HTML

Hyper Text: HyperText simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. **With the help of HTML only, we can create static web pages**.

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

History of HTML:

In the late 1980's , a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet based hypertext system.

Tim Berners-Lee is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5, which we will learn later in this tutorial

HTML Versions:

HTML 1.0: The first version of HTML was 1.0, which was the barebones version of HTML language, and it was released in1991.

HTML

HTML 2.0: This was the next version which was released in 1995, and it was standard language version for website design. HTML 2.0 was able to support extra features such as form-based file upload, form elements such as text box, option button, etc.

HTML 3.2: HTML 3.2 version was published by W3C in early 1997. This version was capable of creating tables and providing support for extra options for form elements. It can also support a web page with complex mathematical equations. It became an official standard for any browser till January 1997. Today it is practically supported by most of the browsers.

HTML 4.01: HTML 4.01 version was released on December 1999, and it is a very stable version of HTML language. This version is the current official standard, and it provides added support for stylesheets (CSS) and scripting ability for various multimedia elements.

HTML5 : HTML5 is the newest version of HyperText Markup language. The first draft of this version was announced in January 2008. There are two major organizations one is W3C (World Wide Web Consortium), and another one is WHATWG(Web Hypertext Application Technology Working Group) which are involved in the development of HTML 5 version, and still, it is under development.

Features of HTML:

1) It is a very **easy and simple language**. It can be easily understood and modified.

2) It is very easy to make an **effective presentation** with HTML because it has a lot of formatting tags.

3) It is a **markup language**, so it provides a flexible way to design web pages along with the text.

4) It facilitates programmers to add a **link** on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.

5) It is **platform-independent** because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.

6) It facilitates the programmer to add **Graphics, Videos, and Sound** to the web pages which makes it more attractive and interactive.

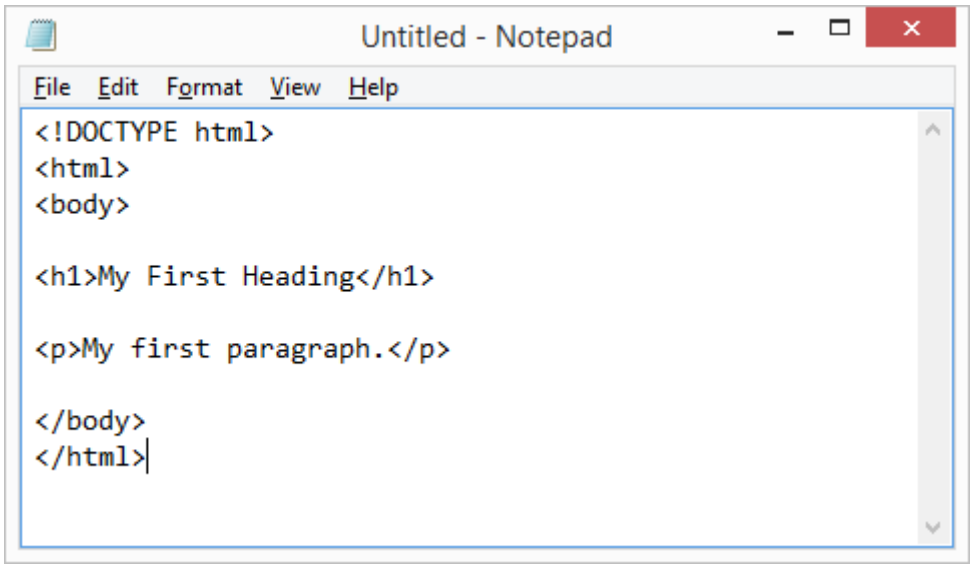
7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

HTML

Text Editors

- An HTML file is a text file, so to create an HTML file we can use any text editors.
- Text editors are the programs which allow editing in a written text, hence to create a web page we need to write our code in some text editor.
- For a beginner, the best text editor is Notepad (Windows) or TextEdit (Mac).
- Professional text editors which are, **Notepad++, Sublime Text, etc**.

Sample ex:



HTML Structure:

All HTML documents must start with a document type declaration:

`<!DOCTYPE html>`.

The HTML document itself begins with `<html>` and ends with `</html>`.

The visible part of the HTML document is between `<body>` and `</body>`.

HTML

HTML Basic Tags:

Heading Tag: HTML headings are defined with the `<h1>` to `<h6>` tags.

Paragraph: The HTML `<p>` element defines a paragraph.

Line Break: The HTML `<br>` element defines a line break.

HTML Styles:

- `style` attribute for styling HTML elements (`<tagname style="property:value;">`)
- `background-color` for background color
- `color` for text colors
- `font-family` for text fonts
- `font-size` for text sizes
- `text-align` for text alignment

HTML Attributes:

`src` attribute specifies the path to the image

`width` and `height` attributes

`alt` attribute

<img **src**="https://static.pexels.com/photos/36753/flower-purple-lical-blosso.jpg" **width**="100" **height**="100" **alt**="flower image">

HTML Formatting Elements:

- `<b>` - Bold text
- `<strong>` - Important text
- `<i>` - Italic text
- `<em>` - Emphasized text
- `<mark>` - Marked text
- `<small>` - Smaller text
- `<del>` - Deleted text
- `<ins>` - Inserted text
- `<sub>` - Subscript text
- `<sup>` - Superscript text

HTML

HTML Comments:

```
<!-- Write your comments here -->
```

HTML Quotation and Citation Elements:

`<blockquote>` - defines a section that is quoted from another source, and used for indent `<blockquote>` elements.

`<q>` - Insert quotation marks around the quotation.

`<abbr>` - tag defines an abbreviation or an acronym, like "HTML".

`<address>` - text in the `<address>` element usually renders in *italic.*

HTML Colors:

RGB: rgb(255, 0, 0)

HEX: #ff0000

HSL: `hsl(hue, saturation, lightness)` - hsl(0,100%,50%)

HTML Images:

`background-image` - To add a background image

HTML Favicon:

A favicon is a small image, so it should be a simple image with high contrast.

It is used inside the <head> tag for giving icon to the webpage.

<link **rel**="icon" **type**="image/x-icon" **href**="https://thumbs.dreamstime.com/z/daisy-flower-isolated-hand-made-clipping-path-14184671.jpg">

HTML

HTML Tables:

`<table>` - Table tag

`<th>` - Table header tag

`<tr>` - Table row tag

`<td>` - Table data tag

1. Table Border

```
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
```

2. Style Border (border-style)

- dotted
- dashed
- solid
- double
- groove
- ridge
- inset
- outset
- none
- hidden

3. Color Border (border-color)
4. Rowspan & Colspan
    i. Colspan: When we want to merge multiple column for a cell
    ii. Rowspan: When we want to merge multiple row for a cell

HTML Lists:

1. Unordered List:

- : unordered list tag
- : list item tag

`list-style-type` property is used to define the style of the list item.

HTML

Value	Description
disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked

2.Ordered List

- : ordered list tag
- : list item tag

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters

type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

3. Description List:

- <dl>: description list
- <dt>: description term
- <dd>: description data

HTML Block and Inline Elements:

A block-level element always starts on a new line, and the browsers automatically add some space before and after the element. A block-level element always takes up the full width available.

Two commonly used block elements are: `<p>` and `<div>`.

<address>
<article>
<aside>
<blockquote>
<canvas>
<dd>
 	
<dl>
<dt>
 	
	<fieldset>
	<figcaption>
	<figure>
	<footer>
	<form>
	<h1>-<h6>
	<header>
	<hr>
	
	<video>
		<main>
		<nav>
		<noscript>
		
		<p>
		<pre>
		<section>
		<table>
		<tfoot>

An inline element does not start on a new line. An inline element only takes up as much width as necessary.

This is a element inside a paragraph.

<a> <abbr> <acronym>	<dfn> <i>	<select> <small>

HTML

<b> <bdo> <big> <button> <cite> <code> <samp> <script>	<img> <input> <kbd> <label> <map> <object> <output> <q> <var>	 <sub> <sup> <textarea> <time> <tt>

HTML Classes:

HTML `class` attribute is used to specify a class for an HTML element.
`class` attribute is often used to point to a class name in a style sheet. It can also be used by a JavaScript to access and manipulate elements with the specific class name.

Syntax: write . followed by the classname to give style to the class.

HTML Id:

The `id` attribute specifies a unique id for an HTML element. The value of the `id` attribute must be unique within the HTML document.

The `id` attribute is used to point to a specific style declaration in a style sheet. It is also used by JavaScript to access and manipulate the element with the specific id.

Syntax for id is: write a hash character (#), followed by an id name. Then, define the CSS properties within curly braces {}

HTML Frame:

Inline frame is used to embed another document within the current HTML document.

<iframe **src**="https://www.facebook.com/" **height**="300" **width**="400">

</iframe>

HTML Responsive:

To create a responsive website, add the following `<meta>` tag to all your web pages. It is used inside the `<head>` tag.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

HTML Forms:

HTML `<form>` element is used to create an HTML form for user input.

Inside the form there is most commonly used elements in the form which is `<input>`. `<input>` element can be displayed in many ways, depending on the `type` attribute.

There are different types of input fields used in the form which are as given below.

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

HTML

Text Field:

`<input type="text">` defines a single-line input field for text input. This tag will be used for the input text field.

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```

Input Type Password:

`<input type="password">` defines a **password field**. The characters in a password field are masked (shown as asterisks or circles).

Input Type Reset:

`<input type="reset">` defines a **reset button** that will reset all form values to their default values.

Input Type Color:

The `<input type="color">` is used for input fields that should contain a color.

Depending on browser support, a color picker can show up in the input field.

Input Type Date:

The `<input type="date">` is used for input fields that should contain a date.

Depending on browser support, a date picker can show up in the input field. We can also use the `min` and `max` attributes to add restrictions to dates

Input Type Datetime-local:

The `<input type="datetime-local">` specifies a date and time input field, with no time zone.

Input Type Email:

The `<input type="email">` is used for input fields that should contain an e-mail address.

HTML

Depending on browser support, the e-mail address can be automatically validated when submitted.

Some smartphones recognize the email type, and add ".com" to the keyboard to match email input.

Input Type File:

The `<input type="file">` defines a file-select field and a "Browse" button for file uploads.

Input Type Month:

The `<input type="month">` allows the user to select a month and year.

Depending on browser support, a date picker can show up in the input field.

Input Type Number:

The `<input type="number">` defines a **numeric** input field.

You can also set restrictions on what numbers are accepted.

The following example displays a numeric input field, where you can enter a value from 1 to 5 with min & max.

Input Type Range:

The `<input type="range">` defines a control for entering a number whose exact value is not important (like a slider control). Default range is 0 to 100. However, you can set restrictions on what numbers are accepted with the `min`, `max` attributes

Radio Buttons:

`<input type="radio">` defines a radio button. Radio buttons let a user select ONE of a limited number of choices.

```
<form>
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language" value="JavaSc
ript">
  <label for="javascript">JavaScript</label>
</form>
```

HTML

Checkboxes:

`<input type="checkbox">` defines a **checkbox**. Checkboxes let a user select ZERO or MORE options of a limited number of choices.

```
<form>
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
  <label for="vehicle1"> I have a bike</label><br>
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
  <label for="vehicle2"> I have a car</label><br>
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
  <label for="vehicle3"> I have a boat</label>
</form>
```

Submit Form:

`<input type="submit">` defines a button for submitting the form data.

Name Attribute:

Notice that each input field must have a `name` attribute to be submitted.

If the `name` attribute is omitted, the value of the input field will not be sent at all.

HTML Form Elements

The HTML `<form>` element can contain one or more of the following form elements:

- `<input>`
- `<label>`
- `<select>`
- `<textarea>`
- `<button>`
- `<fieldset>`
- `<legend>`
- `<datalist>`
- `<option>`
- `<optgroup>`

HTML

<label> Element:

The `<label>` element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The `<label>` element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the `<label>` element, it toggles the radio button/checkbox.

The `for` attribute of the `<label>` tag should be equal to the `id` attribute of the `<input>` element to bind them together.

The <select> Element

The `<select>` element defines a drop-down list

```
<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>
```

The `<option>` elements defines an option that can be selected.

By default, the first item in the drop-down list is selected.

Visible Values:

Use the `size` attribute to specify the number of visible values

The <textarea> Element:

The `<textarea>` element defines a multi-line input field (a text area)

The `rows` attribute specifies the visible number of lines in a text area.

The `cols` attribute specifies the visible width of a text area.

You can also define the size of the text area by using CSS

The <button> Element

The `<button>` element defines a clickable button

HTML

```
<button type="button" onclick="alert('Hello World!')">Click
Me!</button>
```

The <fieldset> and <legend> Elements

The `<fieldset>` element is used to group related data in a form.

The `<legend>` element defines a caption for the `<fieldset>` element.

```
<form>
  <fieldset>
    <legend>Personal Data:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

HTML Input Attributes

The value Attribute

The input `value` attribute specifies an initial value for an input field

The readonly Attribute

The input `readonly` attribute specifies that an input field is read-only.

A read-only input field cannot be modified (however, a user can tab to it, highlight it, and copy the text from it).

The value of a read-only input field will be sent when submitting the form!

The disabled Attribute

The input `disabled` attribute specifies that an input field should be disabled.

A disabled input field is unusable and un-clickable.

The value of a disabled input field will not be sent when submitting the form!

The size Attribute

The input `size` attribute specifies the visible width, in characters, of an input field.

The default value for `size` is 20.

Note: The `size` attribute works with the following input types: text, search, tel, url, email, and password.

The maxlength Attribute

The input `maxlength` attribute specifies the maximum number of characters allowed in an input field.

The min and max Attributes

The input `min` and `max` attributes specify the minimum and maximum values for an input field.

The `min` and `max` attributes work with the following input types: number, range, date, datetime-local, month, time and week.

The pattern Attribute

The input `pattern` attribute specifies a regular expression that the input field's value is checked against, when the form is submitted.

The `pattern` attribute works with the following input types: text, date, search, url, tel, email, and password.

The placeholder Attribute

The input `placeholder` attribute specifies a short hint that describes the expected value of an input field (a sample value or a short description of the expected format).

The short hint is displayed in the input field before the user enters a value.

HTML

The `placeholder` attribute works with the following input types: text, search, url, tel, email, and password.

The required Attribute

The input `required` attribute specifies that an input field must be filled out before submitting the form.

The `required` attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

The autofocus Attribute

The input `autofocus` attribute specifies that an input field should automatically get focus when the page loads.

Title Attribute

Use for pattern matching output to the user for showing proper format data to the user to input valid data.

HTML Form Attributes

The Action Attribute

The action attribute defines the action to be performed when the form is submitted. **The** action **is an attribute of** <form> **element that specifies the** url **of the second web page.**

Syntax

```
<form action="URL of page"> ..... </form>
```

Note

If the `action` attribute is omitted, the action is set to the current page.

The Target Attribute

The `target` attribute specifies where to display the response that is received after submitting the form. The following are the keywords used with the target attribute.

- **_self:** If we use _self as an attribute value, then the response will display in current page only.
- **_blank:** If we use _blank as an attribute it will load the response in a new page.

HTML method attribute

The method attribute defines the HTTP method which browser used to submit the form.

- **post:** We can use the post value of method attribute when we want to process the sensitive data as it does not display the submitted data in URL. It is secured.
- **get:** The get value of method attribute is default value while submitting the form. But this is not secure as it displays data in URL after submitting the form. It is not secure.

The Novalidate Attribute

- The `novalidate` attribute is a boolean attribute.
- When present, it specifies that the form-data (input) should not be validated when submitted.

HTML Canvas Graphics

What is HTML Canvas?

The HTML `<canvas>` element is used to draw graphics on a web page.

The `<canvas>` element is only a container for graphics. You must use JavaScript to actually draw the graphics. Canvas has several methods for drawing paths, boxes, circles, text, and adding images.

A canvas is a rectangular area on an HTML page. By default, a canvas has no border and no content.

Note: Always specify an `id` attribute (to be referred to in a script), and a `width` and `height` attribute to define the size of the canvas. To add a border, use the `style` attribute.

```
<canvas id="myCanvas" width="200" height="100" style="border:1px solid
#000000;">
</canvas>
```

HTML SVG Graphics

- SVG stands for Scalable Vector Graphics
- SVG is used to define graphics for the Web
- SVG is a W3C recommendation

SVG is mostly used for vector type diagrams like pie charts, 2-Dimensional graphs in an X,Y coordinate system etc. The <svg> element specifies the root of a SVG fragment.

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="green" stroke-
width="4" fill="yellow" />
</svg>
```

HTML MEDIA

HTML Video:

The HTML `<video>` element is used to show a video on a web page.

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
</video>
```

The `controls` attribute adds video controls, like play, pause, and volume.

To start a video automatically, use the `autoplay` attribute.

Add `muted` after `autoplay` to let your video start playing automatically.

HTML Image:

`<img>` tag is used to add the image in the webpage.

Attribute like width, height, alt are used in the image tag to specify the image properly.

HTML Audio:

The HTML `<audio>` element is used to play an audio file on a web page.

```
<audio controls>
  <source src="horse.ogg" type="audio/mpeg">
</audio>
```

HTML Youtube:

- Define an `<iframe>` element in your web page
- Let the `src` attribute point to the video URL
- Use the `width` and `height` attributes to specify the dimension of the player

```
<iframe width="420" height="315"
src="https://www.youtube.com/embed/tgbNymZ7vqY">
</iframe>
```