



**BRANCH - IT
SEMESTER 2
SECTION 2**

2022 WOST PRACTICAL

SUBMITTED BY

*Pranav Gupta
(UE218074)*

SUBMITTED TO

*Prof. Kritika Mehta
Panghal*

1

Basic HTML

HTML (HyperText Markup Language) is the code that is used to structure a web page and its content. For example, content could be structured within a set of paragraphs, a list of bulleted points, or using images and data tables.

So, what is HTML?

HTML is a markup language that defines the structure of your content. HTML consists of a series of elements, which you use to enclose, or wrap, different parts of the content to make it appear a certain way, or act a certain way. The enclosing tags can make a word or image hyperlink to somewhere else, can italicize words, can make the font bigger or smaller, and so on.

The main parts of our element are as follows:

1. The opening tag: This consists of the name of the element (in this case, p), wrapped in opening and closing angle brackets. This states where the element begins or starts to take effect — in this case where the paragraph begins.
2. The closing tag: This is the same as the opening tag, except that it includes a forward slash before the element name. This states where the element ends — in this case where the paragraph ends. Failing to add a closing tag is one of the standard beginner errors and can lead to strange results.
3. The content: This is the content of the element, which in this case, is just text.

Anatomy of an HTML document

- `<!DOCTYPE html>` — doctype. It is a required preamble. In the mists of time, when HTML was young (around 1991/92), doctypes were meant to act as links to a set of rules that the HTML page had to follow to be considered good HTML, which could mean automatic error checking and other useful things. However, these days they don't do much and are basically just needed to make sure your document behaves correctly. That's all you need to know for now.
- `<html></html>` — the `<html>` element. This element wraps all the content on the entire page and is sometimes known as the root element.
- `<head></head>` — the `<head>` element. This element acts as a container for all the stuff you want to include on the HTML page that *isn't* the content you are showing to your page's viewers. This includes things like keywords and a page description that you want to appear in search results, CSS to style our content, character set declarations, and more.
- `<meta charset="utf-8">` — This element sets the character set your document should use to UTF-8 which includes most characters from the vast majority of written languages. Essentially, it can now handle any textual content you might put on it. There is no reason not to set this and it can help avoid some problems later on.
- `<title></title>` — the `<title>` element. This sets the title of your page, which is the title that appears in the browser tab the page is loaded in. It is also used to describe the page when you bookmark/favourite it.
- `<body></body>` — the `<body>` element. This contains *all* the content that you want to show to web users when they visit your page, whether that's text, images, videos, games, playable audio tracks, or whatever else.

SYNTAX:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Basic HTML</title>

  </head>

  <body>

    <h1>Basic HTML</h1>

    <h3>What is HTML?</h3>
```

<p>

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.

</p>

<h3>Applications of HTML</h3>

<p>

Web document creation

Web pages development

Internet Navigation

Cutting edge feature

Client-side storage

Game Development usage

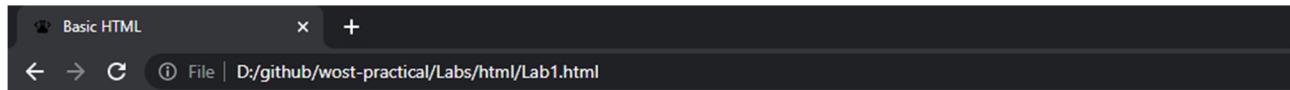
</p>

<h3>Structure of HTML</h3>

<p>


```
<strong>  
    <li>&lt;!DOCTYPE> - </li>  
    <li>&lt;html> - </li>  
    <li>&lt;head> - </li>  
    <li>&lt;title> - </li>  
    <li>&lt;body> - </li>  
</strong>  
</ol>  
</p>  
</body>  
</html>
```

OUTPUT:



Basic HTML

What is HTML?

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

Applications of HTML

1. Web document creation
2. Web pages development
3. Internet Navigation
4. Cutting edge feature
5. Client-side storage
6. Game Development usage

Structure of HTML

1. <!DOCTYPE> -
2. <html> -
3. <head> -
4. <title> -
5. <body> -

2

HTML Attributes

In the previous experiment we studied about the parts of an element as *opening tag*, *closing tag*, *content*. Apart from this, elements can also have attributes that look like the following:

The diagram shows a dark grey rectangular box containing an HTML snippet. A vertical line labeled "Attribute" points to the word "class" in the code. The code is as follows:

```
<p class="editor-note">My cat is very grumpy</p>
```

Attributes contain extra information about the element that you don't want to appear in the actual content. Here, class is the attribute name and editor-note is the attribute value. The class attribute allows you to give the element a non-unique identifier that can be used to target it (and any other elements with the same class value) with style information and other things.

An attribute should always have the following:

- A space between it and the element name (or the previous attribute, if the element already has one or more attributes).
- The attribute name followed by an equal sign.
- The attribute value wrapped by opening and closing quotation marks.

Let's look at some of the most common attributes in HTML elements.

<u>Attribute Name</u>	<u>Elements</u>	<u>Description</u>
bgcolor	<body><col><colgroup><marquee><table><tbody><tfoot><td><th><tr>	Background color of the element.
Class	Global attribute	Often used with CSS to style elements with common properties.
name	<button><form><fieldset><iframe><input><keygen><object><output><select><textarea><map><meta><param>	Name of the element. For example, used by the server to identify the fields in form submits.
title	Global attribute	Text to be displayed in a tooltip when hovering over the element.
value	<button><data><input><meter><option><progress><param>	Defines a default value which will be displayed in the element on page load.
type	<button><input><command><embed><object><script><source><style><menu><link>	Defines the type of the element.

SYNTAX:

```

<!DOCTYPE html>

<html>

  <head>
    <title>HTML Attributes</title>
  </head>

  <body>
    <p style = "color: red;">
      This is the color attribute.
    </p>

    <h3 style = "font-family: 'Times New Roman';">
      This is for using font family.
    </h3>
  
```

```
<h3 style = "font-size: 200%;">
```

This is for changing font size.

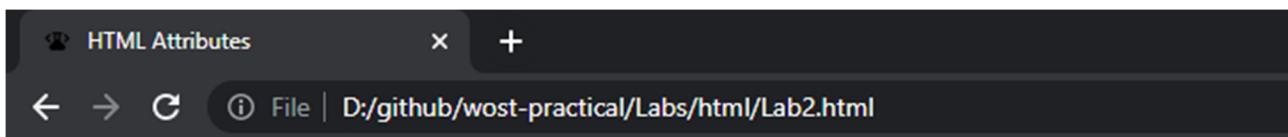
```
</h3>
```

```
<img src =
"https://www.google.com/images/branding/googlelogo/2x/googlelogo_light_color_92x30dp.png"
class="demo">
```

```
</body>
```

```
</html>
```

OUTPUT:



This is the color attribute.

This is for using font family.

This is for changing font size.

Google

3

Image Mapping

: The Image Embed element

The `` HTML element embeds an image into the document.

- The `src` attribute is required, and contains the path to the image you want to embed.
- The `alt` attribute holds a text description of the image, which isn't mandatory but is incredibly useful for accessibility — screen readers read this description out to their users so they know what the image means. Alt text is also displayed on the page if the image can't be loaded for some reason: for example, network errors, content blocking, or linkrot.

Now, we will explore `usemap` attribute of `` element. It is used to define an image map (a clickable link area). It uses two elements along with it to configure the mapping:

1. <map>: The Image Map element
2. <area>: The Image Map Area element

SYNTAX:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Image Mapping</title>

  </head>

  <body>

    <map name="mapping">

      <area shape="rect" coords="184,6,253,27"
            href="https://mozilla.org"
            target="_blank" alt="Mozilla" />

      <area shape="circle" coords="130,136,60"
            href="https://developer.mozilla.org/"
            target="_blank" alt="MDN" />

      <area shape="poly" coords="130,6,253,96,223,106,130,39"
            href="https://developer.mozilla.org/docs/Web/Guide/Graphics"
            target="_blank" alt="Graphics" />

      <area shape="poly" coords="253,96,207,241,189,217,223,103"
            href="https://developer.mozilla.org/docs/Web/HTML"
            target="_blank" alt="HTML" />

      <area shape="poly" coords="207,241,54,241,72,217,189,217"
            href="https://developer.mozilla.org/docs/Web/JavaScript"
            target="_blank" alt="JavaScript" />

      <area shape="poly" coords="54,241,6,97,36,107,72,217"
```

```
        href="https://developer.mozilla.org/docs/Web/API"
        target="_blank" alt="Web APIs" />

<area shape="poly" coords="6,97,130,6,130,39,36,107"
      href="https://developer.mozilla.org/docs/Web/CSS"
      target="_blank" alt="CSS" />

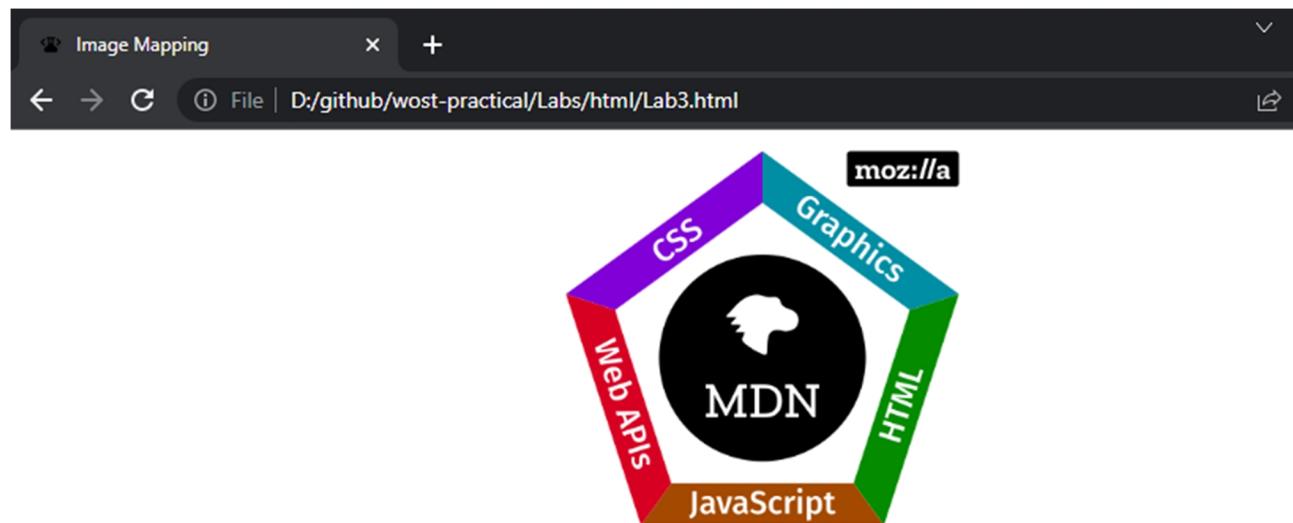
</map>



</body>

</html>
```

OUTPUT:



4

Text Formatting

HTML contains several elements for defining text with a special meaning. Formatting elements were designed to display special types of text:

1. **** - Bold text

The HTML **** element defines bold text, without any extra importance

2. **** - Important text

The HTML **** element defines text with strong importance. The content inside is typically displayed in bold

3. **<i>** - Italic text

The HTML **<i>** element defines a part of text in an alternate voice or mood. The content inside is typically displayed in italic

4. **** - Emphasized text

The HTML **** element defines emphasized text. The content inside is typically displayed in italic

5. **<mark>** - Marked text

The HTML **<mark>** element defines text that should be marked or highlighted

6. **<small>** - Smaller text

The HTML **<small>** element defines smaller text

7. ~~~~ - Deleted text

The HTML ~~~~ element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text

8. <ins> - Inserted text

The HTML <ins> element defines a text that has been inserted into a document. Browsers will usually underline inserted text

9. <sub> - Subscript text

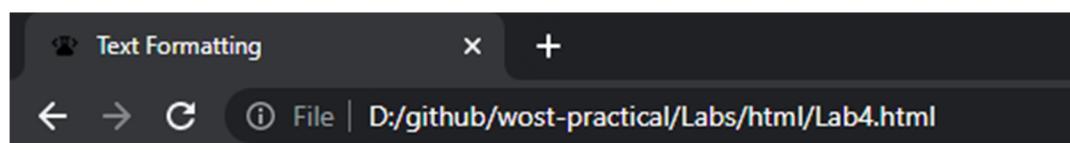
The HTML <sub> element defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas

10. <sup> - Superscript text

The HTML <sup> element defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font. Superscript text can be used for footnotes

SYNTAX:

```
<body>  
  
<p><b>This text is bold</b></p>  
  
<p><i>This text is italic</i></p>  
  
<p>This is<sub> subscript</sub> and <sup>superscript</sup></p>  
  
</body>
```

OUTPUT:

This text is bold

This text is italic

This is subscript and superscript

5

Hyperlinks

Hyperlinks are really important — they are what makes the Web a web.

What is a hyperlink?

Hyperlinks are one of the most exciting innovations the Web has to offer. They've been a feature of the Web since the beginning, and are what makes the Web a web. Hyperlinks allow us to link documents to other documents or resources, link to specific parts of documents, or make apps available at a web address. Almost any web content can be converted to a link so that when clicked or otherwise activated the web browser goes to another web address (URL).

Note: A URL can point to HTML files, text files, images, text documents, video and audio files, or anything else that lives on the Web. If the web browser doesn't know how to display or handle the file, it will ask you if you want to open the file (in which case the duty of opening or handling the file is passed to a suitable native app on the device) or download the file (in which case you can try to deal with it later on).

A basic link is created by wrapping the text or other content, see Block level links, inside an `<a>` element and using the `href` attribute, also known as a Hypertext Reference, or target, that contains the web address.

SYNTAX:

```
<body>
```

Click [here](https://google.com/) to go to google homepage.

Click [here](Lab1.html) to go to Assignment 1.

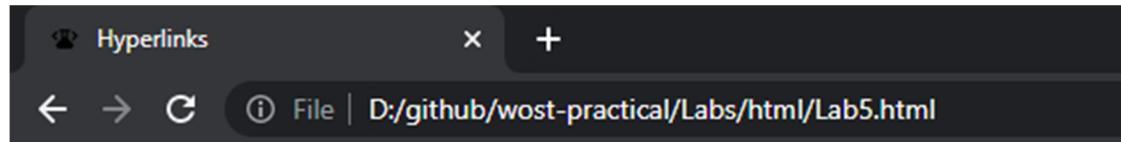
Click [here](Lab2.html) to go to Assignment 2.

Click [here](Lab3.html) to go to Assignment 3.

Click [here](Lab4.html) to go to Assignment 4.


```
</body>
```

OUTPUT:



Click [here](#) to go to google homepage.

Click [here](#) to go to Assignment 1.

Click [here](#) to go to Assignment 2.

Click [here](#) to go to Assignment 3.

Click [here](#) to go to Assignment 4.

6

Lists in HTML

HTML lists allow web developers to group a set of related items in lists. Namely there are 3 types of lists in HTML syntax:

1. Unordered HTML List

An unordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with bullets (small black circles) by default

2. Ordered HTML List

An ordered list starts with the `` tag. Each list item starts with the `` tag. The list items will be marked with numbers by default

3. HTML Description Lists

HTML also supports description lists. A description list is a list of terms, with a description of each term. The `<dl>` tag defines the description list, the `<dt>` tag defines the term (name), and the `<dd>` tag describes each term

SYNTAX:

```
<body>

<h3>Unordered List</h3>

<ul>

<li>Coffee</li>

<li>Tea</li>

<li>Milk</li>

</ul>

<h3>Ordered List</h3>

<ol>

<li>Coffee</li>

<li>Tea</li>

<li>Milk</li>

</ol>

<H3>Description List</H3>

<dl>

<dt>Coffee</dt>

<dd>- black hot drink</dd>

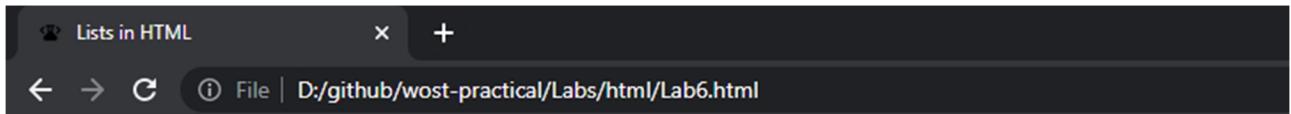
<dt>Milk</dt>

<dd>- white cold drink</dd>

</dl>

</body>
```

OUTPUT:



Unordered List

- Coffee
- Tea
- Milk

Ordered List

1. Coffee
2. Tea
3. Milk

Description List

Coffee
- black hot drink

Milk
- white cold drink

7

Frames in HTML

HTML allows programmers to divide a single browser display into multiple window sections, where each section can load individual URLs. This concept of HTML providing multiple frames at one browser display is called frameset, and all the frame tags are used within the container tag <frameset>. So, the entire separation of HTML pages is possible using the concept of frames. In this chapter, you will learn about the frames and how they are used to create multiple sections in a single browser display.

HTML <frameset> tag

This tag defines a specific window or frame inside the <frameset> tag. Every <frame> within the <frameset> tag may use attributes for different purposes like border, resizing capability, include scrolling, etc. The primary use of these frames was to display a menu in parts of the page with content in one part of the page. Multiple HTML pages can be viewed in a browser window using this tag.

<iframe>: The Inline Frame element

The <iframe> HTML element represents a nested browsing context, embedding another HTML page into the current one. Each embedded browsing context has its own session history and document. The browsing context that embeds the others is called the *parent browsing context*. The *topmost* browsing context — the one with no parent — is usually the browser window, represented by the Window object.

SYNTAX:

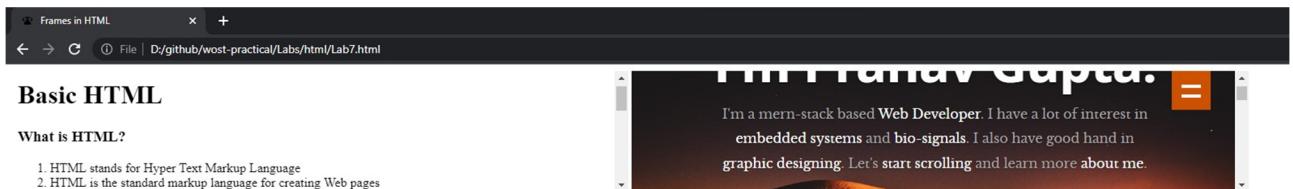
```
<body>

<iframe src="Lab1.html" frameborder="0"></iframe>

<iframe src="https://yolo-pranav.github.io/resume" frameborder="0"></iframe>

</body>
```

OUTPUT:



8

Tables in HTML

A very common task in HTML is structuring tabular data, and it has a number of elements and attributes for just this purpose. Coupled with a little CSS for styling, HTML makes it easy to display tables of information on the web such as your school lesson plan, the timetable at your local swimming pool, or statistics about your favorite dinosaurs or football team. This module takes you through all you need to know about structuring tabular data using HTML.

The `<table>` HTML element represents tabular data — that is, information presented in a two-dimensional table comprised of rows and columns of cells containing data. It has 3 sub elements for its configuration:

1. **<tr>**
This is used to define a new row in the table.
2. **<td>**
This is used to sort data in a particular into multiple cells.
3. **<th>**
It is used to assign heading cells in a row or column.

SYNTAX:

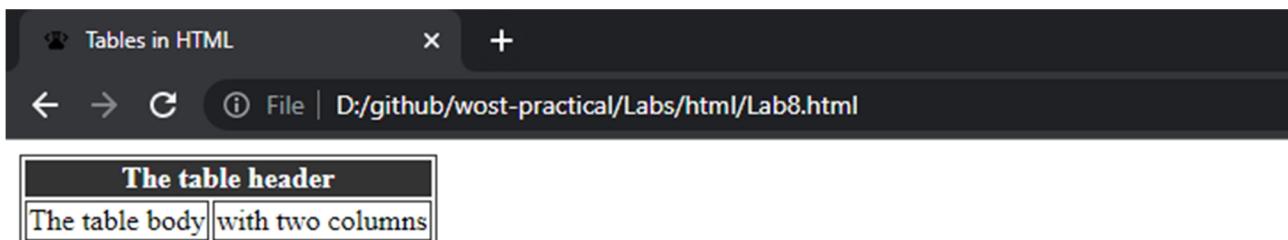
```
<table>

  <thead>
    <tr>
      <th colspan="2">The table header</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td>The table body</td>
      <td>with two columns</td>
    </tr>
  </tbody>

</table>
```

OUTPUT:



9

Basic CSS

Cascading Stylesheets — or CSS — is the first technology you should start learning after HTML. While HTML is used to define the structure and semantics of your content, CSS is used to style it and lay it out. For example, you can use CSS to alter the font, color, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features.

CSS syntax

CSS is a rule-based language — you define the rules by specifying groups of styles that should be applied to particular elements or groups of elements on your web page. For example, you can decide to have the main heading on your page to be shown as large red text.

- the CSS rule opens with a *selector*. This selects the HTML element that we are going to style.
- We then have a set of *curly braces* { }.
- Inside the braces will be one or more *declarations*, which take the form of property and value pairs.

Ways to import CSS

There are 3 different ways to import CSS in our webpage:

1. Inline CSS

In this method we style the elements by using “*style*” attribute within the element itself.

2. Internal CSS

In this method we use `<style>` element inside the `<head>` to define a custom stylesheet for the HTML document.

3. External CSS

In this method we use a separate file in '.css' extension with all css code to style our document. This sheet is linked with HTML document using `<link>` tag.

SYNTAX:

HTML FILE

```
<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" href="../css/Lab9.css">

<title>Basic CSS</title>

<style>

.internal-css {

    color: blue;

}

</style>

</head>

<body>

<p>Inline CSS</p>

<p style="color: red;">This paragraph is used to display inline css.</p>

<p>Internal CSS</p>

<p class="internal-css">This paragraph shows internal css.</p>

<p>External CSS</p>

<p class="external-css">This paragraph will get styling from external css.</p>
```

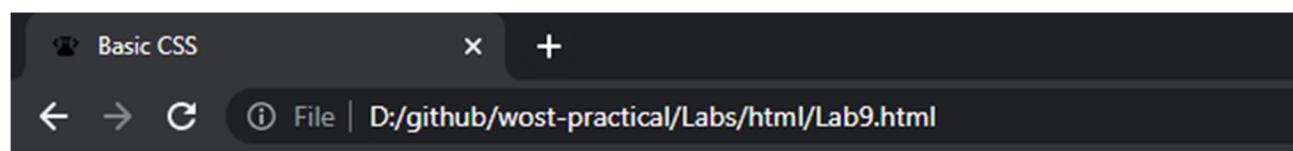
```
</body>
```

```
</html>
```

LAB9.css

```
.external-css {  
    color: aqua;  
}
```

OUTPUT:



Inline CSS

This paragraph is used to display inline css.

Internal CSS

This paragrapgh shows internal css.

External CSS

This paragraph will get styling form external css.