###

HTML By Prasad

###

HTML Introduction:

HTML (Hypertext Markup Language):

HTML stands for "Hypertext Markup Language."

It is the standard markup language used to create and structure content on the World Wide Web.

Hypertext and Hyperlinks:

HTML is designed for creating hypertext documents, which contain text that can be linked to other documents or resources.

Hyperlinks are clickable elements that allow users to navigate between web pages.

Structure and Semantics:

HTML provides a way to structure content into headings, paragraphs, lists, and more.

It uses tags, enclosed in angle brackets, to define the structure and semantics of the content.

Web Browsers:

Web browsers (e.g., Chrome, Firefox, Safari) render HTML documents and display them as web pages.

HTML History:

HTML 1.0 (1993):

The first version of HTML was released by Tim Berners-Lee in 1993.

It included basic elements like headings, paragraphs, links, and lists.

HTML 2.0 (1995):

HTML 2.0 introduced tables, image support, and forms.

HTML 3.2 (1997):

HTML 3.2 added support for frames, which allowed multiple documents to be displayed in a single window.

HTML 4.0 (1997):

HTML 4.0 introduced CSS (Cascading Style Sheets) for better control over the presentation of web pages.

It also added support for scripting with JavaScript.

XHTML (2000):

XHTML (Extensible Hypertext Markup Language) was an XML-based version of HTML.

It aimed to make HTML more compatible with XML and introduced stricter syntax rules.

HTML5 (2014):

HTML5 is the latest version of HTML and introduced many new elements and features.

It includes multimedia support, semantic elements, canvas for graphics, and enhanced form controls.

HTML Layout:

Document Structure:

HTML documents typically have a standard structure, including <!DOCTYPE>, <html>, <head>, and <body> elements.

The <!DOCTYPE> declaration specifies the document type and version.

The <html> element is the root element of the document.

The <head> section contains metadata, while the <body> section contains visible content.

HTML Headings and Formatting Tags

HTML Headings (<h1> to <h6>)

HTML headings are used to define the structure and hierarchy of a document.

There are six levels of headings in HTML, from <h1> (highest) to <h6> (lowest).

Example:

<h1>This is a Heading 1</h1>

<h2>This is a Heading 2</h2>

<h3>This is a Heading 3</h3>

<h4>This is a Heading 4</h4>

<h5>This is a Heading 5</h5>

<h6>This is a Heading 6</h6>

Use headings to provide a clear and logical structure to your content.

Search engines use headings to understand the hierarchy of your content and improve SEO.

HTML Paragraphs (<p>)

The <p> tag is used to define paragraphs of text.

Paragraphs are block-level elements and are typically separated by vertical spacing.

Example:

<p>This is a paragraph of text. It can contain multiple sentences and line breaks.</p>

HTML Line Breaks (<br>)

The <br> tag is used to insert line breaks within text.

It is an inline element and does not require a closing tag.

Example:<p>This is a line of text.
Here is a new line of text.</p>

HTML Horizontal Rule (<hr>)

The <hr> tag is used to insert a horizontal line (horizontal rule) to separate content.

It is a self-closing tag and is typically used to divide sections of content.

Example:<p>This is some content.</p>

<hr>

<p>This is more content.</p>

HTML Emphasis () and Strong ()

The tag is used to emphasize text, typically displayed in italics.

The tag is used to highlight strong importance, typically displayed in bold.

Example:

<p>This text is emphasized.</p>

<p>This text is strongly emphasized.</p>

HTML Abbreviation (<abbr>)

The <abbr> tag is used to define an abbreviation or acronym.

It can include a "title" attribute that provides the full explanation of the abbreviation.

Example:

<p><abbr title="World Health Organization">WHO</abbr> is an important organization.</p>

HTML Subscript (<sub>) and Superscript (<sup>)

The <sub> tag is used to define subscript text (e.g., chemical formulas).

The <sup> tag is used to define superscript text (e.g., mathematical exponents).

Example:

<p>H₂O is the chemical formula for water.</p>

<p>2³ equals 8 in mathematics.</p>

HTML Citation (<cite>)

The <cite> tag is used to define the title of a creative work (e.g., book, movie, article).

It is often used within <blockquote> or <q> elements to cite the source.

Example:

<blockquote>

   <p>HTML is the foundation of modern web development.</p>

   <cite>W3C HTML Specification</cite>

</blockquote>

HTML Code (<code>) and Keyboard Input (<kbd>)

The <code> tag is used to define computer code or inline code snippets.

The <kbd> tag is used to define keyboard input.

Example:

<p>You can use the <code><code></code> tag for inline code.</p>

<p>Press <kbd>Ctrl</kbd> + <kbd>C</kbd> to copy text.</p>

Semantic HTML Tags:

Semantic HTML tags are used to give meaning to the structure of a web page. They convey the purpose or role of the elements to both browsers and developers. Using semantic tags improves accessibility, search engine optimization (SEO), and the overall structure of web content.

1. <header>:

The <header> element represents a container for introductory content or a set of navigational links.

It typically contains the site logo, site title, and main navigation menu.

2. <nav>:

The <nav> element is used to define a section of navigation links.

It often contains links to various parts of the website, such as the main menu or navigation bar.

3. <main>:

The <main> element is used to define the main content area of a document.

It should be unique within the document and exclude content that is repeated, such as site headers or footers.

4. <article>:

The <article> element represents a self-contained composition within a document.

It can include articles, blog posts, comments, or any content that can be independently distributed or syndicated.

5. <section>:

The <section> element is used to group related content within a document.

It helps in organizing content and makes it more meaningful.

6. <aside>:

The <aside> element represents content that is tangentially related to the content around it.

It is typically used for sidebars, pull quotes, or advertisements.

7. <footer>:

The <footer> element represents a container for the footer of a section or the entire page.

It usually contains copyright information, contact details, or links to related documents.

8. <time>:

The <time> element is used to represent dates and times.

It can include a "datetime" attribute to provide machine-readable date and time information.

<a> (Anchor) Tag:

The <a> tag is used for creating hyperlinks to other web pages, resources, or email addresses. It is one of the fundamental elements for navigation on the web.

Example:

Visit Example Website

Contact Us

The "href" attribute specifies the destination URL or email address.

Text or content placed between the opening and closing <a> tags becomes a clickable link.

<img> (Image) Tag:

The <img> tag is used to display images on a web page. Images can enhance content, provide visual context, and improve user engagement.

<img src="image.jpg" alt="A beautiful sunset">

The "src" attribute specifies the image file's source URL.

The "alt" attribute provides alternative text that is displayed if the image cannot be loaded. It is also essential for accessibility.

Lists in HTML:

HTML provides three types of lists: ordered lists, unordered lists, and description lists. Lists are used to organize and structure content.

1. Unordered Lists (<ul>):

Unordered lists are used to create lists of items that do not have a specific order or sequence.

Each item in an unordered list is represented by a bullet point.

Example:

```
<ul>
   <li>Item 1</li>
   <li>Item 2</li>
   <li>Item 3</li>
</ul>
```

2. Ordered Lists (<ol>):

Ordered lists are used to create lists of items that have a specific order or sequence.

Each item in an ordered list is represented by a number or another marker.

Example:

```
<ol>
   <li>First item</li>
   <li>Second item</li>
   <li>Third item</li>
</ol>
```

3. Description Lists (<dl>):

Description lists are used to define terms and their corresponding descriptions.

They consist of term (<dt>) and description (<dd>) pairs.

Example:

```
<dl>
   <dt>Term 1</dt>
```

<dd>Description of Term 1</dd>

    <dt>Term 2</dt>

    <dd>Description of Term 2</dd>

</dl>

Tables in HTML:

HTML tables are used to display data in rows and columns. They are useful for presenting structured information.

Basic Table Structure:

A table is created using the <table> element.

Rows within the table are defined using <tr> elements.

Cells within rows are defined using <td> elements for regular data cells and <th> elements for header cells.

Example:

```
<table>

  <tr>

    <th>Header 1</th>

    <th>Header 2</th>

  </tr>

  <tr>

    <td>Data 1</td>

    <td>Data 2</td>

  </tr>

  <tr>

    <td>Data 3</td>

    <td>Data 4</td>

  </tr>

</table>
```

Attributes:

The "colspan" attribute is used to specify how many columns a cell should span horizontally.

The "rowspan" attribute is used to specify how many rows a cell should span vertically.

Example:

```
<table>

  <tr>

    <th colspan="2">Header 1 and Header 2 Combined</th>

  </tr>
```

```
  <tr>
    <td rowspan="2">Data 1</td>
    <td>Data 2</td>
  </tr>
  <tr>
    <td>Data 3</td>
  </tr>
</table>
```

Block vs. Inline Elements:

Block Elements:

Block-level elements are used to create structural divisions or blocks in the content of a web page.

They typically start on a new line and extend the full width of their parent container.

Common block-level elements include <div>, <p>, <h1> to <h6>, <ul>, <ol>, <li>, <table>, <form>, and more.

Example:

<div>This is a block-level element.</div>

<p>This is another block-level element.</p>

<h1>This is a heading, which is also a block-level element.</h1>

Inline Elements:

Inline elements are used to style or wrap specific portions of text within a block-level element.

They do not start on a new line and only occupy the space needed for their content.

Common inline elements include , <a>, <strong>, , <abbr>, <img>, and more.

Example:

<p>This is a portion of text within a paragraph.</p>

This is an inline element.

Class and ID Attributes:

Class Attribute:

The class attribute is used to assign one or more class names to an HTML element.

Classes are typically used to apply CSS styles to one or more elements with the same class.

Multiple elements can share the same class.

Example:

<p class="highlight">This is a highlighted paragraph.</p>

<p class="highlight">Another highlighted paragraph.</p>

ID Attribute:

The id attribute is used to assign a unique identifier to an HTML element within a document.

IDs must be unique within the entire HTML document, meaning no two elements should share the same ID.

IDs are often used for specific JavaScript interactions or unique CSS styling.

Example:

<div id="header">This is the header section.</div>

<p id="intro">This is the introductory paragraph.</p>

Combining Classes and IDs:

It is common to combine classes and IDs to target specific elements with CSS or JavaScript.

An element can have both a class and an ID.

Example:

<div id="header" class="highlight">This is a highlighted header.</div>

HTML Forms:

HTML forms are used to collect user input and send it to a server for processing. They consist of various form elements for text input, selection, and interaction.

<form> Element:

The <form> element is used to create a form on a web page.

It acts as a container for form elements.

The "action" attribute specifies the URL to which the form data should be sent.

The "method" attribute defines the HTTP method (GET or POST) used to send the data.

Example:

<form action="/submit" method="POST">

   <!-- Form elements go here -->

</form>

Text Input Fields (<input type="text">):

Text input fields are used for single-line text input.

The "type" attribute is set to "text."

The "name" attribute specifies the name of the input field.

Example:

<label for="username">Username:</label>

<input type="text" id="username" name="username">

Password Input Fields (<input type="password">):

Password input fields are used for entering sensitive information, like passwords.

The "type" attribute is set to "password."

Example:

<label for="password">Password:</label>

<input type="password" id="password" name="password">

Radio Buttons (<input type="radio">):

Radio buttons allow users to select a single option from a list.

Each radio button has a unique "name" attribute, but they share the same "name" to form a group.

The "value" attribute specifies the value sent when the radio button is selected.

Example:

<label for="male">Male</label>

<input type="radio" id="male" name="gender" value="male">

<label for="female">Female</label>

<input type="radio" id="female" name="gender" value="female">

Checkboxes (<input type="checkbox">):

Checkboxes allow users to select one or more options from a list.

Each checkbox is independent and can have its own "name" and "value" attributes.

Example:

<label for="subscribe">Subscribe to Newsletter:</label>

<input type="checkbox" id="subscribe" name="newsletter" value="yes">

Textarea (<textarea>):

Textareas are used for multi-line text input.

They do not have a "type" attribute but can have "name" and "id" attributes.

The content between the opening and closing <textarea> tags defines the default text.

Example:

<label for="comments">Comments:</label>

<textarea id="comments" name="comments" rows="4" cols="50">Enter your comments here...</textarea>

Dropdown Lists (<select> and <option>):

Dropdown lists are used to select one option from a list of options.

The <select> element creates the dropdown list, and <option> elements define the options.

The "value" attribute specifies the value sent to the server when an option is selected.

Example:

<label for="country">Country:</label>

<select id="country" name="country">

   <option value="usa">United States</option>

   <option value="canada">Canada</option>

    <option value="uk">United Kingdom</option>

</select>

File Upload (<input type="file">):

File upload fields allow users to select and upload files.

The "type" attribute is set to "file."

The "name" attribute specifies the name for the uploaded file.

Example:

<label for="file">Upload a file:</label>

<input type="file" id="file" name="uploadfile">

Fieldset and Legend (<fieldset> and <legend>):

The <fieldset> element groups related form elements together and creates a visual border.

The <legend> element provides a caption or title for the <fieldset>.

They improve the organization and accessibility of the form.

Example:

<fieldset>

   <legend>Contact Information</legend>

   <!-- Form elements go here -->

</fieldset>

Miscellaneous HTML Tags:

1. <hr> (Horizontal Rule):

The <hr> tag is used to create a horizontal line or thematic break in the content.

It is a self-closing tag and does not have any content.

Example:

<p>This is some content.</p>

<hr>

<p>This is more content.</p>

2.
 (Line Break):

The <br> tag is used to insert a line break within text or content.

It is a self-closing tag and is typically used to force a new line.

Example:

<p>This is a line of text.
Here is a new line of text.</p>

3. <blockquote>:

The <blockquote> element is used to represent a block of quoted text from another source.

It is often used in conjunction with the <cite> element to cite the source.

Example:

<blockquote>

  <p>This is a quoted text from another source.</p>

</blockquote>

<cite>Source Name</cite>

4. <q> (Inline Quotation):

The <q> tag is used to define a short inline quotation.

Browsers often add quotation marks around the content enclosed by <q>.

Example:

<p>This is an example of an <q>inline quotation</q> within a paragraph.</p>

5. <abbr> (Abbreviation):

The <abbr> tag is used to define an abbreviation or acronym.

It can include a "title" attribute that provides the full explanation of the abbreviation.

Example:

<p><abbr title="World Health Organization">WHO</abbr> is an important organization.</p>

6. <address>:

The <address> element is used to define contact information for the author or owner of a document.

It can include postal addresses, phone numbers, email addresses, and more.

Example:

<address>

  John Doe

  123 Main Street

  City, State, ZIP

  Email: john@example.com

  Phone: (123) 456-7890

</address>

7. <mark> (Marked or Highlighted Text):

The <mark> tag is used to highlight or mark portions of text within content.

It indicates that the highlighted text has special relevance or importance.

Example:

<p>This is an example of <mark>highlighted text</mark> in a paragraph.</p>

8. <del> (Deleted Text):

The <del> tag is used to represent text that has been deleted or removed from a document.

It is often displayed with a strikethrough.

Example:

`<p>This is some <del>deleted</del> text in a paragraph.</p>`

9. <ins> (Inserted Text):

The <ins> tag is used to represent text that has been inserted or added to a document.

It is often displayed with an underline.

Example:

`<p>This is some <ins>inserted</ins> text in a paragraph.</p>`

10. <s> (Strikethrough):

The <s> tag is used to apply a strikethrough style to text, indicating that it is no longer accurate or relevant.

Example:

`<p>This is some <s>strikethrough</s> text in a paragraph.</p>`

1. Semantic Elements:

HTML5 introduced semantic elements that provide better structure and meaning to web documents. Here are a few examples:

```
<header>
    <h1>Welcome to My Website</h1>
    <nav>
        <ul>
            <li><a href="#">Home</a></li>
            <li><a href="#">About</a></li>
            <li><a href="#">Contact</a></li>
        </ul>
    </nav>
</header>
<main>
    <section>
        <h2>About Us</h2>
        <p>Learn more about our company.</p>
    </section>
    <article>
        <h2>Latest News</h2>
        <p>Read our latest articles and updates.</p>
    </article>
</main>
```

```html
<aside>

  <h3>Related Links</h3>

  <ul>

    <li><a href="#">FAQ</a></li>

    <li><a href="#">Terms of Service</a></li>

  </ul>

</aside>

<footer>

  <p>&copy; 2023 My Website</p>

</footer>
```

2. Audio and Video Support:

HTML5 introduced native support for audio and video. You can embed audio and video content like this:

```html
<audio controls>

  <source src="audio.mp3" type="audio/mpeg">

  Your browser does not support the audio element.

</audio>

<video controls width="400">

  <source src="video.mp4" type="video/mp4">

  Your browser does not support the video element.

</video>
```

3. Canvas:

The <canvas> element allows for dynamic rendering of graphics and animations using JavaScript. Here's a simple example:

```html
<canvas id="myCanvas" width="200" height="100"></canvas>

<script>

  var canvas = document.getElementById("myCanvas");

  var ctx = canvas.getContext("2d");

  ctx.fillStyle = "blue";

  ctx.fillRect(0, 0, 200, 100);

</script>
```

4. Geolocation:

You can access a user's geographical location using the Geolocation API. Here's an example:

```html
<button onclick="getLocation()">Get Location</button>

<p id="demo"></p>
```

```
<script>

   function getLocation() {

      if (navigator.geolocation) {

         navigator.geolocation.getCurrentPosition(showPosition);

      } else {

         document.getElementById("demo").innerHTML = "Geolocation is not supported by this browser.";

      }

   }

   function showPosition(position) {

      document.getElementById("demo").innerHTML = "Latitude: " + position.coords.latitude +
"<br>Longitude: " + position.coords.longitude;

   }

</script>
```

5. Local Storage (localStorage) and Session Storage:

localStorage and sessionStorage are used for client-side storage. Here's an example using localStorage:

```
<input type="text" id="data">

<button onclick="saveData()">Save Data</button>

<p id="result"></p>

<script>

   function saveData() {

      var data = document.getElementById("data").value;

      localStorage.setItem("user_data", data);

      document.getElementById("result").innerHTML = "Data saved: " + data;

   }

</script>
```

6. Web Workers:

Web Workers enable multi-threaded processing. Here's a simple example:

```
<script>

   var worker = new Worker("worker.js");

   worker.onmessage = function(event) {

      document.getElementById("result").innerHTML = "Result: " + event.data;

   };

   worker.postMessage("Hello from the main thread!");

</script>
```

7. WebSockets:

HTML5 WebSockets enable real-time communication. Here's a client-side example:

```
<script>
  var socket = new WebSocket("ws://example.com/socket");
  socket.onmessage = function(event) {
    document.getElementById("message").innerHTML = "Received: " + event.data;
  };
  function sendMessage() {
    var message = document.getElementById("messageInput").value;
    socket.send(message);
  }
</script>
```

8. Responsive Web Design:

HTML5 supports responsive web design with media queries. Here's a basic media query example:

```
<style>
  @media (max-width: 600px) {
    body {
      background-color: lightblue;
    }
  }
</style>
```