visible content of the webpage -->

    <!-- Header Section -->
    <header>
        <!-- Semantic tag used for introductory content or headings -->

        <h1>Welcome to My Styled Page</h1>
        <!-- Main heading (most important heading on page) -->

        <h2>Subheading with Styling</h2>
        <!-- Secondary heading for structured content -->

        <p>This paragraph introduces the webpage and demonstrates text styling.</p>
        <!-- Paragraph tag used for normal text content -->
    </header>

    <!-- Main Section with Background and Borders -->
```

```
<section class="styled-section">
    <!-- Section groups related content together.
        class is used to apply CSS styling -->

        <h3>Section Title</h3>
        <!-- Third-level heading inside section -->

        <p>This section has a background color, border, and padding to show styling
effects.</p>
    </section>

    <!-- Box Model Demonstration -->
    <div class="box-model-demo">
        <!-- div is a generic container used mainly for styling and layout -->

        <p>
            This div demonstrates
            <strong>margin</strong>
            <!-- strong tag makes text bold and adds importance -->

            and
            <strong>padding</strong>.
        </p>
    </div>

    <!-- Form Section -->
    <section>
        <!-- Section groups form-related content -->

        <h3>Sign Up Form</h3>

        <form class="signup-form">
            <!-- form is used to collect user input.
                class allows CSS styling -->

            <label for="name">Name:</label>
            <!-- label improves accessibility.
                for attribute connects label to input id -->

            <input type="text" id="name" name="name" placeholder="Enter your name">
            <!-- input field for text entry.
                id links to label.
                name used when submitting data.
                placeholder shows hint text -->

            <label for="email">Email:</label>

            <input type="email" id="email" name="email" placeholder="Enter your email">
            <!-- type="email" ensures proper email format validation -->

            <label for="course">Course:</label>

            <select id="course" name="course">
                <!-- select creates dropdown list -->
```

```

        <option value="">
            Select a course
        </option>
        <!-- option defines selectable item -->

        <option value="java">Java</option>
        <option value="python">Python</option>
        <option value="web">Web Development</option>
    </select>

    <button type="submit">Submit</button>
    <!-- button submits the form data -->
</form>
</section>

</body>

</html>

```

CODE : style.css

```

/* Universal Selector */
* {
    font-family: Arial, Helvetica, sans-serif;
    /* Applies default font to entire page */

    box-sizing: border-box;
    /* Includes padding & border inside element's total width/height
       Makes layout easier to control */
}

/* ----- TEXT STYLES ----- */

/* h1 - Main Heading */
h1 {
    font-family: 'Arial', sans-serif;
    /* Sets font type */

    color: #54a9fd;
    /* Sets text color */

    font-size: 36px;
    /* Controls size of heading */

    text-align: center;
    /* Aligns text to center */
}

/* h2 - Subheading */
h2 {
    font-family: 'Verdana', sans-serif;
    color: #41e0c0;
    font-size: 28px;
    text-align: center;
}

```

```
}

/* h3 - Section Heading */
h3 {
    font-family: 'Georgia', serif;
    /* Serif font gives different style look */

    color: #ca65f5;
}

/* Paragraph Styling */
p {
    font-family: 'Tahoma', sans-serif;
    color: #8aaccf;
    font-size: 16px;

    line-height: 1.6;
    /* Controls spacing between lines for readability */
}

/* ----- SECTION STYLING ----- */

.styled-section {
    background-color: #ffcc00;
    /* Adds background color */

    border: 2px solid #ff8d2a;
    /* Adds border around section */

    padding: 20px;
    /* Space inside the section */

    margin: 20px 0;
    /* Space outside section (top & bottom only) */
}

/* ----- BOX MODEL DEMO ----- */

.box-model-demo {
    background-color: #ecf0f186;
    /* Light background */

    border: 2px dashed #48b6ff;
    /* Dashed border style */

    padding: 30px;
    /* Space inside div (content to border) */

    margin: 40px;
    /* Space outside div (between elements) */

    text-align: center;
    /* Centers text inside box */
}
```

```
}

/* ----- FORM STYLING ----- */

.signup-form {
  background-color: #bdc3c7;
  /* Background for form container */

  padding: 20px;
  /* Inner spacing */

  border-radius: 8px;
  /* Rounds the corners */

  max-width: 400px;
  /* Limits form width */

  margin: 0 auto;
  /* Centers form horizontally */
}

/* Label Styling */
.signup-form label {
  display: block;
  /* Makes label take full width */

  margin-top: 10px;
  /* Adds spacing above label */

  font-weight: bold;
  /* Makes text bold */
}

/* Input and Select Styling */
.signup-form input,
.signup-form select {
  width: 100%;
  /* Makes input full width of container */

  padding: 8px;
  /* Inner spacing */

  margin-top: 5px;
  /* Space above input */

  border: 1px solid #bebebe;
  /* Adds border */

  border-radius: 4px;
  /* Slight rounded corners */
}
```

```

/* Submit Button */
.signup-form button {
    margin-top: 15px;
    /* Space above button */

    padding: 10px 20px;
    /* Vertical & horizontal spacing inside button */

    background-color: #4bff96;
    /* Button background */

    color: white;
    /* Text color */

    border: none;
    /* Removes default border */

    border-radius: 4px;
    /* Rounded corners */

    cursor: pointer;
    /* Changes cursor to hand icon on hover */
}

/* Hover Effect */
.signup-form button:hover {
    background-color: #1c9950;
    /* Changes color when mouse hovers over button */
}

```

OUTPUT :

The screenshot displays a styled web page with the following features:

- Welcome to My Styled Page**: Main title in blue.
- Subheading with Styling**: Subtitle in green.
- A yellow **Section Title** box containing text about paragraph styling.
- A light gray **Div with Margin and Padding** box demonstrating layout.
- A **Sign Up Form** box containing input fields for Name, Email, Course, and a green **Submit** button.

Problem 2: Personal Profile Card (Level-1)

Scenario

You are building a simple personal profile card for a portfolio website.

The card will display a user's name, role, short bio, and contact button, styled purely using CSS.

📌 Requirements

Create a profile card layout containing:

- Name (heading)
- Role/title
- Short description paragraph
- Contact button

Apply:

- Text formatting (alignment, decoration)
- Font styles (font-family, size, weight)
- Typography enhancements (line-height, letter-spacing)

Use the CSS Box Model:

- Padding for spacing inside the card
- Border to define the card boundary
- Margin to separate the card from the page edges

Add:

- Background color
- Border radius
- Hover effect on the button

🛠️ Technical Constraints

- Use only HTML + CSS
- No images or icons
- No frameworks (Bootstrap, Tailwind, etc.)
- Inline CSS not allowed (use `<style>` or external CSS)

🎯 Learning Outcome

You should be able to:

- Understand how CSS styles text and fonts
- Apply box model concepts effectively
- Style elements using colors, borders, and spacing
- Improve visual hierarchy using typography

CODE:

```
<!DOCTYPE html>
<!-- Declares HTML5 document type --&gt;

&lt;html&gt;
<!-- Root element of the webpage --&gt;

&lt;head&gt;
    &lt;title&gt;Profile Card&lt;/title&gt;
    <!-- Title displayed in browser tab --&gt;

    &lt;style&gt;
        /* Internal CSS styling (used for small projects or demos) */

        body {
            background-color: #f4f4f4;
            /* Light gray background for page */

            font-family: Arial, sans-serif;
            /* Sets default font */

            margin: 0;
            /* Removes default browser margin */

            padding: 50px;
            /* Adds spacing around content */

            text-align: center;
            /* Centers inline content horizontally */
        }

        .card {
            background-color: white;
            /* Card background */

            width: 300px;
            /* Fixed width for card */

            margin: 0 auto;
            /* Centers card horizontally */

            padding: 25px;
            /* Space inside card */

            border: 2px solid #ddd;
        }
    &lt;/style&gt;
&lt;/head&gt;
&lt;body&gt;
    &lt;div class="card"&gt;
        &lt;h1&gt;John Doe&lt;/h1&gt;
        &lt;p&gt;Software Engineer at Google&lt;/p&gt;
        &lt;img alt="Profile picture of John Doe" src="profile-picture.jpg" style="width: 100%; height: auto;"/>
        <ul>
            <li>Age: 30</li>
            <li>Location: San Francisco, CA</li>
            <li>Skills: JavaScript, Python, Machine Learning</li>
        </ul>
    </div>
</body>
</html>
```

```
/* Light border around card */

border-radius: 12px;
/* Rounded corners */
}

.card h2 {
  margin-bottom: 5px;
  /* Space below name */

  font-size: 24px;
  /* Heading size */

  font-weight: bold;
  /* Bold text */

  letter-spacing: 1px;
  /* Adds spacing between letters */
}

.role {
  color: #555;
  /* Gray color for role */

  font-size: 16px;

  font-weight: 600;
  /* Semi-bold text */

  margin-bottom: 15px;
}

.bio {
  font-size: 14px;

  line-height: 1.6;
  /* Improves readability */

  margin-bottom: 20px;
}

.btn {
  display: inline-block;
  /* Allows padding & margin like block element */

  padding: 10px 20px;
  /* Button spacing */

  background-color: #007BFF;
  /* Blue button background */

  color: white;
  /* Text color */

  text-decoration: none;
}
```

```

/* Removes underline from link */

border-radius: 6px;
/* Rounded corners */

font-weight: bold;
}

.btn:hover {
    background-color: #0056b3;
    /* Darker blue on hover */
}
</style>
</head>

<body>
<!-- Contains visible webpage content --&gt;

&lt;div class="card"&gt;
    <!-- div used as container for profile card component --&gt;

    &lt;h2&gt;keerthivasan&lt;/h2&gt;
    <!-- Main name heading --&gt;

    &lt;div class="role"&gt;
        Frontend Developer
    &lt;/div&gt;
    <!-- Role description (separate div for styling control) --&gt;

    &lt;p class="bio"&gt;
        <!-- Paragraph for profile description --&gt;
        Passionate about creating clean and responsive web designs.
        Loves building modern UI experiences.
    &lt;/p&gt;

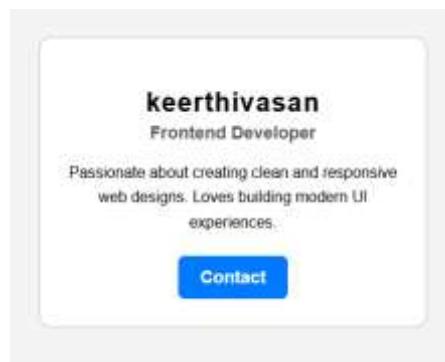
    &lt;a href="#" class="btn"&gt;
        Contact
    &lt;/a&gt;
    <!-- Anchor styled as button --&gt;
&lt;/div&gt;

&lt;/body&gt;
</pre>

```

</html>

OUTPUT:



Problem 3: Product Feature List Section (Level-1)

Scenario

You are designing a feature highlight section for a product landing page that lists key features in a visually appealing way.

📌 Requirements

Create a section with:

- Section heading
- List of 4–5 features

Style the list using:

- Custom fonts
- Text color and background color
- Border and padding for each feature item

Use:

- CSS selectors (element, class, descendant)

Apply hover effects on feature items

🛠️ Technical Constraints

- Use unordered list ()
- No JavaScript
- No layout frameworks
- Only basic HTML structure

🎯 Learning Outcome

You will be able to:

- Practice CSS selectors
- Style lists professionally
- Improve UI consistency

CODE:

```
<!DOCTYPE html>
<!-- Declares HTML5 document -->

<html>

<head>

    <!-- Makes website responsive on mobile devices -->
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Product Features</title>

<style>
    /* Body Styling */
    body {
        font-family: Verdana, sans-serif;
        /* Sets default font */

        background-color: #f4f6f9;
        /* Light background */

        margin: 0;
        /* Removes default browser margin */

        padding: 2rem;
        /* Adds spacing around page content */
    }

    /* Main Section Container */
    section {
        max-width: 900px;
        /* Limits content width for better readability */

        margin: auto;
        /* Centers section horizontally */
    }

    /* Heading Style */
    h2 {
        text-align: center;
        /* Centers heading */

        margin-bottom: 2rem;
        /* Space below heading */

        font-size: 2rem;
        /* Responsive font size using rem */

        letter-spacing: 1px;
        /* Adds spacing between letters */
    }

    /* Unordered List Layout */
    ul {
        list-style: none;
        /* Removes default bullets */

        padding: 0;
        margin: 0;

        display: flex;
        /* Enables flexbox layout */

        flex-wrap: wrap;
    }

```

```
/* Allows items to wrap to next line */

gap: 20px;
/* Space between flex items */
}

/* List Item Styling */
ul li {
    background-color: #ffffff;

    flex: 1 1 250px;
    /*
    flex-grow: 1 → Can grow
    flex-shrink: 1 → Can shrink
    flex-basis: 250px → Default width
    */

    padding: 20px;

    border: 2px solid #ddd;
    border-radius: 10px;
    color: #333;
    font-size: 1rem;
    line-height: 1.6;

    transition: 0.3s ease;
    /* Smooth animation effect */

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.05);
    /* Soft shadow effect */
}

/* Hover Effect */
ul li:hover {
    background-color: #007BFF;
    color: white;
    border-color: #007BFF;

    transform: translateY(-5px);
    /* Moves card slightly upward */

    /* Creates interactive feel */
}

/* ----- Responsive Design ----- */

/* Tablet View */
@media (max-width: 768px) {
    h2 {
```

```
        font-size: 1.6rem;
        /* Smaller heading for tablets */
    }
}

/* Mobile View */
@media (max-width: 500px) {

    body {
        padding: 1rem;
        /* Reduce outer spacing */
    }

    ul {
        flex-direction: column;
        /* Stack items vertically */
    }

    ul li {
        flex: 1 1 100%;
        /* Each item takes full width */
    }
}
</style>
</head>

<body>

<section>
    <!-- Groups related feature content --&gt;

    &lt;h2&gt;Product Features&lt;/h2&gt;

    &lt;ul&gt;
        <!-- List used to represent feature items --&gt;

        &lt;li&gt;Fast Performance with optimized loading speed.&lt;/li&gt;
        &lt;li&gt;Fully Responsive Design across all devices.&lt;/li&gt;
        &lt;li&gt;User-Friendly Interface with intuitive navigation.&lt;/li&gt;
        &lt;li&gt;Secure Authentication system with encryption.&lt;/li&gt;
        &lt;li&gt;Cloud Integration for seamless data access.&lt;/li&gt;
    &lt;/ul&gt;
&lt;/section&gt;

&lt;/body&gt;

&lt;/html&gt;</pre>
```

OUTPUT:

Product Features

Fast Performance with optimized loading speed.

Fully Responsive Design across all devices.

User-Friendly Interface with intuitive navigation.

Secure Authentication system with encryption.

Cloud Integration for seamless data access.

Problem 4: Responsive Blog Layout (Level-2)

Scenario

You are creating a blog homepage that displays articles in a modern responsive layout suitable for desktop, tablet, and mobile screens.

📌 Requirements

Create a page layout with:

- Header
- Main content area with multiple blog cards

Use:

- **Flexbox or CSS Grid** for layout
- Box model for spacing

Each blog card should contain:

- Title
- Short description
- Read More button

Apply:

- Background colors
- Borders
- Typography enhancements

Use **media queries** to:

- Display multiple columns on desktop

- Switch to single column on mobile

Technical Constraints

- Only HTML + CSS
- No CSS frameworks
- No fixed widths (use percentages or flexible units)

Learning Outcome

Learners should be able to:

- Understand Flexbox/Grid layouts
- Apply responsive design principles
- Write effective media queries
- Build real-world page layouts

CODE:

```
<!DOCTYPE html>
<!-- Declares HTML5 document type --&gt;

&lt;html&gt;
&lt;head&gt;

    &lt;!-- Makes layout responsive on mobile devices --&gt;
    &lt;meta name="viewport" content="width=device-width, initial-scale=1.0"&gt;

    &lt;title&gt;Responsive Blog&lt;/title&gt;

    &lt;style&gt;

        /* Body Styling */
        body {
            margin: 0;
            /* Removes default browser margin */

            font-family: Arial, sans-serif;
            /* Sets default font */

            background-color: #f4f4f4;
            /* Light background color */
        }

        /* Header Section */
        header {
            background-color: #333;
            /* Dark background */

            color: white;
            /* White text */

            padding: 1.5rem;
        }

        /* Main Content Area */
        .content {
            margin-top: 20px;
            padding: 0 10px;
        }

        /* Footer Section */
        footer {
            text-align: center;
            margin-top: 20px;
        }
    </style>

```

```
/* Space inside header */

    text-align: center;
    /* Centers text horizontally */
}

header h1 {
    margin: 0;
    /* Removes default margin */

    font-size: 2rem;
    /* Controls heading size */
}

/* Container for blog cards */
.container {
    display: flex;
    /* Enables flexbox layout */

    flex-wrap: wrap;
    /* Allows items to wrap to next row */

    gap: 1.5rem;
    /* Space between cards */

    padding: 2rem;
    /* Outer spacing */

    justify-content: center;
    /* Centers cards horizontally */
}

/* Individual Blog Card */
.card {
    background-color: white;

    padding: 1.5rem;

    border: 1px solid #ddd;

    border-radius: 10px;

    box-sizing: border-box;
    /* Includes padding and border inside width */

    flex: 1 1 300px;
    /*
    flex-grow: 1
    flex-shrink: 1
    flex-basis: 300px
    Automatically adjusts width
    */

    max-width: 400px;
    /* Prevents card from becoming too wide */
}
```

```
        transition: 0.3s;
        /* Smooth animation */
    }

/* Hover effect */
.card:hover {
    transform: translateY(-5px);
    /* Moves card slightly upward */
}

/* Card Heading */
.card h3 {
    margin-top: 0;
    /* Removes extra space at top */

    font-size: 1.3rem;
}

/* Card Paragraph */
.card p {
    line-height: 1.6;
    /* Improves readability */

    font-size: 0.95rem;
}

/* Button Styling */
.btn {
    display: inline-block;
    /* Allows padding and margin */

    padding: 0.5rem 1rem;

    background-color: #007BFF;

    color: white;

    text-decoration: none;
    /* Removes underline */

    border-radius: 5px;

    margin-top: 10px;
}

.btn:hover {
    background-color: #0056b3;
    /* Darker blue on hover */
}

/* ----- Responsive Design ----- */

/* Tablet View */
@media (max-width: 992px) {
```

```
        header h1 {
            font-size: 1.8rem;
            /* Slightly smaller heading */
        }
    }

    /* Mobile View */
    @media (max-width: 600px) {

        .container {
            padding: 1rem;
            /* Reduce spacing */
        }

        .card {
            flex: 1 1 100%;
            /* Full width cards */

            max-width: 100%;
        }

        header h1 {
            font-size: 1.5rem;
            /* Smaller heading for small screens */
        }
    }
}

</style>
</head>

<body>

    <!-- Header contains blog title -->
    <header>
        <h1>My Blog</h1>
    </header>

    <!-- Container holds multiple blog cards -->
    <div class="container">

        <!-- Blog Card 1 -->
        <div class="card">
            <h3>Article One</h3>
            <p>This is a short blog description explaining the topic in a clean and responsive layout.</p>
            <a href="#" class="btn">Read More</a>
        </div>

        <!-- Blog Card 2 -->
        <div class="card">
            <h3>Article Two</h3>
            <p>This article demonstrates how flexbox automatically adjusts to screen size.</p>
            <a href="#" class="btn">Read More</a>
        </div>
    </div>

```

```

<!-- Blog Card 3 -->
<div class="card">
    <h3>Article Three</h3>
    <p>Modern layouts should adapt smoothly across devices of all sizes.</p>
    <a href="#" class="btn">Read More</a>
</div>

</div>

</body>

</html>

```

OUTPUT:

The screenshot shows a web page titled "My Blog" with a dark header bar. Below the header, there are three cards arranged horizontally, each representing an article. The first card is titled "Article One" and contains a short description: "This is a short blog description explaining the topic in a clean and responsive layout." It has a "Read More" button. The second card is titled "Article Two" and contains the text: "This article demonstrates how flexbox automatically adjusts to screen size." It also has a "Read More" button. The third card is titled "Article Three" and contains the text: "Modern layouts should adapt smoothly across devices of all sizes." It has a "Read More" button. The cards have rounded corners and shadows, giving them a 3D effect.

Problem 5: Responsive Dashboard Layout (Level-2)

Scenario

You are designing a simple admin dashboard layout for a web application.

Requirements

Create a layout with:

- Sidebar navigation
- Main content area
- Info cards inside the main section

Use:

- display properties
- Flexbox or Grid for page structure

Style:

- Sidebar with background color
- Cards with borders, padding, and margins

Apply **media queries** to:

- Convert sidebar into top navigation on mobile
- Stack cards vertically on small screens

Technical Constraints

- HTML + CSS only
- No JavaScript
- No external UI libraries

Learning Outcome

You will be able to:

- Master page layouts using Flexbox/Grid
- Apply modern responsive web design techniques
- Understand how layout adapts across devices
- Build industry-relevant UI structures

CODE:

```
<!DOCTYPE html>
<!-- HTML5 document declaration -->

<html>
<head>

    <!-- Makes layout responsive on mobile devices -->
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Responsive Dashboard with Toggle</title>

    <style>

        /* ----- Base Styling ----- */

        body {
            margin: 0;
            /* Removes default browser margin */

            font-family: Arial, sans-serif;

            background-color: #f4f6f9;
        }

        /* ----- Toggle Button ----- */

        .menu-toggle {
```

```
    display: none;
    /* Hides the checkbox (used only for logic) */
}

.menu-icon {
    display: none;
    /* Hidden on desktop */

    background-color: #222;
    color: white;

    padding: 15px;

    font-size: 22px;

    cursor: pointer;
    /* Changes cursor to hand */
}

/* ----- Layout Container ----- */

.dashboard {
    display: flex;
    /* Flexbox layout for sidebar + main */

    min-height: 100vh;
    /* Full viewport height */
}

/* ----- Sidebar ----- */

.sidebar {
    background-color: #222;

    color: white;

    padding: 2rem 1.5rem;

    width: 240px;

    box-sizing: border-box;

    transition: 0.3s;
    /* Smooth slide animation */
}

.sidebar h2 {
    margin-top: 0;
}

.sidebar ul {
    list-style: none;
    /* Removes bullets */

    padding: 0;
```

```
}

.sidebar ul li {
    margin-bottom: 1rem;

    cursor: pointer;

    transition: 0.3s;
}

.sidebar ul li:hover {
    color: #ff7300;
    /* Highlight menu item on hover */
}

/* ----- Main Content ----- */

.main {
    flex: 1;
    /* Takes remaining space */

    padding: 2rem;

    display: flex;

    flex-wrap: wrap;
    /* Cards wrap on smaller screens */

    gap: 1.5rem;
}

.card {
    background-color: white;

    padding: 1.5rem;

    border-radius: 10px;

    border: 1px solid #ddd;

    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.05);

    flex: 1 1 250px;
    /* Responsive card width */
}

/* ----- Mobile Styles ----- */

@media (max-width: 768px) {

    .menu-icon {
        display: block;
        /* Show hamburger icon on mobile */
    }
}
```

```
.dashboard {
    flex-direction: column;
    /* Stack sidebar + main vertically */
}

.sidebar {
    position: absolute;

    top: 55px;

    left: -100%;
    /* Hide sidebar off screen */

    width: 200px;

    height: 100%;
}

/*
When checkbox is checked,
move sidebar into view
*/
.menu-toggle:checked ~ .dashboard .sidebar {
    left: 0;
}

.main {
    padding: 1rem;
}

.card {
    flex: 1 1 100%;
    /* Cards full width on mobile */
}

</style>
</head>

<body>

<!-- Hidden Checkbox (Controls Sidebar Visibility) -->
<input type="checkbox" id="menu-toggle" class="menu-toggle">

<!-- Label acts as toggle button -->
<label for="menu-toggle" class="menu-icon">
    &#9776; Menu
    <!-- Unicode hamburger icon -->
</label>

<!-- Main Dashboard Layout -->
<div class="dashboard">

    <!-- Sidebar Navigation -->
    <div class="sidebar">
```

```
<h2>Admin Panel</h2>

<ul>
    <li>Dashboard</li>
    <li>Users</li>
    <li>Reports</li>
    <li>Settings</li>
</ul>
</div>

<!-- Main Content Area --&gt;
&lt;div class="main"&gt;

    &lt;div class="card"&gt;Total Users: 1200&lt;/div&gt;
    &lt;div class="card"&gt;Revenue: $8,500&lt;/div&gt;
    &lt;div class="card"&gt;New Orders: 75&lt;/div&gt;
    &lt;div class="card"&gt;Active Sessions: 320&lt;/div&gt;

&lt;/div&gt;

&lt;/div&gt;

&lt;/body&gt;

&lt;/html&gt;</pre>
```

OUTPUT:

The screenshot shows the rendered HTML output. On the left is a dark sidebar with the title "Admin Panel" and a navigation menu with links for Dashboard, Users, Reports, and Settings. To the right of the sidebar are four white cards, each containing a statistic: "Total Users: 1200", "Revenue: \$8,500", "New Orders: 75", and "Active Sessions: 320".