LIVE STREAM

Understanding HTML

An Introduction to the Fundamental Language of the Web

Tim Burner Lee

Active Live Classes

PRAGYA INSTITUTE OF IT & RESEARCH ®

Tim Berners-Lee: A Brief Note

Tim Berners-Lee is a renowned computer scientist best known for inventing the World Wide Web. Born in London in 1955, he graduated from the University of Oxford with a degree in physics. In 1989, while working at CERN, the European Organization for Nuclear Research, Berners-Lee proposed a system for sharing information over the internet, which led to the development of the first web browser and server software the following year. His creation revolutionized communication and accessibility of information, making the internet an integral part of modern life. Berners-Lee continues to advocate for an open and accessible web and is recognized as a pivotal figure in the history of technology.

What is HTML?

HTML, which stands for **HyperText Markup Language**, is the standard language used to create and design documents on the World Wide Web. It forms the backbone of all web pages and is used to structure content in a meaningful way.

Key Features of HTML

- **Markup Language**: HTML is a markup language, not a programming language. It uses tags to annotate text, images, and other content for display in a web browser.
- **Structure of Web Pages**: HTML provides the basic structure for web pages by using a series of elements represented by tags, such as <html>, <head>, <title>, <body>, <h1> through <h6> for headers, and <p> for paragraphs.
- **Hyperlinks**: One of the most powerful features of HTML is the ability to create hyperlinks using the <a> tag, allowing users to navigate between web pages easily.

- **Forms and Input**: HTML supports the creation of forms and user input elements like buttons, text fields, and checkboxes using tags like <form>, <input>, <button>, and others.

Basic Structure of an HTML Document

A simple HTML document typically includes the following basic structure:

```
<!DOCTYPE html>
<html>
 <head>
  <title>Page Title</title>
 </head>
 <body>
  <h1>This is a Heading</h1>
  <p>This is a paragraph.</p>
  <a href="https://example.com">This is a link</a>
 </body>
</html>
```

- **<!DOCTYPE html>**: Declares the document type and version of HTML.
- **<html>**: The root element of an HTML page.
- **<head>**: Contains meta-information about the document, such as its title.
- **<title>**: Specifies the title of the web page, displayed in the browser's title bar or tab.
- **<body>**: Contains the content of the document, such as text, images, and links.

HTML Versions: A Journey Through Time

HTML has evolved significantly since its inception, with each version introducing new features and improvements to enhance web development capabilities. Here's a brief overview of the major versions of HTML and their contributions to the web:

HTML 1.0

- **Release Date: 1993**
- **Overview: The first official version of HTML, HTML 1.0, was a simple language for creating basic web pages. It supported rudimentary elements like text, hyperlinks, and headers but lacked sophisticated features.**

HTML 2.0

- **Release Date: 1995**
- **Overview: HTML 2.0 standardized the language further, adding support for forms, tables, and more complex document structures. It laid the groundwork for future enhancements and became the basis for web interactivity.**

HTML 3.2

- **Release Date: 1997**
- **Overview: With the introduction of HTML 3.2, web developers gained access to new elements for creating interactive and visually appealing content, such as applets, improved tables, and support for image maps.**

HTML 4.01

- **Release Date: 1999**
- **Overview: HTML 4.01 was a significant milestone, introducing concepts like CSS for styling and JavaScript for client-side scripting. It also emphasized separating content from presentation, promoting a more structured web development approach.**

XHTML 1.0

- **Release Date: 2000**
- **Overview: XHTML 1.0 aimed to bring the rigor of XML to HTML, enforcing stricter syntax rules. It encouraged cleaner, more consistent coding practices and was designed to be more extensible than previous versions.**

HTML5

- **Release Date: 2014**
- **Overview: HTML5 represents the most comprehensive update to the language, focusing on improving multimedia support, semantic elements, and application capabilities. It introduced native audio and video, canvas for graphics, and APIs for offline storage and geolocation, among other features.**

Importance of HTML

HTML is essential for web development as it:

- Defines the structure and layout of a web page.
- Is the foundation of web content that can be styled with CSS (Cascading Style Sheets) and made interactive with JavaScript.
- Is universally supported by all web browsers, ensuring consistent display across different platforms and devices.

In conclusion, HTML is a pivotal technology for the internet, allowing for the creation and sharing of content in a standardized and accessible way. Understanding HTML is fundamental for anyone interested in web design or development.

HTML Tags

HTML tags are the building blocks of HTML, used to create elements on a web page. They consist of a keyword enclosed in angle brackets, such as <tagname>. Most HTML tags come in pairs, with an opening tag and a closing tag, like and , to define the start and end of an element. Here are some commonly used HTML tags and their purposes:

Common HTML Tags

1. **Heading Tags (<h1> to <h6>)**
   - **Define headings on a page. <h1> is the highest level (largest) and <h6> is the lowest level (smallest).**
2. **Paragraph Tag ( )**
   - **Used to define a block of text as a paragraph.**
3. **Anchor Tag (<a>)**
   - **Creates hyperlinks, allowing users to navigate to other pages or sections. Example: Visit Example.**
4. **Image Tag (<img>)**
   - **Embeds images into a web page. Requires the src attribute to specify the image source. Example: <img src="image.jpg" alt="Description of Image">.**
5. **List Tags**
   - **Ordered List (<ol>): Creates a numbered list.**
   - **Unordered List (<ul>): Creates a bulleted list.**
   - **List Item (<li>): Defines items within both <ol> and <ul> lists.**
6. **Table Tags**
   - **Table (<table>): Contains the entire table.**
   - **Table Row (<tr>): Defines a row in the table.**
   - **Table Header (<th>): Specifies a header cell in a table.**
   - **Table Data (<td>): Specifies a standard cell in a table.**
7. **Division Tag ( )**
   - **Used to group block-level elements for styling and layout purposes.**
8. **Span Tag ()**
   - **Used to group inline elements for styling purposes.**
9. **Form Tag (<form>)**
   - **Creates a form for user input. Works with input elements like <input>, <textarea>, <button>, etc.**
10. **Bold Tag ()**
    - **Makes text bold, indicating importance.**
11. **Italic Tag ()**
    - **Emphasizes text, typically rendering it in italics.**

Attributes in HTML Tags

HTML tags often include attributes, which provide additional information about elements. Attributes are placed within the opening tag and typically come in name/value pairs, like

name="value". For example, in <img src="image.jpg" alt="Description of Image">, src and alt are attributes of the <img> tag.

Advanced HTML Tags

1. Multimedia Tags

HTML provides several tags for incorporating multimedia elements, allowing for richer web pages.

- : Embeds sound content, such as music or podcasts. Supports multiple formats using the <source> tag.
  - Example:
  - <audio controls>
    <source src="audio.mp3" type="audio/mpeg">
    Your browser does not support the audio element.
    </audio>
- : Embeds video content. Like <audio>, it supports multiple formats with <source>.
  - Example:
  - <video width="320" height="240" controls>
    <source src="movie.mp4" type="video/mp4">
    Your browser does not support the video tag.
    </video>
- : Used to draw graphics on the fly via scripting (usually JavaScript).
  - Example:
  - <canvas id="myCanvas" width="200" height="100" style="border:1px solid #000000;">
    </canvas>

Understanding the <pre> Tag in HTML

The <pre> tag is a unique and useful HTML element that stands for "preformatted text." It is designed to display text exactly as it is typed in the HTML document, preserving all spaces and line breaks. This can be particularly helpful when you want to display code snippets, poetry, or any text where the format and spacing are important.

Usage Examples

Displaying Code

The <pre> tag is often used in conjunction with the <code> tag to display programming code snippets, as it maintains the code's formatting.

<pre>
<code>

```
function greet(name) {
  console.log("Hello, " + name + "!");
}
</code>
</pre>
```

Formatting Text with Consistent Spacing

If you need to display a poem or structured text with specific spacing, the <pre> tag ensures the format is maintained.

```
<pre>
Roses are red,
  Violets are blue,
Sugar is sweet,
  And so are you.
</pre>
```

Understanding the <table> Tag in HTML

The <table> tag in HTML is used to create tables, which are essential for displaying data in a structured, grid-like format. Tables are highly versatile and can be used to present information clearly and concisely.

Basic Structure of a Table

A simple HTML table is composed of several key elements:

```
<table>
 <thead>
  <tr>
   <th>Header 1</th>
   <th>Header 2</th>
  </tr>
 </thead>
 <tbody>
  <tr>
   <td>Data 1</td>
   <td>Data 2</td>
  </tr>
  <tr>
   <td>Data 3</td>
```

```
    <td>Data 4</td>
   </tr>
  </tbody>
</table>
```

Important Table Elements

1. **<table>**
   - The <table> tag is the container for all the table elements. It defines the start and end of the table.
2. **<thead>**
   - The <thead> element groups the table header content, usually containing one or more <tr> tags with <th> elements inside.
3. **<tbody>**
   - The <tbody> element contains the main body of the table, grouping rows of data represented with <tr> tags and <td> cells.
4. **<tr>**
   - The <tr> tag defines a table row. It's used within <thead>, <tbody>, and <tfoot> elements.
5. **<th>**
   - The <th> tag defines a header cell in a table, typically containing labels for each column or row. Text within <th> is usually bold and centered.
6. **<td>**
   - The <td> tag defines a standard data cell in a table, used to hold data values.
7. **<tfoot>**
   - The <tfoot> element is used to group footer content in a table, often containing summary information.

Attributes in Table Tags

- **border**
  - The border attribute specifies the width of the border around the table. While CSS is preferred for styling, this attribute can be used for quick styling.
- **cellpadding and cellspacing**
  - cellpadding defines the space between the cell content and its border, while cellspacing defines the space between cells. These are also more commonly controlled using CSS now.
- **colspan and rowspan**
  - colspan allows a cell to span multiple columns. Similarly, rowspan allows a cell to span multiple rows.

List Tags in HTML

HTML provides tags to create lists, which help organize content in a structured and readable format. There are several types of lists you can create:

1. **Ordered List (<ol>)**

- Creates a numbered list. Each item is automatically numbered in ascending order.
- Example:
-
  First item
  Second item
  Third item

2. **Unordered List (<ul>)**
   - Creates a bulleted list, where each item is preceded by a bullet point.
   - Example:
   -
     First item
     Second item
     Third item

3. **List Item (<li>)**
   - Defines an item in either an ordered or unordered list. Each <li> element represents a single entry in the list.
   - Example:
   -
     Apple
     Banana
     Cherry

4. **Description List (<dl>)**
   - Creates a list of terms and their descriptions.
   - Example:
   - <dl>
     <dt>HTML</dt>
     <dd>HyperText Markup Language, the standard language for creating web pages.
     </dd>
     <dt>CSS</dt>
     <dd>Cascading Style Sheets, used for styling web pages.</dd>
     </dl>
5. **Description Term (<dt>)**
   - Specifies a term or name in a description list.
   - Example:
   - <dl>
     <dt>Term</dt>
     <dd>Definition of the term.</dd>
     </dl>
6. **Description Definition (<dd>)**
   - Provides the description or definition of a term in a description list.
   - Example:
   - <dl>
     <dt>Browser</dt>

<dd>A software application for accessing information on the World Wide Web.</dd>
</dl>

Understanding the <marquee> Tag in HTML

The <marquee> tag is an HTML element used to create a scrolling piece of text or image across a webpage. It was introduced in early versions of HTML but is now deprecated in HTML5 due to accessibility and usability concerns. Instead, CSS animations and JavaScript are recommended for creating scrolling effects.

Usage of the <marquee> Tag

Despite its deprecation, the <marquee> tag is still supported by most browsers and can be used for simple scrolling text or images. Here is a basic example of how the <marquee> tag works:

<marquee>Scrolling Text Example</marquee>

Attributes of the <marquee> Tag

The <marquee> tag comes with several attributes that allow you to customize its behavior:

- **direction**: Specifies the direction of the scroll. Options include left, right, up, and down. The default is left.
  - Example: <marquee direction="right">Text scrolling to the right</marquee>
- **behavior**: Defines the scrolling behavior, such as scroll, slide, or alternate.
  - Example: <marquee behavior="alternate">Text moving back and forth</marquee>
- **scrollamount**: Sets the speed of the marquee, where a higher number means faster scrolling.
  - Example: <marquee scrollamount="10">Faster scrolling text</marquee>
- **scrolldelay**: Specifies the delay between each scroll movement, measured in milliseconds.
  - Example: <marquee scrolldelay="100">Slower scrolling text</marquee>
- **loop**: Determines how many times the marquee will loop before stopping. A value of -1 means infinite looping.
  - Example: <marquee loop="3">Text scrolling three times</marquee>
- **bgcolor**: Sets the background color of the marquee.
  - Example: <marquee bgcolor="#f0f0f0">Text with a background color</marquee>

Understanding the Tag in HTML

The tag in HTML was traditionally used to define the font size, color, and face for text on a web page. However, it has been deprecated in HTML4 and is not supported in HTML5. Modern web development practices recommend using CSS for styling text instead.

Usage of the Tag

Despite its deprecation, you may still encounter the tag in older HTML documents. Here's how it was typically used:

This is a sample text.

Attributes of the Tag

The tag included several attributes that allowed customization of text appearance:

- **size**: Specifies the size of the text. Values could be from 1 to 7, or relative sizes such as +1 or -1.
  - Example: Larger text
- **color**: Sets the color of the text, using either a named color, a hexadecimal value, or an RGB value.
  - Example: Red text
- **face**: Defines the font face (or type) for the text, such as Arial, Times New Roman, etc.
  - Example: Text in Verdana font

Understanding the <hr> Tag in HTML

The <hr> tag in HTML is a semantic element used to represent a thematic break or horizontal rule in a web page. It is an empty tag, meaning it does not require a closing tag. Historically, it was used primarily for creating visual breaks between sections of content.

Usage of the <hr> Tag

The <hr> tag is typically used to separate content, indicating a shift in topic or a break in the flow of the document. It is a simple yet effective way to improve the readability and organization of a web page.

Example:

<h1>Main Heading</h1>
This is some introductory text.
<hr>
<h2>Subheading</h2>
This is a new section of content.

Attributes of the <hr> Tag

The <hr> tag, while traditionally used to create a horizontal line, can be customized using various attributes and CSS for more specific design needs. Although HTML attributes for styling are limited, CSS provides more flexibility.

HTML Attributes

Historically, the <hr> tag could be styled using a few attributes, though these are not part of the HTML5 specification and it's better to use CSS.

- **align: Specifies the alignment of the <hr>. Acceptable values are left, center, and right.**
  - **Example: <hr align="center">**
- **size: Sets the thickness of the line.**
  - **Example: <hr size="5">**
- **width: Defines the length of the line. This can be given as a percentage or in pixels.**
  - **Example: <hr width="50%">**
- **noshade: Removes the default shading, making the line solid.**
  - **Example: <hr noshade>**

Example of Styling with CSS

To achieve more modern and aesthetically pleasing designs, CSS is the recommended way to style the <hr> tag:

```
<!DOCTYPE html>
<html>
<head>
 <style>
  hr {
   border: none;
   height: 2px;
   background-color: #333;
   margin: 20px 0;
  }
 </style>
</head>
<body>
 <h1>Content Heading</h1>
 <p>This is an example of using the <hr> tag to separate sections.</p>
 <hr>
 <h2>Subsection Heading</h2>
 <p>More content follows here, divided by a styled horizontal rule.</p>
</body>
</html>
```

Text Formatting Tags in HTML

HTML provides various tags to format text to enhance readability and emphasize important content. These tags can be used to change the appearance of text, such as making it bold,

italicized, or underlined. Understanding these tags is essential for creating visually appealing web pages.

Common Text Formatting Tags

1. **Bold Tag ()**
   - Purpose: Used to make text bold, typically for highlighting importance.
   - Example:
   - This text is important.
2. **Italic Tag ()**
   - Purpose: Emphasizes text, usually rendering it in italics. It is often used to stress a word or phrase.
   - Example:
   - This text is emphasized.
3. **Underline Tag ()**
   - Purpose: Underlines the text. This tag is less commonly used in modern web design due to CSS alternatives.
   - Example:
   - <u>This text is underlined.</u>
4. **Strikethrough Tag ()**
   - Purpose: Strikes through text to indicate that it is incorrect or no longer relevant.
   - Example:
   - <s>This text is struck through.</s>
5. **Superscript Tag ()**
   - Purpose: Displays text as superscript, which is slightly above the baseline. Commonly used in mathematical expressions or chemical formulas.
   - Example:
   - H²O
6. **Subscript Tag ()**
   - Purpose: Displays text as subscript, which is slightly below the baseline. Often used in chemical formulas or mathematical expressions.
   - Example:
   - CO₂
7. **Code Tag ()**
   - Purpose: Displays text in a monospaced font, typically used for presenting programming code.
   - Example:
   - <code>console.log('Hello, world!');</code>
8. **Blockquote Tag ()**
   - Purpose: Used for quoting long sections of text from another source. It usually indents the quoted text.
   - Example:
   - <blockquote>
     "To be, or not to be, that is the question."
     </blockquote>
9. **Preformatted Text Tag ()**

- Purpose: Preserves the formatting of text, including spaces and line breaks. Often used for displaying code blocks.
- Example:
- <pre>
  function greet(name) {
    console.log("Hello, " + name + "!");
  }
  </pre>

10. Highlight Tag ()
    - Purpose: Highlights text, usually with a yellow background, to draw attention to it.
    - Example:
    - <mark>This text is highlighted.</mark>

Understanding the <form> Tag in HTML

The <form> tag is a fundamental part of HTML, used to collect user input and interact with web servers. It is the starting point for creating forms that allow users to submit data, whether it's for signing up for a newsletter, entering search queries, or submitting feedback.

Basic Structure of a Form

A basic HTML form is defined using the <form> tag and can contain various input elements, such as text fields, checkboxes, and buttons. Here is a simple example of an HTML form:

```
<form action="/submit-form" method="post">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name"><br><br>

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

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

Key Attributes of the <form> Tag

- action: Specifies the URL where the form data should be sent when submitted. This can be a server-side script or endpoint.
  - Example: <form action="/submit-form" method="post">
- method: Defines the HTTP method used to send form data. Common values are get and post.
  - get: Appends form data to the URL, suitable for non-sensitive data.
  - post: Sends form data as a separate HTTP request, recommended for sensitive or large amounts of data.

Common Input Elements within a Form

1. **Text Input:** Allows users to enter text.
   - Example: <input type="text" name="username">
2. **Password Input:** Masks input characters for sensitive information.
   - Example: <input type="password" name="password">
3. **Radio Buttons:** Lets users select one option from a set.
   - Example:
   - <input type="radio" name="gender" value="male"> Male
     <input type="radio" name="gender" value="female"> Female
4. **Checkboxes:** Allows multiple selections from a set of options.
   - Example:
   - <input type="checkbox" name="subscribe" value="newsletter"> Subscribe to newsletter
5. **Submit Button:** Submits the form data.
   - Example: <input type="submit" value="Submit">
6. **Select Dropdown:** Provides a dropdown menu for selecting one option.
   - Example:
   - <select name="country">
     <option value="usa">USA</option>
     <option value="canada">Canada</option>
     </select>
7. **Textarea:** Allows for multi-line text input.
   - Example: <textarea name="message" rows="4" cols="50"></textarea>

Enhancing Forms with Attributes

- name: Identifies form elements, crucial for accessing form data on the server-side.
  - Example: <input type="text" name="username">
- id: Provides a unique identifier for form elements, helpful in JavaScript manipulation and styling with CSS.
  - Example: <input type="text" id="username">
- placeholder: Offers a hint to the user about what to enter in the form field.
  - Example: <input type="text" placeholder="Enter your name">
- required: Ensures users do not skip mandatory fields.
  - Example: <input type="email" required>

Understanding the <select> Tag in HTML

The <select> tag in HTML is used to create a drop-down list, allowing users to choose one option from a predefined set. This element is commonly used in forms to present a list of options in a compact and user-friendly manner.

Basic Structure of a <select> Element

A <select> element is created using the <select> tag, which contains multiple <option> tags. Each <option> tag represents an item in the drop-down list.

Example

```
<form>
  <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>
</form>
```

In this example, a user can select a car from the list of options: Volvo, Saab, Fiat, or Audi.

Key Attributes of the <select> Tag

- **name: Specifies the name of the drop-down list, which is used to identify the selected value upon form submission.**
  - ***Example*: <select name="cars">**
- **id: Provides a unique identifier for the drop-down list, useful for JavaScript manipulation and CSS styling.**
  - ***Example*: <select id="cars">**
- **multiple: Allows the user to select more than one option from the list. When using this attribute, the list will appear as a multiple-selection box.**
  - ***Example*: <select name="cars" multiple>**
- **size: Sets the number of visible options in the list. This attribute is useful when multiple is enabled, as it defines how many options are visible at once without scrolling.**
  - ***Example*: <select name="cars" size="3">**

Key Attributes of the <option> Tag

- **value: Specifies the value that will be sent to the server when the option is selected. If omitted, the text between the opening and closing <option> tags will be the value.**
  - ***Example*: <option value="volvo">Volvo</option>**
- **selected: Pre-selects an option when the page loads. This attribute can be useful for setting default choices.**
  - ***Example*: <option value="fiat" selected>Fiat</option>**

Enhancing the <select> Element

Grouping Options with <optgroup>

The <optgroup> tag is used to group related options within a <select> element, improving the organization and readability of the list.

```
<select name="cars">
 <optgroup label="Swedish Cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
 </optgroup>
 <optgroup label="German Cars">
  <option value="audi">Audi</option>
  <option value="mercedes">Mercedes</option>
 </optgroup>
</select>
```

In this example, cars are grouped by country, making it easier for users to navigate the options.

Providing a Default Placeholder

Although the <select> tag does not natively support a placeholder option, you can simulate this by adding a non-selectable default option.

```
<select name="cars">
 <option value="" disabled selected>Select a car</option>
 <option value="volvo">Volvo</option>
 <option value="saab">Saab</option>
 <option value="fiat">Fiat</option>
 <option value="audi">Audi</option>
</select>
```

Warning

Dear student, you have to make your notes from e-books only.

Thank you

LIVE STREAM

SUPPORT

7268941266

piitroffice@gmail.com

www.piitr.org || www.piitr.com

Pragya Institute of IT & Research