Introduction to HTML

- HTML is used to display the content in the browser.
- HTML is an interpreted language where each and every statement will be executed line by line.
- HTML stands for **Hypertext Markup Language.**

Any clickable things in the browser we call it as hypertext (Like images, links, video, audio, text etc)

OR

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites.

No computation or logical operations will be performed

Example: 3+3

Result: 3+3

OR

A computer language that consists of easily understood keywords, names, or tags that help format the overall view of a page and the data it contains.

History of HTML

- Tim Berners-Lee, a physicist at the CERN research institute in Switzerland invented HTML in 1991.
- 1991- Tim Berners-Lee invents HTML 1.0
- 1993- HTML 1.0 is released.
- 1995- HTML 2.0 is published.
- 1997- HTML 3.0 was invented.
- 1999- The widely-used HTML 4.0
- 2014- HTML 5.0 is released and used worldwide.

Types of Applications

- Mobile Application (Calculator)
- Desktop Application (Eclipse, MySQL Workbench)
- Web Application (Gmail, Facebook etc)

Note: Anything that is display in the browser by using an internet we call as website or web application.

What is Web?

A WWW or W3 is also known as a web, it is a system of interconnected public webpages accessible through the internet.

What is Web Page?

A web page is a simple document displayable by a browser.

What is Website?

A website is a collection of many web pages, and web pages are digital files that are written using HTML.

What is Web Application?

A web application (or web app) is application software that is accessed using a web browser.

What is the difference between website and web application?

Web Application	Website
In a web application we can interact with the users.	In a website we cannot interact with the users because we will have a static content.
In web application we can read and manipulate the contents.	In website we can only read the contents but we cannot manipulate the contents.
Example: Amazon etc	Example: Tap Academy website

Note: If we want to communicate with the browser by using the HTML, we need to use the TAG.

What is tag?

A tag is a keyword that holds the contents that should be displayed in the browser in webpage in the proper format. It will be enclosed within the angle brackets (<>).

Example: <html>, <head>, <body> etc.

Types of tags

1. **Paired tag:** Paired tags are such tags where you need to start it and you need to end it as well by using the forward slash as: **<html>--------</html> etc.**
2. **Unpaired tag:** Unpaired tags are such tags where you need to start it but you no need to end as: **
 etc.**

STRUCTURE OF HTML

First_page.html

```
<!DOCTYPE html>----------------------------------------1
<html>-------------------------------------------------2
  <head>-----------------------------------------------4
    <title>First_page</title>----------------------6
  </head>----------------------------------------------5
  <body>-----------------------------------------------7
    <p>Hi, welcome to HTML </p>--------------------9
  </body>----------------------------------------------8
</html>------------------------------------------------3
```

Output:

Hi, welcome to HTML

Note: In the about statements the numbering for each statement according to the execution flow.

<!DOCTYPE html>: Doctype HTML is a declaration that tells the browser what version of HTML the document is written in.

<html>: The <html> tag is a paired so we need to start it and we need to end it as well and before we start writing statements in html, we need to tell the browser that we are starting with html by using this <html> tag.

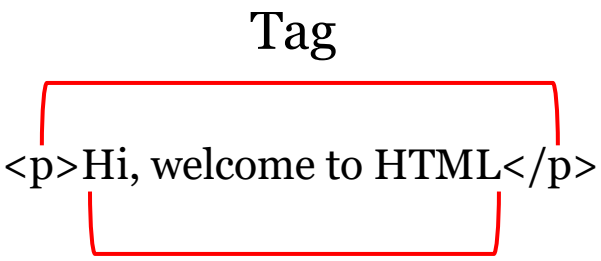
<head>: The <head> tag is used to define the head section where all the titles of the web page will be display and it is also paired tag once we start it, we need to end it as well.

<title>: The <title> tag is used to give the title for the web page and it is also a paired tag once you start it you need to end it as well.

<body>: The <body> tag is used to give and display the contents inside the body of the webpage and it is also a paired tag once you start it you need to end it as well.

<p>: The <p> tag is a paragraph tag which is used to display the character, word, text, sentence or paragraph in the web page, its stand for paragraph and it is also a paired tag where once you start it you need to end it as well.

What is an Element?



The tag+content we call it as element.

BASIC HTML TAGS

- **<p>**
- **<pre>**
- **
**
- **<hr>**
- **Heading tags**
  - **<h1>**
  - **<h2>**
  - **<h3>**
  - **<h4>**
  - **<h5>**
  - **<h6>**
- **<img>**
- **<video>**

- **<audio>**
- **<a>**

➢ **<p>:** The <p> is a paragraph tag which is used to display the character, word, text, sentence or paragraph in the web page and it is a paired tag. It will occupy the entire width of your screen to display the paragraph and then it will break the paragraph to next line.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Paragraph_tag</title>
</head>
<body>
      <p>One of the Unique features of Tap Academy is the use of Augmented Reality in our training. Through this concepts are visualized using cutting-edge animation to ensure an in-depth understanding of the concepts & experience the pure joy of learning as well as upskilling your coding skills
     </p>
</body>
</html>
```

Output:

One of the Unique features of Tap Academy is the use of Augmented Reality in our training. Through this concepts are visualized using cutting-edge animation to ensure an in-depth understanding of the concepts & experience the pure joy of learning as well as upskilling your coding skills

---

- **<pre>:** The <pre> is a preformatted tag which is used to display the character, word, text, sentence or paragraph in the webpage same as given in the statement(code). This type of tag is used most to display the poems, short statements etc.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Preformatted_tag</title>
</head>
<body>
  <pre>
  Twinkle, twinkle, little star
  How I wonder what you are
  Up above the world so high
  Like a diamond in the sky
  Twinkle, twinkle little star
  How I wonder what you are
  </pre>
</body>
</html>
```

Output:

- **
:** The
 tag stands for break and it is used to break the link in between the paragraph and it is an unpaired tag.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Break_tag</title>
</head>
<body>
  <p>
      One of the Unique features of Tap Academy is the use of
      <br>Augmented Reality in our training.Through this concepts are
      visualized using cutting-edge animation<br> to ensure an in-depth
      understanding of the concepts & experience the pure joy<br> of
      learning as well as upskilling your coding skills
  </p>
</body>
</html>
```

Note: In the above example we are using a paragraph tag so it will occupy the entire width of the screen to display the paragraph, but we are using the
 tag to explicitly break the line as per our requirement.

Output:

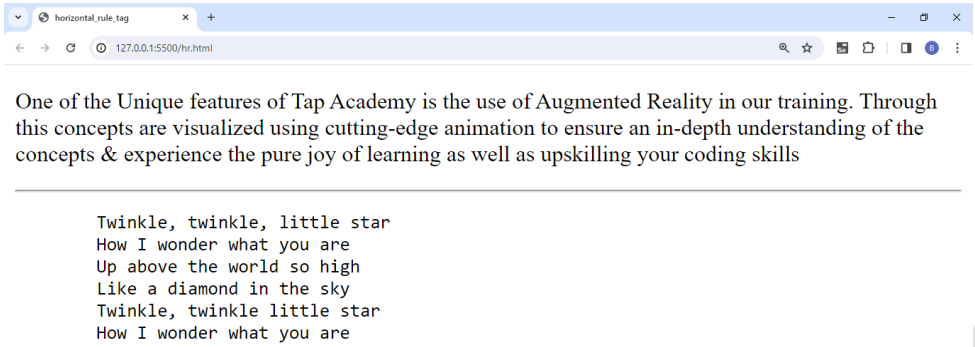
One of the Unique features of Tap Academy is the use of
Augmented Reality in our training.Through this concepts are visualized using cutting-edge animation
to ensure an in-depth understanding of the concepts & experience the pure joy
of learning as well as upskilling your coding skills

- **<hr>:** The <hr> tag stands for horizontal rule and it is used to give one horizontal in between the contents in the webpage to differentiate between the contents and it is an unpaired tag.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>horizontal_rule_tag</title>
</head>
<body>
  <p>
      One of the Unique features of Tap Academy is the use of Augmented
      Reality in our training.  Through this concepts are visualized
      using cutting-edge animation to ensure an in-depth understanding
      of the concepts & experience the pure joy of learning as well as
      upskilling your coding skills
  </p>
  <hr>
  <pre>
    Twinkle, twinkle, little star
    How I wonder what you are
    Up above the world so high
    Like a diamond in the sky
    Twinkle, twinkle little star
    How I wonder what you are
  </pre>
</body>
</html>
```

Output:

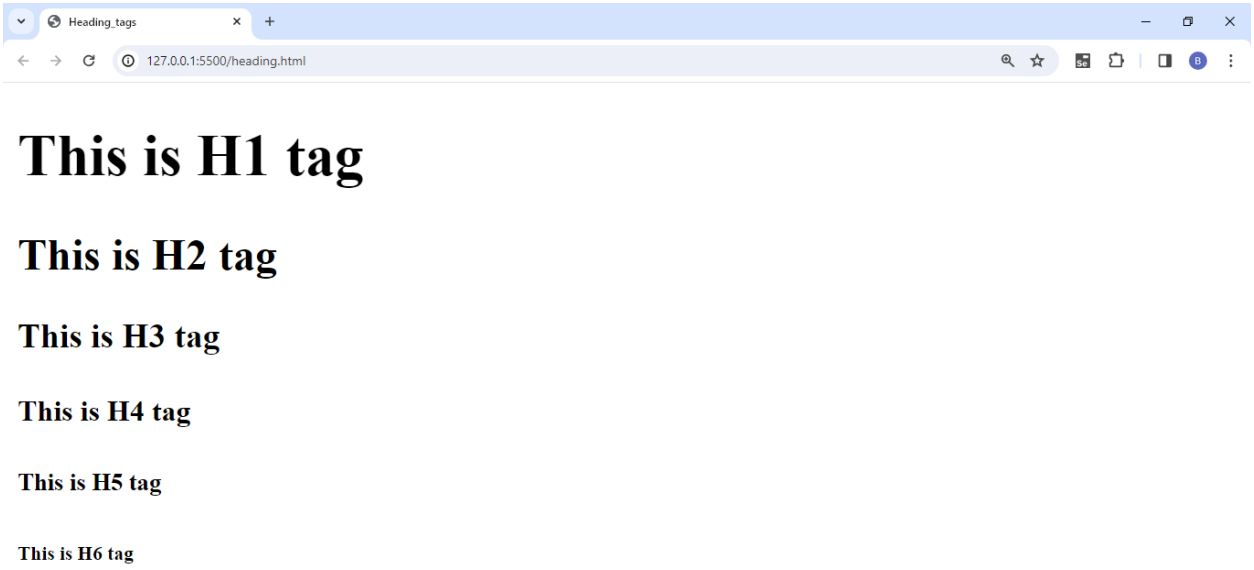


- **heading:** The heading is used to display the headings in the webpage. There are 6 types of heading tags such as: <h1>, <h2>, <h3>, <h4>, <h5>, <h6>. All these tags are differentiated by the size. The most used heading tag is <h1> tag and least used heading tag is <h6> tag. All these tags are paired tags.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Heading_tags</title>
</head>
<body>
  <h1>This is H1 tag</h1>
  <h2>This is H2 tag</h2>
  <h3>This is H3 tag</h3>
  <h4>This is H4 tag</h4>
  <h5>This is H5 tag</h5>
  <h6>This is H6 tag</h6>
</body>
</html>
```

Output:



- **<img>:** The <img> tag stands for image and it is used to display the image in the webpage. It has some main attributes such as:
  - **src**: It stands for source where the name of the image should be specified.
  - **alt**: It stand for alternate which is used to display any message if the image is not displayed in the webpage.
  - **height:** It is used to set the proper height of the image to display in the webpage
  - **width:** It is used to set the proper width of the image to display in the webpage

Alternate where the message is given to display if image is not working.

```
<img src="Tap_academy_logo.webp" alt="Image not found">
```

Source where the name of the image is specified.

Note: HTML attributes are used to describe the characteristic of an HTML element in detail (provide more information about the tag).

Example:

```
<!DOCTYPE html>
<html>
<head>
   <title>Image_tag</title>
</head>
<body>
  <img src="Tap_academy_logo.webp" alt="Image not found">
</body>
</html>
```

Note: image format can be anything like png or jpg etc, but make sure your browser supports that format and height and width are optional if required use it and set them accordingly. And make sure that image should be present in the current folder in which you're working.

Output:

➢ **<audio>:** The <audio> tag is used to display the audio in the webpage. It has some attributes such as:

- **src**: It stands for source where the name of the audio should be specified.
- **controls:** It is used to display all the controls of the audio such as play, pause etc.

Controls to display the controls of audio like play, pause.

```
<audio src="aud.mp3" controls></audio>
```

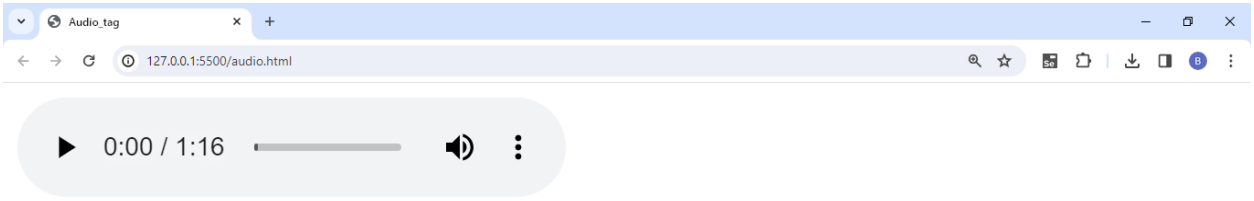
Source where the name of the audio is specified.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Audio_tag</title>
</head>
<body>
  <audio src="aud.mp3" controls></audio>
</body>
</html>
```

Note: audio format can be anything like but make sure your browser supports. And make sure that audio should be present in the current folder in which you're working. You can used height and width attributes if its required.

Output:



- **<video>:** The <video> tag is used to display the video in the webpage. It has some attributes such as:
  - **src**: It stands for source where the name of the video should be specified.
  - **controls:** It is used to display all the controls of the video such as play, pause etc.

Source where the name of the video is specified.

width is used to set the width of the video.

```
<video src="LAXMI KHOT 1.mp4" controls width="50%" height="160px"
</video>
```

Controls to display the controls of audio like play, pause.

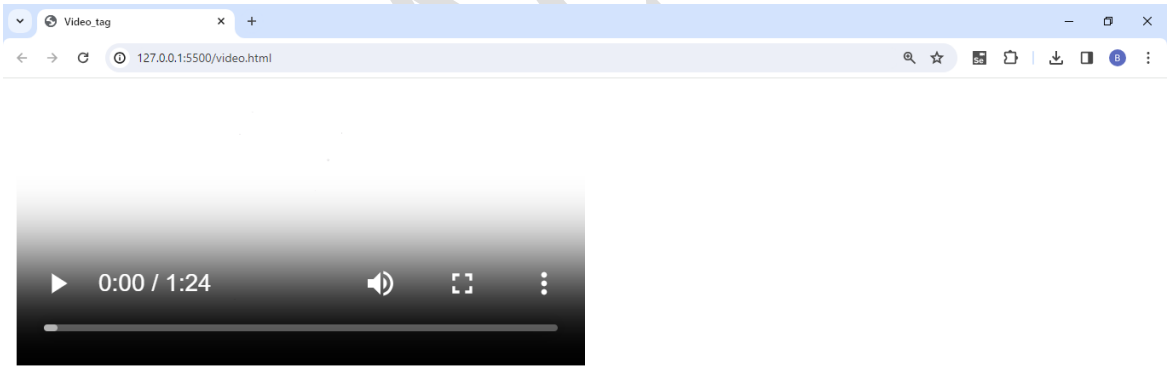
height is used to set the height of the video.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <title>Video_tag</title>
</head>
<body>
  <video src="LAXMI KHOT 1.mp4" controls width="50%"
height="160px"></video>
</body>
</html>
```

Note: video format can be anything like but make sure your browser supports. And make sure that video should be present in the current folder in which you're working. You can used height and width attributes if its required.

Output:



FORMATTING TAGS

HTML provides us ability to format text without using CSS. By using the below formatting tags.

- **<u>**
- **<b>**
- **<i>**
- **<mark>**
- **<sub>**
- **<sup>**
- **<small>**
- **<big>**
- **<s>**

➢ **<u>:** The <u> tag stands for underline, it is used to underline the character, word or sentence.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
    <p>A <u>Full-stack developer</u> is a person who has great
    knowledge of the latest and upcoming trends of web development
    With complete knowledge of front-end and back-end web
    development.
    </p>
</body>
</html>
```

In the above example we are taking one paragraph and in that we are underlining the Full-stack developer.

Output:

A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

- **<b>:** The <b> tag stands for bold, it is used to bold the character, word or sentence.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
   <p>A Full-stack developer is a person who has <b>great knowledge</b> of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.
   </p>
</body>
</html>
```

Output:

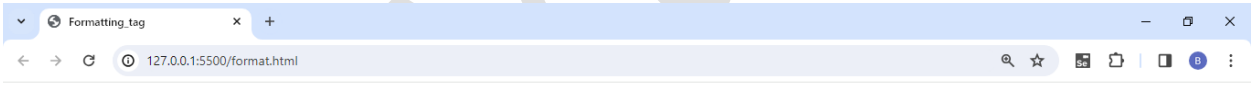
A Full-stack developer is a person who has **great knowledge** of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

- **<i>:** The <i> tag stands for italic, it is used to change the font style to italic of the character, word or sentence.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
  <p>A Full-stack developer is a person who has great knowledge
      of the latest and upcoming trends of <i>web development</i> with
      complete knowledge of front-end and back-end web development.
   </p>
</body>
</html>
```

Output:



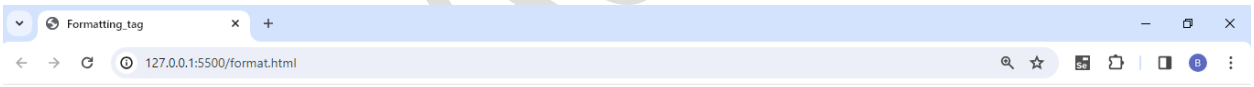
A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of *web development* with complete knowledge of front-end and back-end web development.

- **<mark>:** The <mark> is used to highlight the character, word or sentence by default the background color that will be displayed in yellow color.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
  <p>A Full-stack developer is a person who has great knowledge
     of the latest and upcoming trends of web development with
     complete <mark>knowledge</mark> of front-end and back-end
     web development.
  </p>
</body>
</html>
```

Output:



A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

➢ **<sub>:** The <sub> tag stands for subscript it is used to display the subscript character, word or sentences as show below:

front-end and $_{\text{back-end}}$ web development.

Subscript text or word

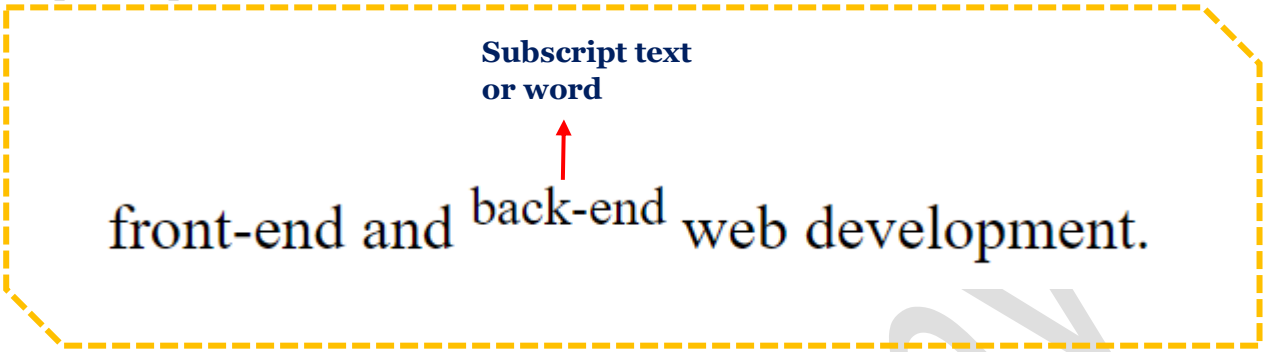
Example:

```
<!DOCTYPE html>
<html>
<head>
   <title>Formatting_tag</title>
</head>
<body>
   <p>A Full-stack developer is a person who has great knowledge
        of the latest and upcoming trends of web development with
        complete knowledge of front-end and <sub>back-end</sub>
        web development.
   </p>
</body>
</html>
```

Output:

A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and $_{\text{back-end}}$ web development.

➢ **<sup>:** The <sup> tag stands for superscript it is used to display the superscript character, word or sentences as show below:



Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
  <p>A Full-stack developer is a person who has great knowledge
     of the latest and upcoming trends of web development with
     complete knowledge of front-end and <sup>back-end</sup>
     web development.
  </p>
</body>
</html>
```

Output:

A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

➢ **<small>:** The <small> tag is used to display the character, word or sentences in small size compared to others as show below:

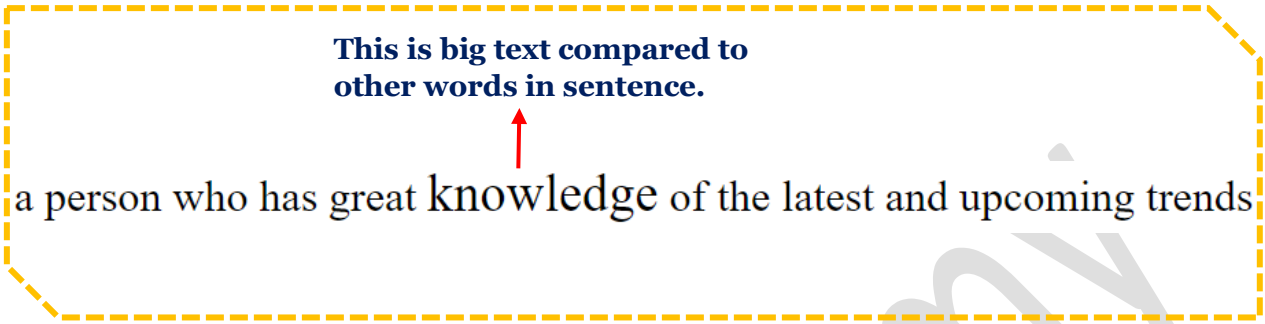
Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
  <p>A Full-stack developer is a person who has great knowledge
      of the latest and <small>upcoming</small> trends of web
     development with complete knowledge of front-end and back-
     end web development.
  </p>
</body>
</html>
```

Output:

A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

➢ **<big>:** The <big> tag is used to display the character, word or sentences in big size compared to other words as show below:



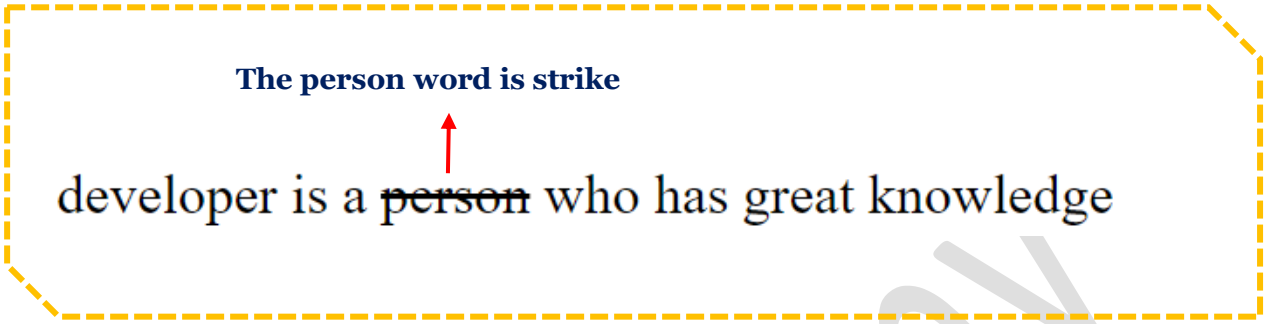
Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
      <p>A Full-stack developer is a person who has great
      <big>knowledge</big> of the latest and upcoming trends of web
      development with complete knowledge of front-end and back-
      end web development.
      </p>
</body>
</html>
```

Output:

A Full-stack developer is a person who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.

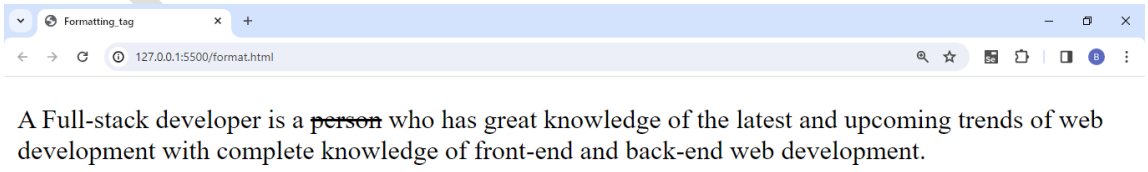
➢ **<s>:** The <s> tag stands for strike and it is used to strike the character, word or as show below:



Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Formatting_tag</title>
</head>
<body>
      <p>A Full-stack developer is a <s>person</s> who has great knowledge of the latest and upcoming trends of web development with complete knowledge of front-end and back-end web development.
      </p>
</body>
</html>
```

Output:



LINKS

- Links are hyperlinks which can be clicked and redirect from one page to another page.
- Links can be anything like text, image or any other elements.
- If any text or image or any other element is link as soon as you keep your cursor on that it will change into little hand.
- To create a link, we need to use the <a> anchor tag.

href is a hypertext reference where the path is given to redirect.

```
<a href="https://www.flipkart.com/">Visit Flipkart</a>
```

The text which will be displayed as a link

- The link has three default colors as follows:
  Unvisited----------Blue color
  Visited----------Purple color
  Active----------Red color

Example for Unvisited link:

```
<!DOCTYPE html>
<html>
<head>
  <title>Links</title>
</head>
<body>
  <a href="https://www.flipkart.com/">Visit Flipkart</a>
</body>
</html>
```

Output:

Visit Flipkart

Note: The above link is unvisited hence it is displaying in blue color.

Example for Visited link:

```
<!DOCTYPE html>

<html>
<head>
  <title>Visited</title>
</head>
<body>
  <a href="https://www.flipkart.com/">Visit Flipkart</a>
</body>
</html>
```

Output:

Visit Flipkart

Note: The above link is visited hence it is displaying in purple color.

Example for Active link:

```
<!DOCTYPE html>

<html>
<head>
  <title>Active</title>
</head>
<body>
  <a href="https://www.flipkart.com/">Visit Flipkart</a>
</body>
</html>
```

Output:

Visit Flipkart

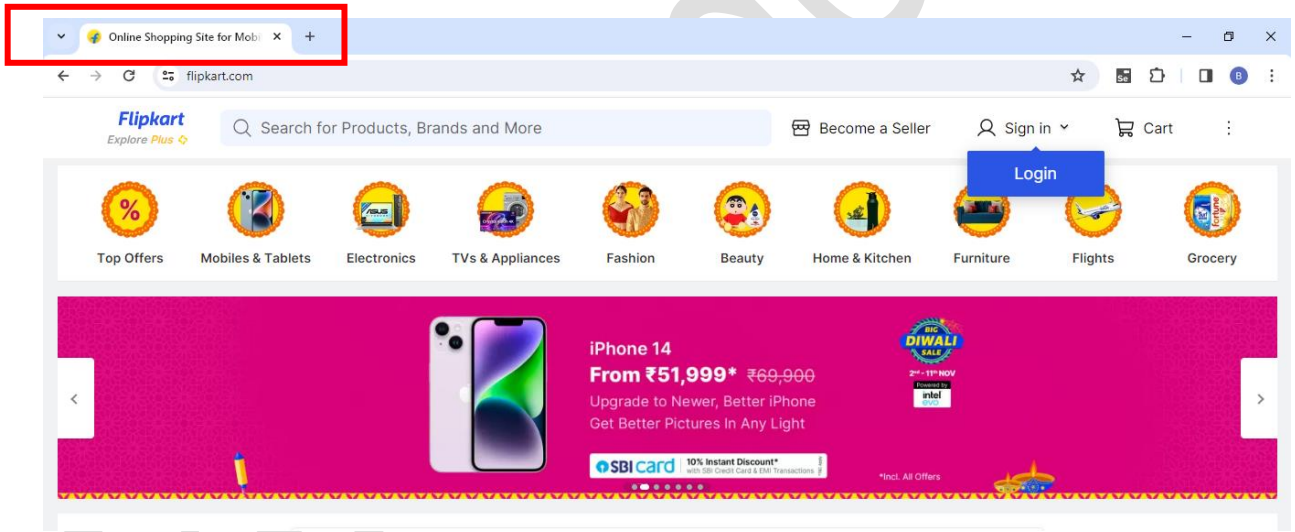
Note:

- The above link is active hence it is displaying in red color.
- For the anchor tag we have target attribute which helps us to open the webpage in same page or in new tab:
  Values: _blank will open the webpage in new tab.
  _self will open the webpage in the same tab only.
- By default, the link(webpage) will open in the same tab only.

Example for _self:

```
<!DOCTYPE html>
<html>
<head>
   <title>self</title>
</head>
<body>
   <a href="https://www.flipkart.com/" target="_self">Visit
Flipkart</a>
</body>
</html>
```

Output:

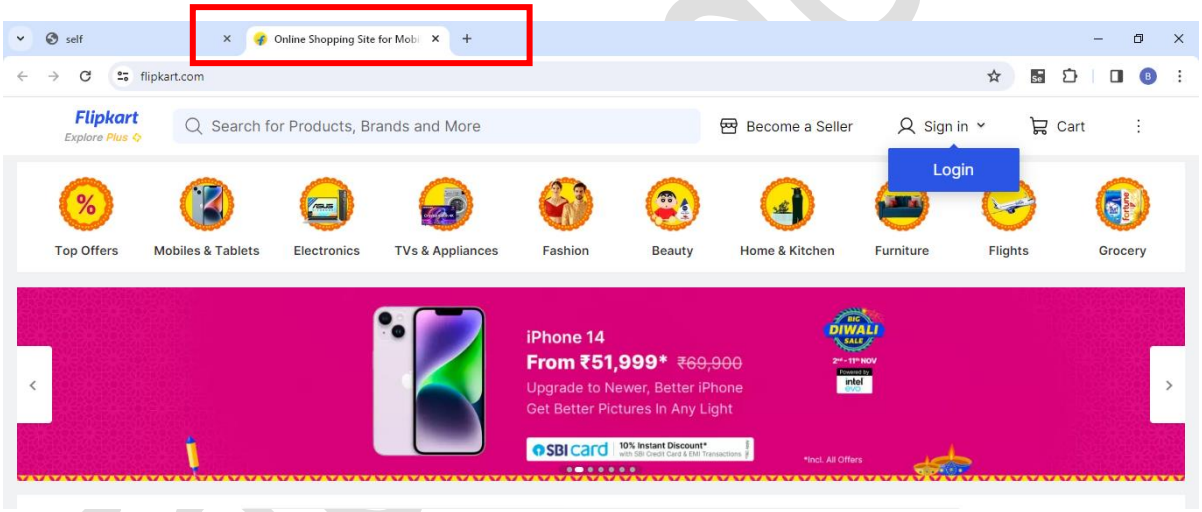


Note: In the above example since we have used as _self value for target attribute the page is displaying the same tab only.

Example for _blank:

```
<!DOCTYPE html>
<html>
<head>
   <title>self</title>
</head>
<body>
   <a href="https://www.flipkart.com/" target="_blank">Visit
Flipkart</a>
</body>
</html>
```

Output



Note: In the above example since we have used as _blank value for target attribute the page is displaying in the new tab.

Image as a Link Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Image_as_link</title>
</head>
<body>
<a href="https://thetapacademy.com/">
  <img src="Tap_academy_logo.webp" alt="Tap_logo">
</a>
</body>
</html>
```

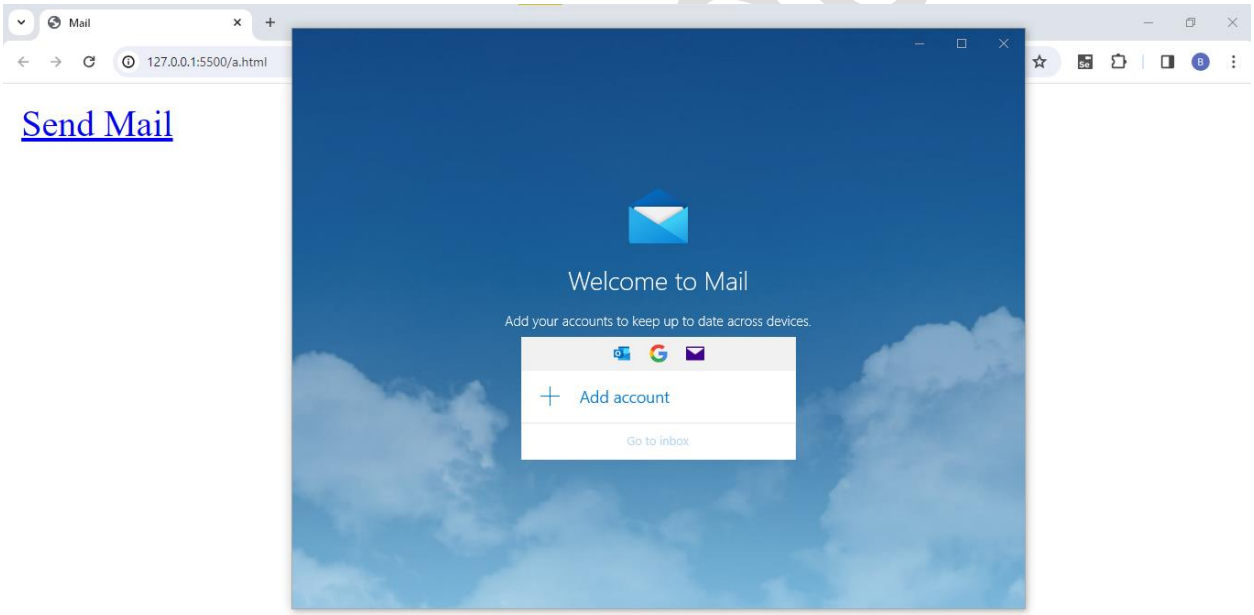
Output:

Note: In the above example inside the anchor tag as a content we are giving image instead of text.

Mail as a Link Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Mail</title>
</head>
<body>
  <a href="mailto:tapacademy@gmail.com"  target="_blank">
  Send Mail</a>
</body>
</html>
```

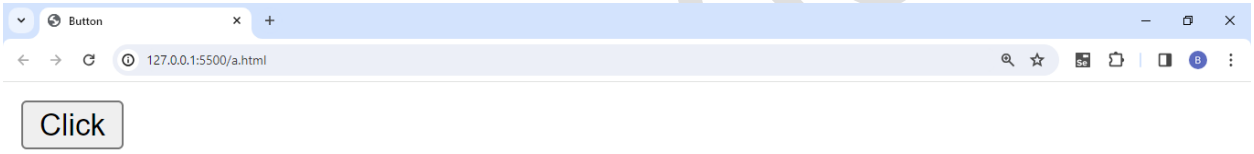
Output:



Button as a Link Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Button</title>
</head>
<body>
  <a href="https://www.flipkart.com/" target="_blank">
    <button>Click</button>
  </a>
</body>
</html>
```

Output:



Note: In the href as a link we can give absolute url (Uniform Resource Locator) or relative url.

Absolute Url: An absolute URL contains all the information necessary to locate a resource.

Example: https://www.flipkart.com/

Relative Url: A relative URL typically contains only the path to a specific file.

Example: /images/photo.png

TABLE

- The table contains the rows and column in a webpage.
- We need to use the **<table>** tag to start creating the table.
- We need to use the **<tr>** tag which stands for table row, and it is used to specify the row in a table.
- We need to use the **<th>** tag which stands for table header, and it is used to specify the header(column) in a table.
- We need to use the **<td>** tag which stands for table data, and it is used to specify the table data in a table.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Table</title>
</head>
<body>
   <table>
      <tr>
        <th>Student_Id</th>
        <th>Student_Name</th>
        <th>Student_Age</th>
        <th>Student_Gender</th>
      </tr>
      <tr>
        <td>101</td>
        <td>Ravi</td>
        <td>23</td>
        <td>Male</td>
      </tr>
```

```
        <tr>
          <td>102</td>
          <td>Ramya</td>
          <td>23</td>
          <td>Female</td>
        </tr>
        <tr>
          <td>103</td>
          <td>Sam</td>
          <td>24</td>
          <td>Male</td>
        </tr>
        <tr>
          <td>104</td>
          <td>Roopa</td>
          <td>23</td>
          <td>Female</td>
        </tr>
        <tr>
          <td>105</td>
          <td>Ram</td>
          <td>23</td>
          <td>Male</td>
        </tr>
      </table>
    </body>
    </html>
```

Note: If we create a table just by specifying the row, header and table data as above example the result will be as shown below:

Student_Id	Student_Name	Student_Age	Student_Gender
101	Ravi	23	Male
102	Ramya	23	Female
103	Sam	24	Male
104	Roopa	23	Female
105	Ram	23	Male

In order to get the proper table look we need to use **border attribute** in the table tag. **Example:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Table</title>
</head>
<body>
   <table border>
      <tr>
         <th>Student_Id</th>
         <th>Student_Name</th>
         <th>Student_Age</th>
         <th>Student_Gender</th>
      </tr>
      <tr>
         <td>101</td>
         <td>Ravi</td>
         <td>23</td>
         <td>Male</td>
      </tr>
      <tr>
         <td>102</td>
         <td>Ramya</td>
         <td>23</td>
         <td>Female</td>
      </tr>
      <tr>
         <td>103</td>
         <td>Sam</td>
         <td>24</td>
         <td>Male</td>
      </tr>
      <tr>
         <td>104</td>
         <td>Roopa</td>
         <td>23</td>
         <td>Female</td>
```

```
        </tr>
        <tr>
          <td>105</td>
          <td>Ram</td>
          <td>23</td>
          <td>Male</td>
        </tr>
      </table>
    </body>
    </html>
```

Note: If we create a table using the border attribute and specify the row, header and table data as above example the result will be as shown below:

Student_Id	Student_Name	Student_Age	Student_Gender
101	Ravi	23	Male
102	Ramya	23	Female
103	Sam	24	Male
104	Roopa	23	Female
105	Ram	23	Male

Creation of vertical table Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Vertical_table</title>
</head>
<body>
  <table border>
    <tr>
      <th>Student_Id</th>
      <td>101</td>
      <td>102</td>
      <td>103</td>
      <td>104</td>
      <td>105</td>
    </tr>
    <tr>
      <th>Student_Name</th>
      <td>Ravi</td>
      <td>Ramya</td>
      <td>Sam</td>
      <td>Roopa</td>
      <td>Ram</td>
    </tr>
    <tr>
      <th>Student_Age</th>
      <td>23</td>
      <td>23</td>
      <td>24</td>
      <td>23</td>
      <td>23</td>
    </tr>
    <tr>
      <th>Student_Gender</th>
      <td>Male</td>
      <td>Female</td>
      <td>Male</td>
      <td>Female</td>
```

```
        <td>Male</td>
      </tr>
    </table>
  </body>
  </html>
```

Output:

Student_Id	101	102	103	104	105
Student_Name	Ravi	Ramya	Sam	Roopa	Ram
Student_Age	23	23	24	23	23
Student_Gender	Male	Female	Male	Female	Male

Note: We can provide the table caption as well by using the **<caption> tag.**

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Vertical_table</title>
</head>
<body>
  <table border>
    <caption>Student_Details</caption>
    <tr>
      <th>Student_Id</th>
      <td>101</td>
      <td>102</td>
      <td>103</td>
      <td>104</td>
```

```
            <td>105</td>
          </tr>
          <tr>
            <th>Student_Name</th>
            <td>Ravi</td>
            <td>Ramya</td>
            <td>Sam</td>
            <td>Roopa</td>
            <td>Ram</td>
          </tr>
          <tr>
            <th>Student_Age</th>
            <td>23</td>
            <td>23</td>
            <td>24</td>
            <td>23</td>
            <td>23</td>
          </tr>
          <tr>
            <th>Student_Gender</th>
            <td>Male</td>
            <td>Female</td>
            <td>Male</td>
            <td>Female</td>
            <td>Male</td>
          </tr>
        </table>
      </body>
      </html>
```


Output:

Student_Details

Student_Id	101	102	103	104	105
Student_Name	Ravi	Ramya	Sam	Roopa	Ram
Student_Age	23	23	24	23	23
Student_Gender	Male	Female	Male	Female	Male

Example for merging multiple columns as one:

Note: We need to use the **colspan** attribute to merge the column and display as one.

```
<!DOCTYPE html>
<html>
<head>
  <title>colspan</title>
</head>
<body>
  <table border>
    <tr>
      <th colspan="2">Full Name</th>
      <th>Rating</th>
    </tr>
    <tr>
      <td>Ravi</td>
      <td>V</td>
      <td>7.8</td>
    </tr>
    <tr>
      <td>Ramya</td>
      <td>Suresh</td>
      <td>8</td>
    </tr>
    <tr>
      <td>Sachin</td>
      <td>Shekar</td>
      <td>6.7</td>
    </tr>
    <tr>
      <td>Smitha</td>
      <td>R</td>
      <td>9</td>
```

```
            </tr>
            <tr>
              <td>Divya</td>
              <td>D</td>
              <td>9</td>
            </tr>

          </table>
        </body>
        </html>
```

In the above example we have merged two columns as one by using the colspan attribute and by specifying the value as 2.

Output:

Full Name		Rating
Ravi	V	7.8
Ramya	Suresh	8
Sachin	Shekar	6.7
Smitha	R	9
Divya	D	9

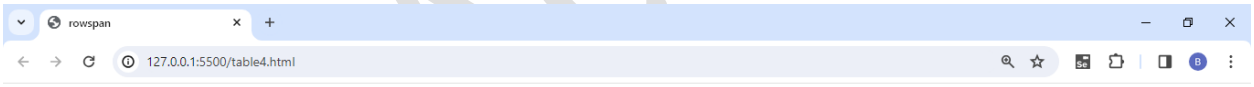
Example for merging multiple columns and rows as one:

Note: We need to use the **rowspan** attribute to merge the rows and display as one.

```
<!DOCTYPE html>
<html>
<head>
  <title>rowspan</title>
</head>
<body>
  <table border>
    <tr>
      <th rowspan="5">Full Name</th>
      <td>Ravi</td>
      <td>V</td>
    </tr>
    <tr>
      <td>Ramya</td>
      <td>Suresh</td>
    </tr>
    <tr>
      <td>Sachin</td>
      <td>Shekar</td>
    </tr>
    <tr>
      <td>Smitha</td>
      <td>R</td>
    </tr>
    <tr>
      <td>Divya</td>
      <td>D</td>
    </tr>
    <tr>
```

```
            <th rowspan="5">Rating</th>
            <td colspan="2">7.8</td>
          </tr>
          <tr>
            <td colspan="2">8</td>
          </tr>
          <tr>
            <td colspan="2">6.7</td>
          </tr>
          <tr>
            <td colspan="2">9</td>
          </tr>
          <tr>
            <td colspan="2">9</td>
          </tr>
        </table>
      </body>
      </html>
```

Output:



Full Name	Ravi	V
	Ramya	Suresh
	Sachin	Shekar
	Smitha	R
	Divya	D
Rating	7.8	
	8	
	6.7	
	9	
	9	

LIST

- In HTML a list will allow us to display the group of items as list.
- There are two types of list:
  - Ordered List
  - Unordered List
  - Description List

Ordered List

- We can create the ordered list by using the <ol> tag.
- We need to put the items inside the list by using the <li> tag.
- By default, the ordered list will be display as 1,2,3....

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Ordered List</title>
</head>
<body>
  <ol>
    <li>Java</li>
    <li>SQL</li>
    <li>Python</li>
    <li>C++</li>
  </ol>
</body>
</html>
```

Output:

1. Java
2. SQL
3. Python
4. C++

Note: we can change the default numbering for ordered list by using the type attribute.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Ordered List</title>
</head>
<body>
  <ol type="A">
    <li>Java</li>
    <li>SQL</li>
    <li>Python</li>
    <li>C++</li>
  </ol>
</body>
</html>
```

Output:

A. Java
B. SQL
C. Python
D. C++

Unordered List

- We can create the unordered list by using the <ul> tag.
- We need to put the items inside the list by using the <li> tag.
- By default, the unordered list will be display in bullets.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <title>Unordered List</title>
</head>
<body>
  <ul>
   <li>HTML</li>
   <li>CSS</li>
   <li>JavaScript</li>
   <li>Bootstrap</li>
  </ul>
</body>
</html>
```

Output:

- HTML
- CSS
- JavaScript
- Bootstrap

Note: In HTML we cannot the default bullets to any other, if we want to do it then we need to apply the styling for that.

Description List

- We can create the description list by using the <dl> tag.
- We can give the description term by using the <dt> tag.
- We can give the definition for the description term by using the <dd> tag.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Description List</title>
</head>
<body>
  <dl>
    <dt>Java</dt>
    <dd>Java is a object oriented programming language</dd>
    <dt>SQL</dt>
    <dd>SQL stands for structured query language</dd>
    <dt>HTML</dt>
    <dd>HTML stands for Hypertext markup language</dd>
```

```
    </dl>
</body>
</html>
```

Output:

Java
    Java is a object oriented programming language
SQL
    SQL stands for structured query language
HTML
    HTML stands for Hypertext markup language

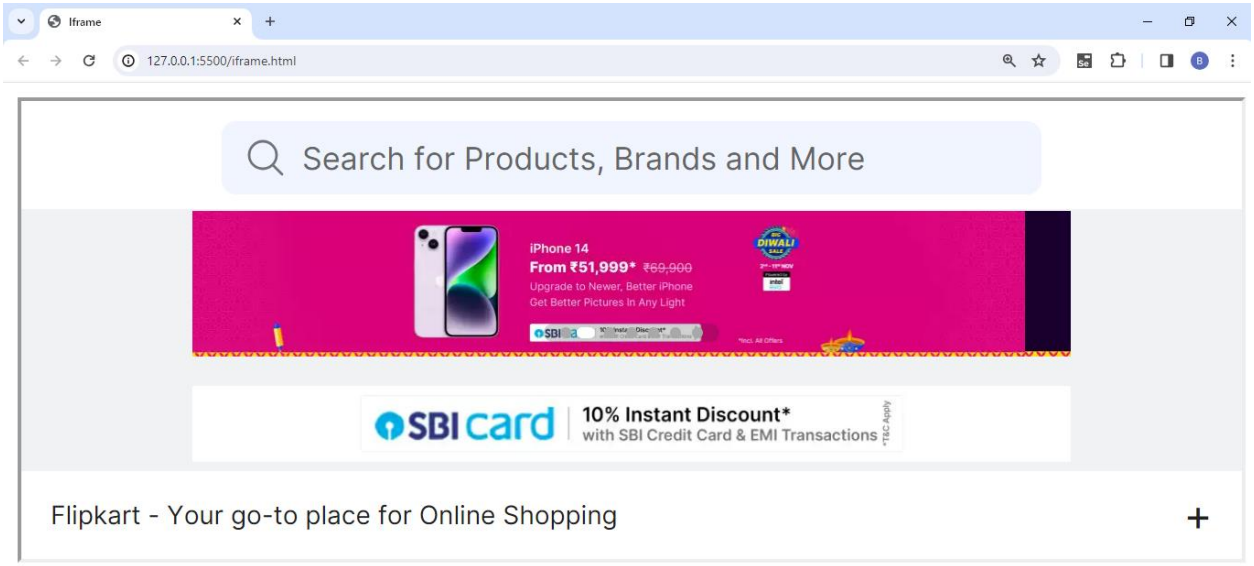
Iframe

- Iframe is used to display a webpage inside a webpage as embedded or nested.
- We can achieve this by using the <iframe> tag.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Iframe</title>
</head>
<body>
  <iframe src="https://www.flipkart.com/" height="250px"
width="100%"></iframe>
</body>
</html>
```

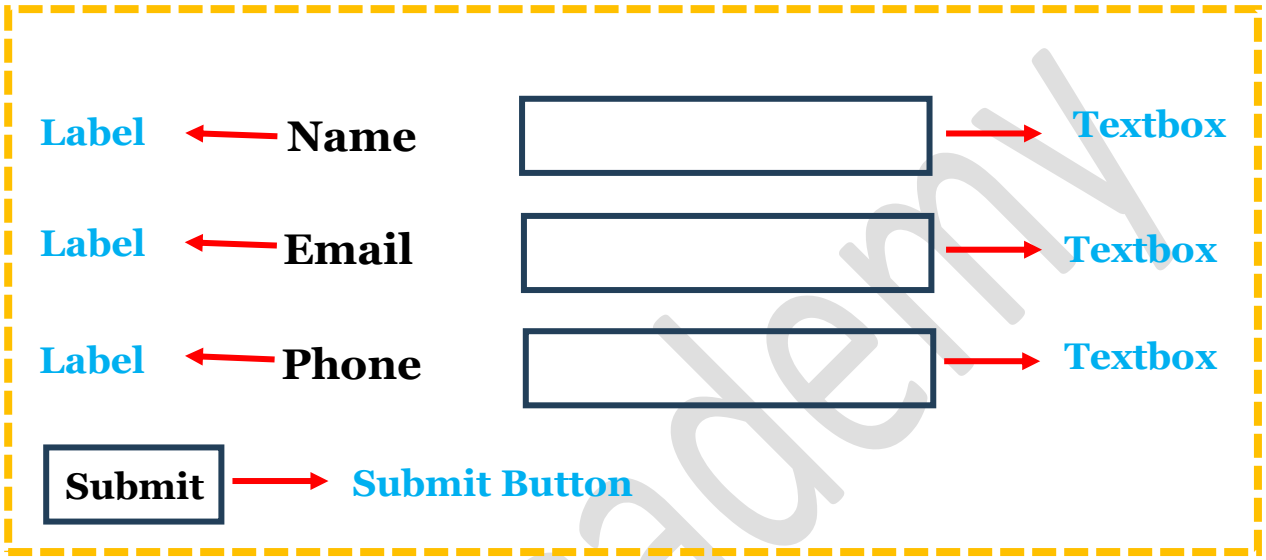
Output:



In the above example we are embedding the flipkart website inside our webpage by using the <iframe> tag, hence the entire website we can go through is in our webpage only and we can set the proper height and width of the website.

Form

- The form is used to collect the data or information from the user side.
- We can create a form by using the <form> tag.
- Form has the structure as shown below:



Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Form</title>
</head>
<body>
  <form>
    <label>Name</label>
    <input type="text"><br><br>
    <label>Email</label>
    <input type="text"><br><br>
    <label>Phone</label>
    <input type="text"><br><br>
    <input type="submit">
  </form>
</body></html>
```

Output:

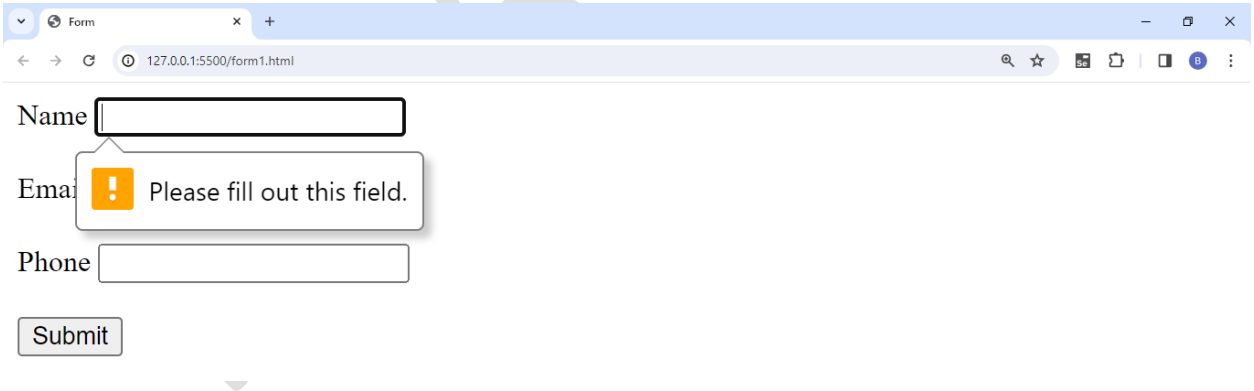
Note:

- To make the text fields mandatory or important to fill we can use the **required attribute**.
- Once we click on the submit button if every data is correct, we can perform any action by using the **action attribute.**
- To identify the textbox uniquely we can use the **id attribute** and assign the value for it and by keeping that value we can identify them uniquely.
- To tell the textbox is to created to which label we can use the **for attribute** in that we can give the **id value** so that it will map it properly.
- According the input can change the type of the input box.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Form</title>
</head>
<body>
  <form action="https://www.thetapacademy.com">
    <label for="name">Name</label>
    <input type="text" required id="name"><br><br>
    <label for="email">Email</label>
    <input type="email" required id="email"><br><br>
    <label for="phone">Phone</label>
    <input type="text" required id="phone"><br><br>
    <input type="submit">
  </form>
</body>
</html>
```

Output:



- You can see in the above example once we give as required if we don't fill that field, it will be displaying a message as "Please fill out this field".
- Once all the fields are filled properly then only you can submit the form and you can perform the given option.

Creating the Checkbox in Form

- In the forms usually we use a checkbox button to allow only **multiple option** to click like **Skills, Course obtained etc.**
- To create the checkbox the input type we need to give as **checkbox**.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Checkbox</title>
</head>
<body>
   <form action="https://www.thetapacademy.com">
      <label>Skills</label><br><br>
      <input type="checkbox" required id="Java">
      <label for="Java">Java</label><br><br>
      <input type="checkbox" required id="SQL">
      <label for="SQL">SQL</label><br><br>
      <input type="checkbox" required id="HTML">
      <label for="HTML">HTML</label><br><br>
      <input type="checkbox" required id="CSS">
      <label for="CSS">CSS</label><br><br>
      <input type="checkbox" required id="JavaScript">
      <label for="JavaScript">JavaScript</label><br><br>
      <input type="submit">
   </form>
</body>
</html>
```

Output:

Creating the Radio Buttons in Form

- In the forms usually we use a radio button to allow only **one option** to click like **gender, department** etc.
- To create the radio button the input type we need to give as **radio**.
- When ever we are creating the radio buttons if it should allow only one option the all the provided options, we need to **group it as one**, for this we need to use the name attribute for every text box and value of that **name attribute** should be same.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Radio Button</title>
</head>
<body>
  <form action="https://www.thetapacademy.com">
    <label>Gender</label><br><br>
    <input type="Radio" required id="Male" name="gender">
    <label for="Male">Male</label>
    <input type="Radio" required id="Female" name="gender">
    <label for="Female">Female</label>
    <input type="Radio" required id="Others" name="gender">
    <label for="Others">Others</label><br><br>
    <input type="submit">
  </form>
</body>
</html>
```

Output:

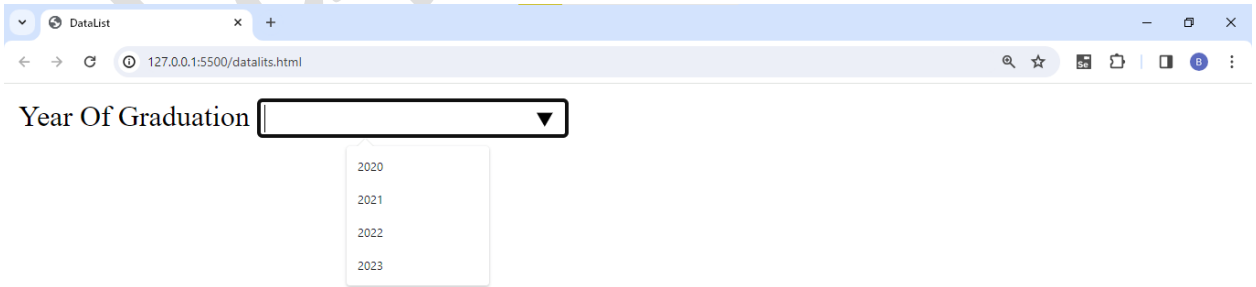
Creating the Data List in Form

- To create the data list in the forms we need to use the **<datalist> tag** and inside that we need to use the **<option> tag** to specify the options.
- To create this, we need to use the **list attribute** and the value of **list attribute** for **input** and value of **id attribute** in **<datalist> tag** should be same.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>DataList</title>
</head>
<body>
  <label>Year Of Graduation</label>
  <input list="YOG">
  <datalist id="YOG">
    <option>2020</option>
    <option>2021</option>
    <option>2022</option>
    <option>2023</option>
  </datalist>
</body>
</html>
```

Output:



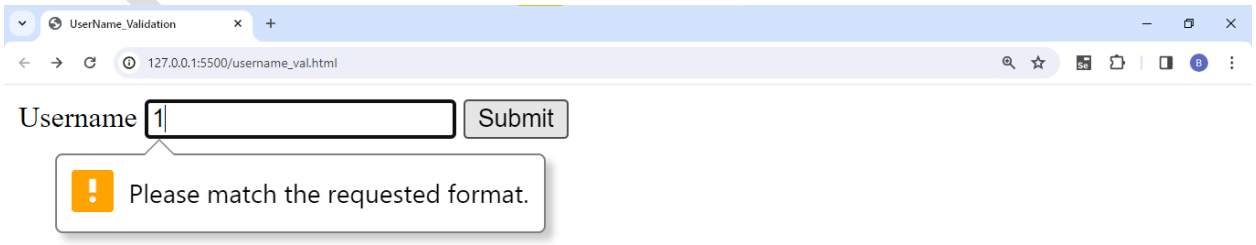
UserName Validation

- To validate the username, in html we have to use pattern attribute and give the proper pattern to the username.
- Username should contain only the alphabets that is from lowercase a to z and uppercase A to Z.
- We should also tell the minimum and maximum characters it should accept.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>UserName_Validation</title>
</head>
<body>
  <form action="https://www.thetapacademy.com">
  <label for="uname">Username</label>
  <input type="text" required id="uname" pattern="[a-zA-z]{3,8}">
  <input type="submit">
  </form>
</body>
</html>
```

Output:



Note: In the above example as we have given the pattern, we have to match the data as per the given pattern only if we try to give any other format it will not allow.

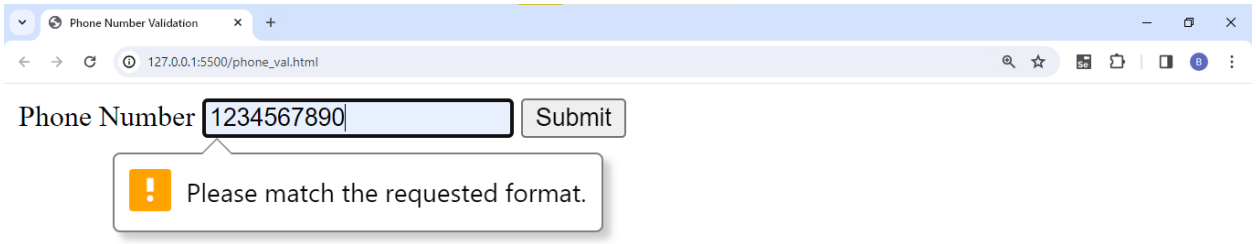
Phone Number Validation

- To validate the phone number, in html we have to use pattern attribute and give the proper pattern to the phone number.
- Phone Number should contain only the numbers, it should not contain any alphabets or special characters.
- Phone number should contain only 10 digits.
- Phone number first digit should start with either 6,7,8 or 9 and remaining 9 digits can be anything from 0 to 9.

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Phone Number Validation</title>
</head>
<body>
  <form>
    <label for="phone">Phone Number</label>
    <input type="text" required id="phone" pattern="[6-9]{1}[0-
9]{9}">
    <input type="submit">
  </form>
</body>
</html>
```

Output:



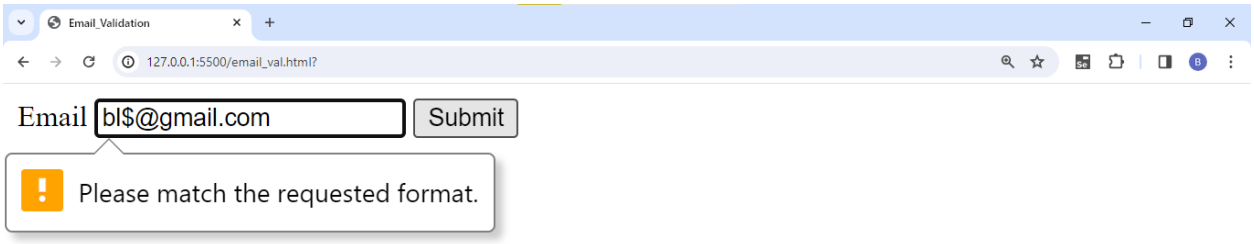
Note: In the above example as we have given the pattern, we have to match the data as per the given pattern only if we try to give any other format it will not allow so the phone number should contain only 10 digits and in that the first digit should be in range 6 to 9 and remaining 9 digits can be in range 0 to 9.

Email Validation

Example:

```
<!DOCTYPE html>
<html>
<head>
  <title>Email_Validation</title>
</head>
<body>
   <form action="">
      <label for="">Email</label>
      <input type="email" id="email" required pattern="[a-z0-
9_.]+@[a-z]+.[a-z]{2,}">
      <input type="submit">
   </form>
</body>
</html>
```

Output:

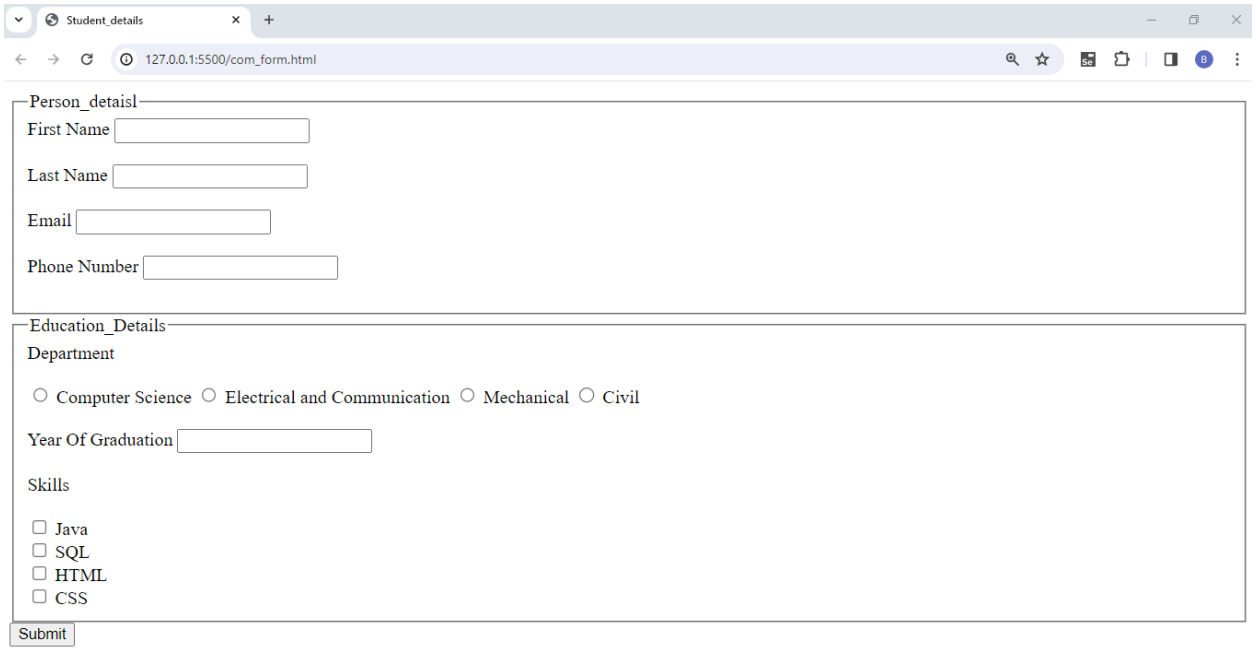


Example for student details form:

```
<!DOCTYPE html>
<html>
<head>
  <title>Student_details</title>
</head>
<body>
<form action="https://www.thetapacademy.com">
  <fieldset>
    <legend>Person_details</legend>
      <label for="fname">First Name</label>
      <input type="text" required id="fname" pattern="[a-zA-
z]{3,8}"><br><br>
      <label for="lname">Last Name</label>
      <input type="text" required id="lname" pattern="[a-zA-
z]{3,8}"><br><br>
      <label for="email">Email</label>
      <input type="email" required id="email" pattern="[a-z0-
9_.]+@[a-z]+.[a-z]{2,}
"><br><br>
      <label for="phone">Phone Number</label>
      <input type="text" required id="phone" pattern="[6-
9]{1}[0-9]{9}"><br><br>
  </fieldset>
  <fieldset>
    <legend>Education_Details</legend>
    <label>Department</label><br><br>
```

```
        <input type="radio" id="cs" name="dept">
        <label for="cs">Computer Science</label>
        <input type="radio" id="ec" name="dept">
        <label for="ec">Electrical and Communication</label>
        <input type="radio" id="mech" name="dept">
        <label for="mech">Mechanical</label>
        <input type="radio" id="civil" name="dept">
        <label for="civil">Civil</label><br><br>
        <label>Year Of Graduation</label>
        <input list="yog"><br><br>
        <datalist id="yog">
            <option>2020</option>
            <option>2021</option>
            <option>2022</option>
            <option>2023</option>
            <option>2024</option>
        </datalist>
        <label>Skills</label><br><br>
        <input type="checkbox" id="Java">
        <label for="Java">Java</label><br>
        <input type="checkbox" id="SQL">
        <label for="SQL">SQL</label><br>
        <input type="checkbox" id="HTML">
        <label for="HTML">HTML</label><br>
        <input type="checkbox" id="CSS">
        <label for="CSS">CSS</label><br>
    </fieldset>
    <input type="submit">
</form>
</body>
</html>
```

Output:



Note: In the above example to different and segregate the information of student we are using the **<fieldset> tag** and to give the name/label for that field we are using the **<legend> tag** which is a child tag of <fieldset> tag.