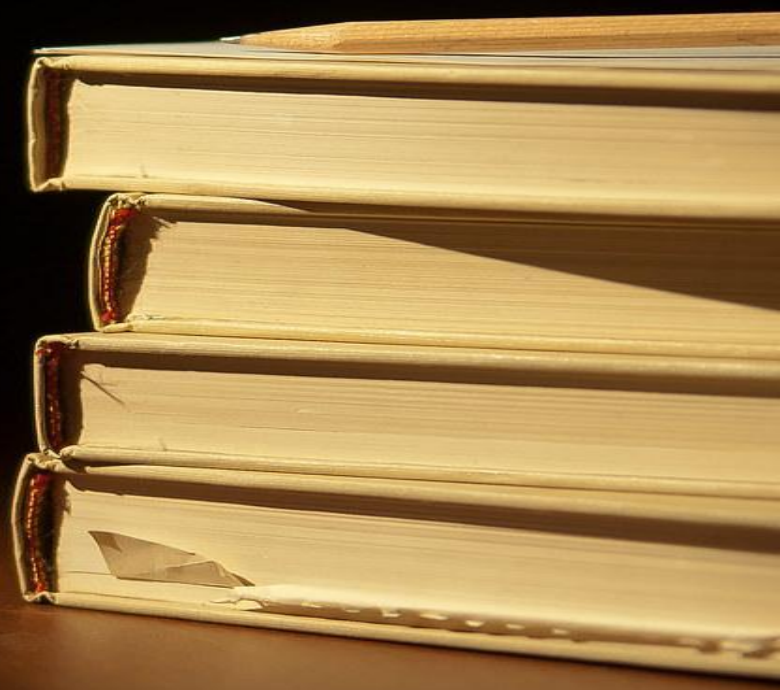
Welcome to lecture 5!



Agenda

Session Objectives

- Introduction to HTML
  - Understand the purpose and structure
  - Discover basic HTML tags
    - div, span, paragraph (p), headers & more
    - Semantic tags
    - Forms
    - Tables
  - Explore HTML attributes
  - Best Practices
- Q&A
- Homework (due next Sunday, March 20th by 5PM)

Introduction to HTML

Introduction to HTML

- HyperText Markup Language
  - Standard markup language for creating web pages
  - HTML is interpreted by web browsers to display content
  - Provides the layout and structure of a webpage (headers, paragraphs, links, and more)
- Markup Language
  - Unlike a programming language, HTML uses tags to "mark up" content, indicating its role (e.g., heading, paragraph, list)
- Why use HTML?
  - Structure: Organizes content into meaningful sections (headings, paragraphs, lists, etc.), making it easy to understand for both users and browsers
    - HTML uses tags to define elements, providing the building block for the sections
  - Semantic Meaning: Provides context to content, helping search engines and assistive technologies understand the purpose of different elements
  - Search Engine Optimization: Improves website ranking by providing search engines with clear information about the page's content

What’s the structure of a HTML?

- Basic structure of a HTML document
  - <!DOCTYPE html>
    - The document type declaration.
    - Tells the browser which HTML version is being used (HTML5 is recommended).
    - It is the very first thing in an HTML document.
  - <html>
    - The root element of the page.
    - Contains all other HTML elements (except <!DOCTYPE>).
  - <head>
    - Contains metadata (information *about* the HTML document), such as:
      - `<title>`: The title of the page (displayed in the browser tab).
      - `<meta>` tags: Information for browsers, search engines, etc.
      - `<link>` tags: Links to external resources (like CSS files)
  - <body>
    - Contains the visible page content (text, images, etc.)

Example: A basic HTML Page

```
<!DOCTYPE html>
<html>
<head>
  <title>My First Page</title>
  <meta charset="UTF-8">
  <meta name="description" content="A simple HTML page">
</head>
<body>
  <h1>Welcome Header!</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

Let’s Tag the HTML!

- What comes to mind when you hear – tags?
  - An example of tags are labels: price tag on a dress, hashtag on a post; they give us information
- What are tags in HTML?
  - Your Instructions to a web browser to display content on a webpage
- Heading Tags
  - <h1> to <h6> define headings of different levels
    - <h1> is the most important heading (main title)
    - <h6> is the least important heading
  - Best Practices
    - Use only one <h1> per page (usually the main title)
    - Use headings in the correct order (don't skip levels)
- Paragraph Tag
  - <p> defines a paragraph of text.
  - Browsers automatically add spacing before and after paragraphs.
  - Text within <p> tags is displayed as a block of text.

Example: Tags

```
<!DOCTYPE html>
<html>
<body>
  <h1>This is the main title</h1>
  <p>This is a paragraph of text.</p>
  <h2>This is a subheading</h2>
  <p>Another paragraph with more text.</p>
  <h3>A smaller heading</h3>
</body>
</html>
```

Uncovering HTML Tags: Continued

- Formatting Tags
  - <b> (Bold): Renders text in bold
  - (Strong): Indicates strong importance
    - Semantic meaning; screen readers may emphasize this text
    - Eg: "**Warning**! This is very dangerous."
  - <i> (Italic): Renders text in italic
  - (Emphasis): Indicates emphasis
    - Semantic meaning; screen readers may emphasize this text
    - Eg: "I *love* carrots" vs. "I love carrots"
  -
 (Line Break): Inserts a single line break
    - Useful for formatting addresses or poems
    - Is an empty tag; meaning, it doesn't have a closing tag
  - <hr> (Horizontal Rule): Creates a horizontal line
    - Used to separate content

Examples: Formatting Tags

```
<!DOCTYPE html>
<html>
<body>
  <p>This is <b>bold</b> text and this is <strong>important</strong> text.</p>
  <p>This is <i>italic</i> text and this is <em>emphasized</em> text.</p>
  <p>This is a line<br>with a line break.</p>
  <hr>
  <p>This is below the horizontal rule.</p>

</body>
</html>
```

Exercise: Create your own HTML Document

- Add the basic structure
  - <!DOCTYPE html>, <html>, <head>, <body>
- Inside <head>, add a <title>
- Inside <body>, add:
  - An <h1> heading tag with a header of your choice
  - A <p> paragraph with text of your choice
- Run it in your browser and verify it displays the content. W3C Link

Organizing Content with HTML

Lists in HTML

- HTML provides two main types of lists to structure information
  - Unordered Lists
    - Used to display a list of items where the order doesn't matter
    - Represented by the <ul> tag (unordered list)
    - Each item in the list is represented by the <li> tag (list item)
    - By default, unordered lists are displayed with bullet points
  - Ordered lists
    - Used to display a list of items where the order does matter
    - Represented by the <ol> tag (ordered list)
    - Each item in the list is also represented by the <li> tag
    - By default, ordered lists are displayed with numbers
- Lists can be nested within each other to create sub-lists
- Where have you seen ordered or unordered lists on the web?
  - Navigation menus, product listings, step-by-step instructions for a recipe, etc.

Example: Unordered vs Unordered List

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Lists</title>
</head>
<body>
  <h2>Unordered List of Fruits</h2>
  <ul>
    <li>Apple</li>
    <li>Banana</li>
    <li>Orange</li>
  </ul>
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Lists</title>
</head>
<body>
  <h2>Ordered List of Instructions</h2>
  <ol>
    <li>First step: Prepare ingredients</li>
    <li>Second step: Mix the batter</li>
    <li>Third step: Bake for 30 minutes</li>
  </ol>
</body>
</html>
```

Example: Nested Ordered List

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Lists</title>
</head>
<body>
  <h2>Nested List Example</h2>
  <ol>
    <li>Category 1
      <ul>
        <li>Item A</li>
        <li>Item B</li>
      </ul>
    </li>
    <li>Category 2</li>
  </ol>
</body>
</html>
```

Images in HTML

- The <img> tag is used to embed images in a web page
- It's an empty tag, meaning it has no closing tag
- Attributes of an <img> tag
  - src (source): Specifies the path or URL to the image file.
    - Can be an absolute URL (links to an image on another website) or a relative URL (links to an image within your website's files)
  - alt (alternative text): Provides a text description of the image
    - Crucial for accessibility: screen readers use the alt text to describe the image to visually impaired users
    - Important for SEO: search engines use alt text to understand the content of the image
    - Displayed if the image fails to load
  - width and height: Specify the dimensions of the image in pixels
    - It's best practice to size images correctly to avoid distortion and improve page load time
- The <img> tag supports the following formats: JPEG, PNG, GIF, SVG

Example: Images

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Images</title>
</head>
<body>
  <h1>My Profile Picture</h1>
  <img src="profile.jpg" alt="A photo of me" width="200" height="200">

  <p>This image is from an external source:</p>
  <img src="https://www.example.com/image.png" alt="An example image">
</body>
</html>
```

Links in HTML

- The <a> (anchor) tag is used to create hyperlinks, which connect one web page to another page or resource
- The <a> tag can link to:
  - Another web page on the same website
  - A web page on a different website
  - A specific section within the same page (using fragment identifiers)
  - Other resources, such as files to download
- Key Attributes
  - href (hypertext reference): Specifies the destination of the link
    - Can be a URL or a relative path
  - target: Specifies where to open the linked document
    - _self (default): Opens the link in the same tab/window
    - _blank: Opens the link in a new tab/window
- Links can be applied to text, images, or any other HTML element

Example: Links

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Links</title>
</head>
<body>
  <p>Visit <a href="https://www.example.com" target="_blank">Example Website</a> for more information.</p>
  <a href="about.html">Learn more about us</a>
  <a href="download.pdf">Download the document</a>
  <a href="#section3">Jump to Section 3</a>

  <img src="home.png" alt="Home" width="50" height="50" href="index.html">
</body>
</html>
```

Structuring Content & Introduction to Semantics

Let’s understand div vs span

- and are both container elements in HTML, but they serve different purposes:
  - (division): Is a block-level element
    - Creates a distinct block of content on the page
    - By default, it takes up the full width available, starting a new line before and after itself
    - Used for structuring the overall layout of a web page into sections (header, footer, main content, sidebar, etc.)
  - : Is an inline element
    - Used to group small pieces of inline content, such as text or images.
    - It only takes up as much width as its content requires and does not force line breaks
    - Used for styling or manipulating specific parts of text within a paragraph or other inline elements
- Key Differences
  - Layout: affects the overall page layout, while affects the content flow within a block
  - Width: expands to fill available width, wraps its content
  - Line Breaks: causes line breaks, does not

Example: Div & Span

```
<!DOCTYPE html>
<html>
<head>
  <title>Div and Span</title>
</head>
<body>
  <div>
    <h1>Welcome to My Website</h1>
    <p>This is the header section.</p>
  </div>

  <div>
    <p>This is the main content. It contains <span>important</span> information.</p>
    <p>Another paragraph in the content.</p>
  </div>

  <div>
    <p>&copy; 2024 My Website</p>
  </div>
</body>
</html>
```

Introduction to Semantic HTML

- What is Semantic HTML?
  - Semantic HTML uses HTML elements to reinforce the meaning of the content, rather than just its presentation.
  - It's about using the right tag for the right job.
  - Example: Using <article> to represent an article, <nav> for navigation, etc
- Why is Semantic HTML Important?
  - Accessibility: Screen readers rely on semantic tags to understand the structure and purpose of content, making it easier for visually impaired users to navigate
  - Search Engine Optimization: search engines use semantic tags to better understand the content and structure of a webpage, which can improve search engine rankings
  - Maintainability: semantic HTML makes code easier to read, understand, and maintain for developers
  - Interoperability: semantic HTML ensures that web pages are interpreted consistently across different browsers and devices

Learning Semantic Tags in HTML5

- <article>
  - Represents a self-contained composition in a document, page, application, or site.
  - Examples: a blog post, a news article, a forum post, a user comment.
  - Can be nested, where inner <article> elements represent content related to the outer article
- <section>
  - Represents a thematic grouping of content within a document.
  - Used to group related content within a page.
  - Examples: chapters, headings, or themed groups of content.
  - <section> is typically used with a heading
- <nav>
  - Represents a section of a page that provides navigation links.
  - Intended for major navigation blocks (e.g., primary navigation, table of contents).
  - Not all groups of links need to be in a <nav> element

Redo: div & span example with semantic tags

```html
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>Semantic HTML Example</title>
</head>
<body>
  <header>
    <h1>Welcome to My Website</h1>
    <p>This is the header section.</p>
  </header>

  <main>
    <p>This is the main content. It contains <span>important</span> information.</p>
    <p>Another paragraph in the content.</p>
  </main>

  <footer>
    <p>&copy; 2024 My Website</p>
  </footer>
</body>
</html>
```

Example: Semantic Tags

```
<!DOCTYPE html>
<html>
<head>
  <title>Semantic Tags</title>
</head>
<body>
  <nav>
    <ul>
      <li><a href="#">Home</a></li>
      <li><a href="#">About</a></li>
      <li><a href="#">Blog</a></li>
    </ul>
  </nav>

  <section>
    <h2>Introduction</h2>
    <p>This section introduces the topic.</p>
  </section>
```

```
  <article>
    <h2>My Blog Post</h2>
    <p>This is the content of my blog post.</p>
    <section>
      <h3>Comments</h3>
      <article>
        <p>Great post!</p>
      </article>
    </section>
  </article>
</body>
</html>
```

Learning Semantic Tags in HTML5 (Contd.)

- <header>
  - Represents introductory content for a section or page
  - Typically contains a heading, logo, navigation, or author information
  - Can be used multiple times in a document (e.g., for the main page header and for section headers)
- <footer>
  - Represents a footer for a section or page.
  - Typically contains information about the author, copyright, contact information, site map, or related documents.
  - Can also be used multiple times
- <aside>
  - Represents content that is tangentially related to the main content.
  - Often used for sidebars, callouts, or explanatory notes
- To understand more, review the documentation

Example: Semantic Tags

```
<!DOCTYPE html>
<html>
<head>
  <title>More Semantic Tags</title>
</head>
<body>
  <header>
    <h1>Welcome to My Blog</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">About</a></li>
      </ul>
    </nav>
  </header>
```

```
<main>
    <article>
      <h2>My Article Title</h2>
      <p>Main content of the article.</p>
      <aside>
        <p>Related information or sidebar content.</p>
      </aside>
    </article>
  </main>

  <footer>
    <p>&copy; 2024 My Blog</p>
    <p>Contact: <a
href="mailto:info@example.com">info@example.com</a></p>
  </footer>
</body>
</html>
```

Forms

Introduction to HTML Forms

- What are forms?
  - Forms are a fundamental part of web interaction, allowing users to submit data to a server
  - The <form> tag defines an HTML form
    - It acts as a container for various input elements
- Key attributes of the <form> element:
  - action: Specifies the URL where the form data should be sent when the form is submitted
    - This is typically a server-side script or application that will process the data
  - method: Specifies the HTTP method used to send the form data
    - get: Sends the data as part of the URL (visible in the address bar)
      - Should be used for short forms and non-sensitive data as it's insecure
    - post: Sends the data in the HTTP request body (not visible in the URL)
      - Should be used for longer forms and sensitive data (like passwords)

Example: Form

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Forms</title>
</head>
<body>
  <h2>Registration Form</h2>
  <form action="/submit-form" method="post">
    </form>
</body>
</html>
```

Capturing input in Forms

- As a user submitting forms, what are the ways you've entered information?
  - Text field, radio button, checkboxes and more!
- What's the <input> tag?
  - <input>: a versatile form element, used to create various input fields.
    - This is typically a server-side script or application that will process the data
  - type attribute: Specifies the type of input control to display.
    - text: A single-line text input field.
      - Used for names, usernames, etc.
    - password: A password input field.
      - Masks the entered characters for security.
    - email: An input field for email addresses.
      - Often includes basic validation to ensure the input has a valid email format

Example: Form Input

```
<!DOCTYPE html>
<html>
<head>
  <title>Form Inputs</title>
</head>
<body>
  <form action="/submit-form" method="post">
    <label for="username">Username:</label><br>
    <input type="text" id="username" name="username"><br><br>

    <label for="password">Password:</label><br>
    <input type="password" id="password" name="password"><br><br>

    <label for="email">Email:</label><br>
    <input type="email" id="email" name="email"><br><br>

    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Form Input Elements (Continued)

- type attribute: Specifies the type of input control to display
  - radio: Radio buttons
    - Allow the user to select one option from a group
    - Radio buttons in a group must have the same name attribute to work correctly
  - checkbox: Checkboxes
    - Allow the user to select multiple options from a group
    - submit: A button that submits the form data to the server
  - <textarea>: A multi-line text input area
    - Used for comments, messages, etc
  - <select> and <option>: A drop down list
    - <select> defines the dropdown list
    - <option> defines an option within the list

Example: Form Input Attributes

```
<!DOCTYPE html>
<html>
<head>
  <title>More Form Inputs</title>
</head>
<body>
  <form action="/submit-form" method="post">
    <label>Gender:</label><br>
    <input type="radio" id="male" name="gender"
value="male">
    <label for="male">Male</label>
    <input type="radio" id="female" name="gender"
value="female">
    <label for="female">Female</label><br><br>

    <label>I agree to the terms:</label>
    <input type="checkbox" id="agree" name="agree"
value="agree">
    <label for="agree">Agree</label><br><br>
```

```
<textarea id="message" name="message" rows="4"
cols="50"></textarea><br><br>

    <label for="cars">Choose a car:</label>
    <select id="cars" name="cars">
      <option value="volvo">Volvo</option>
      <option value="saab">Saab</option>
      <option value="fiat">Fiat</option>
      <option value="audi">Audi</option>
    </select><br><br>

    <input type="submit" value="Send">
  </form>
</body>
</html>
```

Form Input Attributes

- Attributes provide additional information and control over input elements?
  - Common attributes
    - name: Specifies the name of the input element
      - Used to identify the input when the form data is submitted.
      - Essential for server-side processing
    - id: a unique identifier for the input element
      - Used for associating labels, styling with CSS, and accessing the element with JavaScript
    - value: Specifies the initial value of the input element
      - For text inputs, it's the default text.
      - For radio buttons and checkboxes, it's the value sent to the server when selected
    - placeholder: Provides a hint to the user about what kind of data to enter
      - The placeholder text disappears when the user starts typing
    - required: Specifies that the input field must be filled out before the form can be submitted
  - <label>: Provides a user-friendly caption for form elements
    - The for attribute of the <label> must match the id of the associated input element.
    - Improves accessibility by allowing users to click the label to focus on the input

Example: Form Input Attributes

```
<!DOCTYPE html>
<html>
<head>
  <title>Form Attributes</title>
</head>
<body>
  <form action="/submit-form" method="post">
    <label for="username">Username:</label><br>
    <input type="text" id="username" name="username" placeholder="Enter your username" required><br><br>

    <label for="password">Password:</label><br>
    <input type="password" id="password" name="password" value="mySecret" required><br><br>

    <input type="submit" value="Register">
  </form>
</body>
</html>
```

Introduction to HTML Tables

Introduction to HTML Tables

- When navigating the web, what examples of tables have you seen?
  - Product comparisons, Stats (sports, movies, weather), etc
- Tables in HTML
  - HTML tables are used to display data in a structured format of rows and columns
- Key HTML tags for creating tables
  - <table>: Defines the overall table structure.
  - <tr> (table row): Defines a row in the table.
  - <th> (table header): Defines a header cell in a table row (typically used for column headings).
  - <td> (table data): Defines a standard data cell in a table row.
  - <caption>: Defines a caption or title for the table.

Example: Table

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML Tables</title>
</head>
<body>
  <h2>Student Grades</h2>
  <table>
    <caption>Student Grades</caption>
    <tr>
      <th>Name</th>
      <th>Subject</th>
      <th>Grade</th>
    </tr>
```

```
    <tr>
      <td>Alice</td>
      <td>Math</td>
      <td>A</td>
    </tr>
    <tr>
      <td>Bob</td>
      <td>Science</td>
      <td>B</td>
    </tr>
  </table>
</body>
</html>
```

Table Structure

- HTML provides additional tags to structure the content within a table, improving its semantics and organization
  - <thead>: Groups the header content in a table (typically containing <th> elements).
  - <tbody>: Groups the main body content in a table (typically containing <td> elements).
  - <tfoot>: Groups the footer content in a table (e.g., summaries or totals).
- These tags are optional but recommended for better table structure and styling (especially when using CSS)
- The colspan and rowspan attributes can be used to make a cell span across multiple columns or rows

Example: Table

```
<!DOCTYPE html>
<html>
<head>
  <title>Structured Tables</title>
</head>
<body>
  <h2>Sales Summary</h2>
  <table>
    <caption>Monthly Sales</caption>
    <thead>
      <tr>
        <th>Product</th>
        <th>Jan</th>
        <th>Feb</th>
        <th>Mar</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>Apples</td>
        <td>100</td>
        <td>120</td>
        <td>110</td>
      </tr>
      <tr>
        <td>Bananas</td>
        <td>150</td>
        <td>130</td>
        <td>140</td>
      </tr>
    </tbody>
    <tfoot>
      <tr>
        <td>Total</td>
        <td>250</td>
        <td>250</td>
        <td>250</td>
      </tr>
    </tfoot>
  </table>
</body>
</html>
```

Homework

- Link to homework exercises
- Many of you haven't starred the Github Repository. For future access, make sure to star it!
    - How to start a Github repository? Note: you'll have to sign up to do so

References

- W3C - beginner friendly
- Mozilla Documentation - great for reference
- FreeCodeCamp