MIS203 – LAB01-4 Review
Introduction to Web Development with HTML & CSS

Mert Biricik
mertbiricik@stu.khas.edu.tr
mert.biricik@stu.khas.edu.tr
A210 – BCI-FEAST Lab

2025/10/31
13:00

Contents

Lesson Overview

Lesson Title: Building and Styling Your First Web Pages
Duration: 1 hour 30 minutes
Core Topics: HTML for structure, CSS for styling, and basic page layout.
Objective: By the end of this lesson, students will be able to create a basic multi-section webpage, apply styles using internal CSS, and understand how to customize page layout and appearance.

Lesson Schedule

- **Part 1: HTML Fundamentals**
  - The basic structure of an HTML document.
  - Working with text: headings and paragraphs.
  - Adding images and hyperlinks.
- **Part 2: Styling with CSS**
  - Introduction to Cascading Style Sheets (CSS).
  - Inline vs. Internal CSS.
  - Core Concepts: Selectors (tag, class), Properties, and Values.
  - Styling links and their states (hover, visited).
- **Part 3: Page Layout and Structure**
  - Using the `<div>` tag as a container.
  - Introduction to the Box Model (margin, padding).
  - Creating a simple two-column layout.
- **Part 4: Conclusion & Q&A**
  - Recap of key concepts.
  - Brief mention of external CSS and semantic HTML as next steps.
  - Open discussion and questions.

1 HTML Fundamentals

1.1 The Anatomy of an HTML Page

Every HTML document follows a standard structure. Explain that HTML is a language of ***tags*** that tell the browser how to display content.

- `<!DOCTYPE html>`: Declares the document type.
- `<html>`: The root element that wraps all content.
- `<head>`: Contains meta-information about the page (like the title and styles) that is not displayed directly.
- `<title>`: Sets the title in the browser tab.
- `<body>`: Contains the visible content of the page.

```
<!DOCTYPE html>
<html>
<head>
    <title>My First Page</title>
</head>
<body>
    This is where my content goes.
</body>
</html>
```

Listing 1: Basic HTML Structure (from HTML HW-1)

1.2 Working with Text: Headings and Paragraphs

HTML provides tags to structure text semantically.

- ***Headings*** (`<h1>` to `<h6>`) are used for titles and subtitles, with `<h1>` being the most important.
- ***Paragraphs*** (`<p>`) are used for blocks of text.

1.3 Adding Images and Links

Web pages are interactive. We use tags to include media and link to other pages.

- The ***Image tag*** `<img>` is self-closing and requires the `src` (source) attribute. The `alt` attribute is crucial for accessibility.
- The ***Anchor tag*** `<a>` creates a hyperlink and requires the `href` (hypertext reference) attribute to specify the destination URL.

```
<!-- Adding an image with a specific size -->
<img src="my_image.jpg" alt="A description of the image" width="200">

<!-- Creating a link to a website -->
<a href="http://news.bbc.co.uk/">Click here for news</a>
```

Listing 2: Example of Image and Link

2 Styling with CSS

2.1 What is CSS?

Cascading Style Sheets (CSS) is the language we use to style an HTML document. It describes how HTML elements should be displayed. CSS separates the presentation from the content, making websites easier to maintain.

2.2 Methods of Applying CSS

1. **Inline CSS:** Styles are applied directly to an HTML element using the `style` attribute. This is good for quick, specific changes but not for site-wide styling.

```
<p style="color:blue; font-size:16px;">This paragraph is styled inline.</p>
```

2. **Internal CSS:** Styles are defined within a `<style>` tag inside the `<head>` section. This is ideal for styling a single page and is the focus of LAB2.

```
<head>
    <title>Internal CSS</title>
    <style>
        p {
            background-color: gray;
            margin: 10px;
        }
        div {
            color: white;
            background-color: #009900;
        }
    </style>
</head>
```

Listing 3: Internal CSS Example (from MIS203-LAB2)

3. **External CSS:** (Mention as best practice) Styles are defined in a separate `.css` file and linked from the HTML document. This is the most efficient way to style an entire website.

2.3 CSS Syntax: Selectors, Properties, and Values

A CSS rule consists of a ***selector*** and a ***declaration block***.

- The **selector** points to the HTML element you want to style.
- The **declaration block** contains one or more declarations separated by semicolons.
- Each **declaration** includes a CSS ***property*** name and a ***value***, separated by a colon.

2.4 Types of Selectors

Tag Selector Targets all elements of a specific type. Example: `p { color: black; }`

Class Selector Targets all elements with a specific `class` attribute. It starts with a period (`.`). This is reusable and very powerful.

```
<!-- The CSS in the <head> section -->
<style>
    .cities {
        background-color: black;
        color: white;
    }
    .green-background {
        background-color: green;
    }
</style>

<!-- The HTML in the <body> section -->
<div class="cities">
    <h2>London</h2>
</div>
<div class="cities green-background">
    <h2>Ankara</h2>
</div>
```

Listing 4: Using a Class Selector (from MIS203-LAB2 Exercise 3)

3 The Class Selector

The class selector is a fundamental component of CSS, enabling the application of a single set of styles to multiple, disparate HTML elements. Its primary advantage is reusability.

3.1 Defining a Class in CSS

A class is defined within the `<style>` block or an external stylesheet. The syntax requires a period (`.`) prefixed to a custom name. This name must not contain spaces and is case-sensitive. All style declarations for the class are placed within the subsequent declaration block.

```
/* This class creates a highlighted box */
.highlight-box {
    background-color: yellow;
    border: 1px solid black;
    padding: 10px;
}

/* This class styles text to be red and bold */
.important-text {
    color: red;
    font-weight: bold;
}
```

Listing 5: Defining two distinct classes in CSS

3.2 Applying a Class in HTML

To apply a defined class to an HTML element, the global `class` attribute is used. The value of this attribute must exactly match the name of the class defined in the CSS, but without the leading period.

```
<!-- Applying a class to a div element -->
<div class="highlight-box">
    <p>This entire div is contained within a highlighted box.</p>
</div>

<!-- Applying a class to a span element to style a part of a sentence -->
```

```
<p>This is standard text, but <span class="important-text">this part is important</
    span>.</p>
```

Listing 6: Applying the defined classes to HTML elements

3.3 The HTML-CSS Connection Mechanism

The browser engine parses both the HTML document and the CSS rules. When it encounters an HTML element with a `class` attribute, it searches the CSS rules for a matching class selector. If a match is found, the style declarations within that rule are rendered and applied to that specific element. This mechanism allows a single CSS rule (e.g., `.highlight-box`) to be applied to any number of HTML elements (`div`, `p`, `span`, etc.) across the document.

3.4 Combining Multiple Classes

An HTML element can be assigned multiple classes by listing them within the `class` attribute, separated by spaces. The element will inherit the styles from all specified classes. This is an efficient method for combining modular, single-purpose styles.

```
<div class="highlight-box important-text">
    This box is both highlighted and contains important, red, bold text.
</div>
```

Listing 7: Applying two classes to a single element

In this case, the `<div>` will acquire the `background-color`, `border`, and `padding` from `.highlight-box`, in addition to the `color` and `font-weight` from `.important-text`.

3.5 Styling Link States

Links can be styled based on their state using ***pseudo-classes***.

- `a:link` - a normal, unvisited link
- `a:visited` - a link the user has visited
- `a:hover` - a link when the user mouses over it

4 Page Layout and Structure

4.1 The Tag: A Generic Container

The tag is a block-level element used to group other HTML elements together. It doesn't have any inherent visual representation but is essential for applying CSS for layout purposes. In LAB4, elements like #header, #content, and #sidebar are all divs.

4.2 The Box Model: Margin and Padding

Every HTML element can be considered as a box. The CSS box model defines the layout and design of these elements.

- **Content**: The text, image, or other content inside the element.
- **Padding**: The transparent space around the content, inside the border.
- **Border**: A line that goes around the padding and content.
- **Margin**: The transparent space around the border, separating the element from other elements.

4.3 Creating a Simple Layout

We can create a multi-column layout using tags and CSS. The float property is a traditional way to achieve this.

```
#content {
  float: left;
  width: 520px; /* Main content area */
}

#sidebar {
  float: right;
  width: 180px; /* Sidebar area */
}

#footer {
  clear: both; /* Stops floating, ensuring footer is at the bottom */
}
```

Listing 8: Basic Two-Column Layout CSS (from MIS203-LAB4)

This code creates a layout where the main content and sidebar sit side-by-side. The clear: both; property on the footer is critical to ensure it appears below both floated columns.

5 Conclusion and Next Steps

5.1 Lesson Recap

- **HTML** provides the **structure** and **content** of a webpage.
- **CSS** provides the **style** and **layout**.
- We learned to use tags, classes, and IDs as **selectors** to apply styles.
- tags combined with CSS properties like float, margin, and padding allow us to build complex page layouts.

5.2 Topics for Further Learning

- **External Stylesheets:** The best practice for managing styles on large websites.
- **Semantic HTML5:** Using tags like `<header>`, `<nav>`, `<section>`, and `<footer>` instead of generic `<div>`s to give more meaning to your page structure.
- **Responsive Design:** Using techniques like Flexbox, Grid, and media queries to make your website look good on all devices (desktops, tablets, and phones).

6 References

Refsnes Data. (n.d.). *CSS tutorial*. W3Schools. Retrieved October 30, 2025, from `https://www.w3schools.com/css/`

Refsnes Data. (n.d.). *HTML tutorial*. W3Schools. Retrieved October 30, 2025, from `https://www.w3schools.com/html/`