Chapter 1 Introduction of HTML and basic Tags

What is HTML?

HTML is an acronym which stands for **Hyper Text Markup Language** which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

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**.

Let's see a simple example of HTML.

```
<!DOCTYPE>
<html>
<head>
<title>Web page title</title>
</head>
<body>
<h1>Write Your First Heading</h1>
<p>Write Your First Paragraph.</p>
</body>
</html>
```

Description of HTML Example

<!DOCTYPE>: It defines the document type or it instruct the browser about the version of HTML.

<html > :This tag informs the browser that it is an HTML document. Text between html tag describes the web document. It is a container for all other elements of HTML except <!DOCTYPE>

<head>: It should be the first element inside the <html> element, which contains the metadata(information about the document). It must be closed before the body tag opens.

<title>: As its name suggested, it is used to add title of that HTML page which appears at the top of the browser window. It must be placed inside the head tag and should close immediately. (Optional)

<body> : Text between body tag describes the body content of the page that is visible to the end user. This tag contains the main content of the HTML document.

<h1> : Text between <h1> tag describes the first level heading of the webpage.

<p> : Text between <p> tag describes the paragraph of the webpage.

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.

Building blocks of HTML:

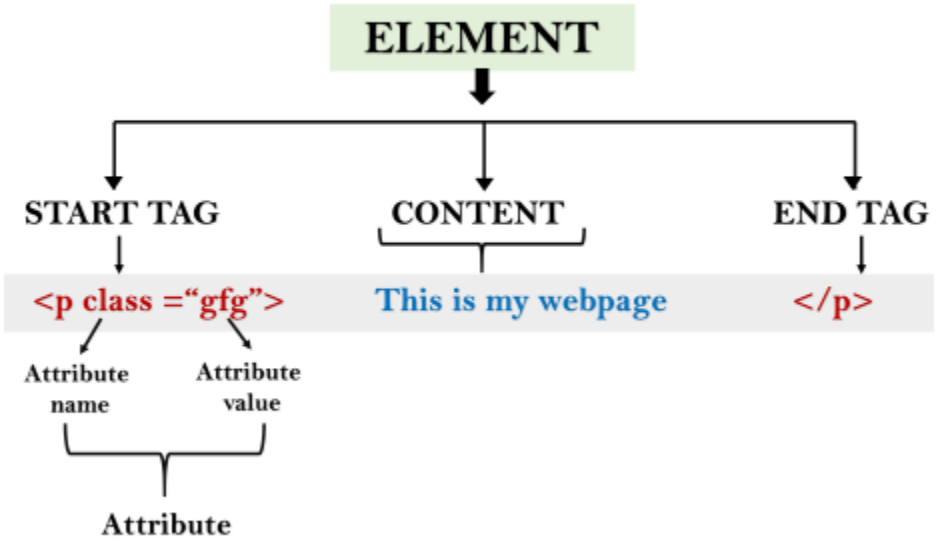
An HTML document consists of its basic building blocks which are:

- **Tags:** An HTML tag surrounds the content and applies meaning to it. It is written between < and > brackets.
- **Attribute:** An attribute in HTML provides extra information about the element, and it is applied within the start tag. An HTML attribute contains two fields: name & value.

Syntax

```
<tag name  attribute_name= " attr_value"> content </ tag name>
```

Elements: An HTML element is an individual component of an HTML file. In an HTML file, everything written within tags are termed as HTML elements.



Example :

```
<!DOCTYPE html>
<html>
  <head>
        <title>The basic building blocks of HTML</title>
 </head>
  <body>
      <h2>The building blocks</h2>
      <p>This is a paragraph tag</p>
      <p style="color: red">The style is attribute of paragraph tag</p>
      <span>The element contains tag, attribute and content</span>
  </body>
</html>
```

HTML Tags

HTML tags are like keywords which define that how web browser will format and display the content. With the help of tags, a web browser can distinguish between an HTML content and a simple content. HTML tags contain three main parts: opening tag, content and closing tag. But some HTML tags are unclosed tags.

When a web browser reads an HTML document, browser reads it from top to bottom and left to right. HTML tags are used to create HTML documents and render their properties. Each HTML tags have different properties.

Unclosed HTML Tags

Some HTML tags are not closed, for example br and hr.

**
 Tag**: br stands for break line, it breaks the line of the code.

<hr> Tag: hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

Paragraph Tag

The and are the HTML tags and "Paragraph Tag" is the HTML element, i.e. the on-page text.

This tag formats any text between the opening tag and the closing tag as a standard paragraph or main body text.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p style="font-size: 20px;">
        Lorem ipsum dolor sit amet consectetur adipisicing elit. Facere ipsa
officia iste, illo voluptatibus vitae consectetur eum fugiat amet temporibus
deserunt asperiores nihil id sequi dignissimos blanditiis sapiente ipsam
repellat!
    </p>

</body>
</html>
```

<h2> Heading Tag </h2>

In this example, <h2> and </h2> are the HTML tags and "Heading Tag" is the HTML element, i.e. the on-page heading.

Using this tag will format any text between the opening <h2> tag and the closing </h2> tag as a Heading 2 (a type of subheading.)

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <h1>This is HTML</h1>
```

```
    <h2>This is HTML</h2>
    <h3>This is HTML</h3>
    <h4>This is HTML</h4>
    <h5>This is HTML</h5>
    <h6>This is HTML</h6>
</body>
</html>
```

Bold Tag

Here the and are the HTML tags and "Bold Tag" is the HTML element, i.e. the on page text.

This tag will format any text between the opening tag and the closing tag as bold.

Italic Tag

Here, the and are the HTML tags and "Italic Tag" is the HTML element (the on-page text.)

This tag will format any text between the opening tag and the closing tag as italic.

<u> Underline Tag </u>

Here the <u> and </u> are the HTML tags and "Underline Tag" is the HTML element, i.e. the on page text.

This tag will format any text between the opening <u> tag and the closing </u> tag as underlined.

Example of bold, Italic and Underline

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p>
      <a href="https://www.google.com/"> <b style="font-size: 40px;"><i><u>
Lorem ipsum</u></i></b></a>
       dolor sit amet consectetur adipisicing elit. Sapiente sint, vitae nemo
quo officiis cum? Nihil sit exercitationem eos asperiores eaque ratione,
excepturi perferendis quam, labore dolores adipisci saepe. Amet?

    </p>
</body></html>
```

<sup> tag :

The <sup> tag is used to display text as a superscript.

Superscript is rendered with a raised baseline and smaller text.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    9<sup>
        2
    </sup>
</body>
</html>
```

<sub> tag:

The <sub> tag is used to display text as a subscript.

Subscript is rendered as small text with a lower baseline.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    H<sub>
        2
    </sub>O
</body>
</html>
```

<pre> tag:

The <pre> is also called a preformatted tag. It is used to present text as it is written in an HTML document.

It renders text using a monospace font. The whitespaces used in the element are displayed the same as written.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
   <pre>

    public class Main {

        public static void main(String[] args) {

          int num = 29;
          boolean flag = false;
          for (int i = 2; i <= num / 2; ++i) {
            // condition for nonprime number
            if (num % i == 0) {
              flag = true;
              break;
            }
          }

          if (!flag)
            System.out.println(num + " is a prime number.");
          else
            System.out.println(num + " is not a prime number.");
        }
      }
   </pre>

</body>
</html>
```

<abbr> tag:

The <abbr> tag is used to represent an abbreviation or acronym. The tag accepts a title attribute that accepts the full form of the word. When the reader hovers the text it shows a full description of the word.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p>
       <abbr title="dollar indicates money"> dolar</abbr> ipsum dolor sit
amet consectetur, adipisicing elit. Eaque possimus ipsum, similique sequi eum
error in placeat velit consequuntur, magni consequatur saepe sit maxime esse,
ad tenetur tempore dolorum eligendi.
    </p>
</body>
</html>
```

<mark> tag

The <mark> tag marks or highlights a text to grab the reader's attention.

According to HTML standards, it is not mandatory to use it, you can achieve it using CSS.

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p>
      <a href="https://www.google.com/"> <b style="font-size: 40px;"><i><u>
Lorem ipsum</u></i></b></a>
       <mark style="background-color: lawngreen;"> sit amet consectetur
adipisicing elit.</mark> Sapiente sint, vitae nemo quo officiis cum? Nihil
sit exercitationem eos asperiores eaque ratione, excepturi perferendis quam,
labore dolores adipisci saepe. Amet?

    </p>
</body>
</html>
```

<del> tag:

The <del> tag defines texts which have been removed from the document or webpage. It can be used to track changes to previous versions of code.

The content of the del element has a strikethrough line that cuts the content horizontally in middle.

<ins> tag

The <ins> tag is used to display a group of text which has been added to the document.

The ins tag adds a solid underline to its text.

Example del and ins tag:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=<device-width>, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <del>
        <p>
            A prime number is a number that is divisible by only two numbers:
        </p>
    </del>
    <ins>
        A prime number is a number that is divisible by only two numbers:
    </ins>
</body>
</html>
```

Div Tag:

The HTML division tag, called "div" for short, is a special element that lets you group similar sets of content together on a web page. You can use it as a generic container for associating similar content**.**

The tag is an empty container that is used to define a division or a section. It does not affect the content or layout and is used to group HTML elements to be styled with CSS or manipulated with scripts.

Example:

```
<div class="div1">
   <h2>This is heading.</h2>
   <p>This is a paragraph</p>
</div>

<div class="div2">
   <h2>This is another heading.</h2>
   <p>This is another paragraph</p>
</div>

<style>
   .div1 {
      color: red;
   }

   .div2 {
      color: green;
   }
</style>
```

Chapter 2 HTML Ordered and Unordered List

What is an HTML List?

List in HTML helps to display a list of information semantically. There are three types of lists in HTML:

- Unordered list or Bulleted list (ul)
- Ordered list or Numbered list (ol)
- Description list or Definition list (dl)

HTML Ordered List or Numbered List

In the ordered HTML lists, all the list items are marked with numbers by default. It is known as numbered list also. The ordered list starts with <ol> tag and the list items start with <li> tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
    <ol>
      <li>Apple</li>
      <li>Mango</li>
      <li>Banana</li>
      <li>Grapes</li>
      <li>Orange</li>
    </ol>
  </body>
</html>
```

Instead of numbers, you can mark your list items with the alphabet: A, B, C or a,b,c, or roman numerals: i, ii, iii, etc. You can do this by using the <ol> tag type attribute.

Let's explore how to order lists with alphabets and roman numbers.

To mark the list items with letters A, B, C, etc., you must specify A as the type attribute's value in the <ol> tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
   <ol type="A">
      <li>Apple</li>
      <li>Mango</li>
     <li>Banana</li>
  </ol>
</body>
</html>
```

Here is an example to show the use of Lower case letters to list the items.

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
   <ol type="a">
      <li>Apple</li>
      <li>Mango</li>
     <li>Banana</li>
  </ol>
</body>
</html>
```

Here is an example to show the use of Roman numerals to list the items.

```
<!DOCTYPE html>
<html>
  <head>
```

```
    <title>HTML Ordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
   <ol type="i">
      <li>Apple</li>
      <li>Mango</li>
     <li>Banana</li>
   </ol>
</body>
</html>
```

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>OL Tag Demo</title>
    <style type ="text/css">
        ol
            {

                font-size: 20px;
                font-family: 'Courier New', Courier, monospace;
                margin-left: 20px;

            }
            ol li{
                list-style-image: url(imgs/1.png);
                list-style-position: inside;
                border: 1px solid black;
            }
    </style>
</head>
<body>

    <ol>
        <li><a href="https://google.com">Open Google</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
    </ol>

</body>
</html>
```

HTML Unordered List or Bulleted List

In HTML unordered list, the list items have no specific order or sequence. An unordered list is also called a Bulleted list, as the items are marked with bullets. It begins with the <ul> tag and and closes with a tag. The list items begin with the <li> tag and end with tag.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Unordered List</title>
  </head>
  <body>
  <h2>List of Fruits</h2>
    <ul>
      <li>Apple</li>
      <li>Mango</li>
      <li>Banana</li>
      <li>Grapes</li>
      <li>Orange</li>
    </ul>
  </body>
</html>
```

Example: Use image also

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style type ="text/css">
        ul
            {

                font-size: 20px;
                font-family: 'Courier New', Courier, monospace;
                margin-left: 20px;

            }
            ul li{
```

```
                list-style-image: url(imgs/1.png);
                list-style-position: inside;
                border: 1px solid black;
            }
    </style>
</head>
<body>
    <ul>
        <li><a href="https://google.com">Open Google</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
        <li><a href="https://google.com">Open Google Page</a></li>
    </ul>
</body>
</html>
```

HTML Description List or Definition List

In an HTML Description list or Definition List, the list items are listed like a dictionary or encyclopedia. Each item in the description list has a description. You can use a description list to display items like a glossary. You will need the following HTML tags to create a description list:

- <dl> (Definition list) tag – Start tag of the definition list
- <dt> (Definition Term) tag – It specifies a term (name)
- <dd> tag (Definition Description) – Specifies the term definition
- </dl> tag (Definition list) – Closing tag of the definition list

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Description List</title>
  </head>
   <body>
    <dl>
      <dt><b>Apple</b></dt>
      <dd>A red colored fruit</dd>
      <dt><b>Honda</b></dt>
      <dd>A brand of a car</dd>
      <dt><b>Spinach</b></dt>
      <dd>A green leafy vegetable</dd>
```

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

Example:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style type ="text/css">
        dl{

            background-color: lightblue;
            font-family: 'Courier New', Courier, monospace;
            font-size: 20px;
        }
        dd{
            color: brown;
            font-weight: bold;
            font-style: italic;
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <dl>
        <dt><b>Apple</b></dt>
        <dd>A red colored fruit</dd>

        <dt><b>Honda</b></dt>
        <dd>A brand of a car</dd>

        <dt><b>Spinach</b></dt>
        <dd>A green leafy vegetable</dd>
     </dl>

</body>
</html>
```

HTML Nested Lists

An HTML Nested list refers to a list within another list. We can create a nested ordered list, a nested unordered list, or a nested ordered list inside an unordered list.

Example of an HTML Nested Ordered List

```
<!DOCTYPE html>
<html>
<head>
   <title>HTML Nested Ordered List</title>
</head>
<body>
   <ol>
      <li>Banana</li>
      <li> Apple
         <ol>
            <li>Green Apple</li>
            <li>Red Apple</li>
         </ol>
      </li>
      <li>Pineapple</li>
    <li>Orange</li>
   </ol>
</body>
</html>
```

Example of an HTML Nested Unordered List

```
<!DOCTYPE html>
<html>
<head>
   <title>HTML Nested Unordered List</title>
</head>
<body>
   <ul>
      <li>Fruits</li>
      <ul>
         <li>Apple</li>
```

```
            <li>Banana</li>
            <li>Mango</li>
            <li>Orange</li>
        </ul>
        <li>Vegetables</li>
        <ul>
            <li>Spinach</li>
            <li>Cauliflower</li>
            <li>Beetroot</li>
        </ul>
        <li>Cereals</li>
        <li>Nuts</li>
    </ul>
</body>
</html>
```

Chapter 3 HTML Links

What are HTML Links?

HTML links or hyperlinks are words, phrases, or images in online content that, on clicking, take the site visitor to another web page with related content. Links are an integral part of the World Wide Web.

Links connect pages both within a website and to other websites. They enable site visitor's to collect relevant information in less time by clicking their way from one page to another on any server across the world.

There are two ends of a link: anchor and direction. The link will start at the 'source' anchor and point to the 'destination' anchor. The destination anchor may be an HTML document, image, video, or a section within an HTML document.

HTML Link Syntax

In HTML, we specify a link using the <a> tag.

Example:

```
<a href="url">Link text</a>
```

Explanation:

- **<a>:** This is the anchor tag that creates a hyperlink. Anything between this tag becomes part of the link. The user can click that part to reach the linked document.
- **href:** Specifies the destination address of the link used.
- **Text link:** The visible part of the link.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Example of HTML Link</title>
```

```
  </head>
  <body>
    <p>Click on the below link</p>
    <a href = "https://www.naukri.com/learning">Shiksha Online</a>
  </body>
</html>
```

target Attribute:

The target attribute helps us define the location where the linked document will open. Some of the target locations include:

Target	Description
_self	The link opens in the same frame. It is the default value, so you need not specify this value.
_blank	The link opens in a new window.
_parent	The linked document opens in the parent frameset.
_top	It will open the link in the full body of the window.
framename	It opens the linked document in the named frame.

Syntax:

```
<a href="URL"_blank|_self|_parent|_top|framename">Link Text</a>
```

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Target Attribute Example</title>
  </head>
  <body>
    <p>Click any of the following links</p>
```

```
    <a href = "https://www.google.com" target = "_self">Link will open in Self</a> |
    <a href = "https://www.google.com" target = "_blank">Link will open in New</a> |
    <a href = "https://www.google.com " target = "_parent"> Link will open in Parent</a> |
     <a href "https://www.google.com target = "_top"> Link will open in Body of the window</a>
   </body>
</html>
```

HTML Link Colors:

By default, in most browsers, links appear in the following colors:

- Unvisited link: blue and underlined.
- Visited link: purple and underlined.
- Hover link: when the user hovers mouse over it.
- Active link (the moment a link is clicked): red and underlined.

We can change the color of the links using the CSS property.

Syntax to change Link text color with CSS styling:

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
a:link {
   color: green;
   background-color: transparent;
}
a:visited {
   color: pink;
   background-color: transparent;
}
a:hover {
   color: red;
   background-color: transparent;
}
```

```
a:active {
   color: yellow;
   background-color: transparent;
}
</style>
</head>
<body>
<p>Changing HTML Link Colors</p>
<a    href="https://www.shiksha.com/online-courses/"    target="_blank">Shiksha
Online</a>
</body>
</html>
```

Chapter 4 HTML Table Tag

Table of Contents

What is an HTML Table?

A table in HTML refers to the arrangement of data in rows and columns. Using tables, we can display data, such as products and their prices; employees, their date of joining and salaries; or flights, their ticket prices, and departure times.

Table Tag in HTML

Table tag in HTML, is used to display data in tabular format (i.e. in row-column format). These table tags manage the layout of the page (i.e. Header, Body Content etc.).
To create a table in HTML, you will need to define the table with the <table> tag. The <table> tag is the container of the table that specifies where the table will begin and where it ends.
You can add a table row with the <tr> tag. To add a table header, we will need to use the <th> tag. A table cell or data can be added using the <td> tag.

Common HTML Table Tags:

HTML Table Tag	Description
<table>	defines a table
<tr>	represents rows
<td>	to create data cells
<th>	to add table headings
<caption>	used to insert captions

HTML Table Syntax

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Example</title>
</head>
<body>
<table>
  <tr>
    <td>Column 1</td>
    <td>Column 2</td>
```

```
  </tr>

  <tr>

    <td>Column 1</td>

    <td>Column 2</td>

</tr>

</table>

</body>

</html>
```

Adding Border to an HTML Table

By default, HTM tables do not have a border. We can set the border using the CSS border property. Let's look at an example to add a 1-pixel border to the HTML table.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <style>
   table,
   th,
   td {
      border: 1px solid black;
   }
   </style>
</head>
<body>
   <table style="width:100%">
      <tr>
         <th>Name</th>
         <th>Age</th>
         <th>Salary</th>
      </tr>
      <tr>
```

```
      <td>John</td>
      <td>24</td>
      <td>20,000</td>
    </tr>
    <tr>
      <td>Adam</td>
      <td>31</td>
      <td>35,000</td>
    </tr>
    <tr>
      <td>Chris</td>
      <td>27</td>
      <td>32,000</td>
    </tr>
  </table>
</body>
</html>
```

Adding Collapsed Borders to an HTML Table

The borders around the cells and the table are separated from each other, by default. We can use the CSS border-collapse property to add a collapsed border in HTML.

Also, by default, the text inside the <th> elements is aligned horizontally center and displayed in bold font. We can change the default alignment using the CSS text-align property.

Example

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

Let's check out the below example to collapse the table borders and align the table header text to left.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <style>
   table,
   th,
   td {
      border: 1px solid black;
      border-collapse: collapse;
   }
th {
   text-align: left;
}
   </style>
</head>
<body>
   <table style="width:100%">
      <tr>
         <th>Name</th>
         <th>Age</th>
         <th>Salary</th>
      </tr>
      <tr>
         <td>John</td>
         <td>24</td>
         <td>20,000</td>
      </tr>
      <tr>
         <td>Adam</td>
         <td>31</td>
         <td>35,000</td>
      </tr>
      <tr>
         <td>Chris</td>
```

```
         <td>27</td>
         <td>32,000</td>
      </tr>
   </table>
</body>
</html>
```

Adding Padding to a Table:

To add more space to the table cells, we can use the CSS padding property.
Here is the syntax to add a 1-pixel border to the table and 20-pixels of padding.

```
<!DOCTYPE html>
<html>
<head>
   <style>
   table,
   th,
   td {
   border: 1px solid black;
   border-collapse: collapse;
   }

   th,
   td {
      padding: 20px;
   }
   </style>
</head>
<body>
   <table style="width:100%">
      <tr>
         <th>Name</th>
         <th>Age</th>
         <th>Salary</th>
      </tr>
      <tr>
         <td>John</td>
         <td>24</td>
         <td>20,000</td>
```

```
      </tr>
      <tr>
         <td>Adam</td>
         <td>31</td>
         <td>35,000</td>
      </tr>
      <tr>
         <td>Chris</td>
         <td>27</td>
         <td>32,000</td>
      </tr>
   </table>
</body>
</html>
```

Spanning Multiple Rows and Columns

With the rowspan and colspan attributes, we can make table rows and columns extend across multiple other rows and columns. In simple words, we can merge two or more rows into a single row; or two or more columns into a single column using the rowspan and colspan attributes respectively.

```
<!DOCTYPE html>
<html>
<head>
      <style>
      table {
         border-collapse: collapse;
      }
      table,
      th,
      td {
         border: 1px solid black;
      }
      th,
      td {
         padding: 10px;
      }
   </style>
</head>
```

```
<body>
   <table>
   <tr>
      <th>Name</th>
      <td>James</td>
   </tr>
   <tr>
      <th rowspan="2">Subjects</th>
      <td>English</td>
   </tr>
   <tr>
      <td>Science</td>
   </tr>
</table>
</body>
</html
```

Colspan Example:

```
<!DOCTYPE html>
<html>
<head>
      <style>
      table {
         border-collapse: collapse;
      }
      table,
      th,
      td {
         border: 1px solid black;
      }
      th,
      td {
         padding: 10px;
      }
   </style>
</head>
<body>
   <table>
   <tr>
      <th>Name</th>
```

```
      <td>James</td>
   </tr>
   <tr>
      <th rowspan="2">Subjects</th>
      <td>English</td>
   </tr>
   <tr>
      <td>Science</td>
   </tr>
</table>
</body>
</html>
```

Adding Captions to Tables

We can use the <caption> element to add a caption or title to our table. The <caption> element should be present after the opening <table> tag.

By default, the caption is displayed at the top of the table. We can change its placement using the CSS caption-side property.

```
<!DOCTYPE html>
<html>
<head>
   <style>
   table,
   th,
   td {
      border: 1px solid black;
      border-collapse: collapse;
   }
   th,
   td {
      padding: 10px;
   }
   </style>
</head>
<body>
   <table style="width:100%">
      <caption>Employee Data</caption>
      <tr>
```

```
            <th>Name</th>
            <th>Age</th>
            <th>Salary</th>
        </tr>
        <tr>
            <td>John</td>
            <td>24</td>
            <td>20,000</td>
        </tr>
        <tr>
            <td>Adam</td>
            <td>31</td>
            <td>35,000</td>
        </tr>
        <tr>
            <td>Chris</td>
            <td>27</td>
            <td>32,000</td>
        </tr>    </table>
</body>
</html>
```

Adding a Background Color to a Table:

We can add a background to an HTML table using the background-color option.
Example to show how to add color to an HTML table:

```
<!DOCTYPE html>
<html>
<head>
    <style>
    table,
    th,
    td {
        border: 1px solid black;
        border-collapse: collapse;
    }
    th,
    td {
        padding: 10px;
```

```
   }
table {
      width: 100%;
      background-color: #00ffbb;
   }
   </style>
</head>
<body>
   <table style="width:100%">
      <caption>Employee Data</caption>
      <tr>
         <th>Name</th>
         <th>Age</th>
         <th>Salary</th>
      </tr>
      <tr>
         <td>John</td>
         <td>24</td>
         <td>20,000</td>
      </tr>
      <tr>
         <td>Adam</td>
         <td>31</td>
         <td>35,000</td>
      </tr>
      <tr>
         <td>Chris</td>
         <td>27</td>
         <td>32,000</td>
      </tr>    </table>
</body>
</html>
```

Table design using css:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        table{
            border: 1px solid black;
            padding: 4px;
            width: 500px;
            border-collapse: collapse;
            margin: auto;
            margin-top: 100px;
            box-shadow: 10px 10px 5px gray;
        }
        th{
            background-color: brown;
            color: white;
            padding: 5px;
        }
        td{
                text-align: center;
        }
        tr:hover{
            background-color: green;
```

```
        }
    </style>
</head>
<body>
    <table border="1">
        <tr>
            <th>Sr.No</th>
            <th>Name</th>
            <th>Salary</th>
        </tr>
        <tr>
            <td>1</td>
            <td>XYZ</td>
            <td>5000</td>
        </tr>
        <tr>
            <td>2</td>
            <td>PQR</td>
            <td>50000</td>
        </tr>
        <tr>
            <td>3</td>
            <td>ABC</td>
            <td>500</td>
        </tr>
    </table>
</body>
</html>
```

Chapter 5 HTML Image Map

We can use HTML image maps to make clickable areas on images. An image map comprises a picture with clickable areas, where you can click on the image and it will open to a new or already specified location.

The <map> tag can include many <area> elements that define the coordinates and type of the area. You can simply link any area of a picture to other pages using the <map> tag without dividing the image.

The HTML <map> tag defines an image map. An image map is just an image with some clickable areas. The clickable areas are marked with one or more <area> tags depending on the requirement.

The idea behind an image map is that you should be able to do different activities based on where you click in the image. To make an image map, you'll need an image and some HTML code that specifies the clickable locations.

The Image

The image is inserted in Html document using the <img> tag. The only difference from other images is that it contains a usemap attribute inside the <img> element.

Example:

<img src="image1.jpg" alt="Desk" usemap="#deskmap" width="500" height="380">

The usemap attribute value starts with a hashtag # followed by the name of the image map. The usemap attribute is used to create a connection between the image and the image map. We can use any image as an image map by using the usemap attribute inside the image element.

Create Image Map

Image map is created using a <map> element. The <map> element is used to generate an image map and is linked to the image by using the name attribute

Example:

<map name="deskmap">

The name attribute must have the identical value as the images usemap attribute then only it will work.

The Areas

After creating the map element we need to create clickable areas. A clickable area is marked using an <area> element.

<area shape="rect" coords="175,242,420,358" alt="Keyboard" target="_blank" href="about.html">

The area can be a rectangle, circle, polygon, or a whole region; we can select any of them depending on our requirements. Within the area element, we must define the shape we are using as well as the coordinates of the area we want to make clickable.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>Image Maps</h2>
<img src="https://cdn.pixabay.com/photo/2017/05/11/11/15/workplace-2303851_960_720.jpg" alt="Desk" usemap="#deskmap" width="500" height="380">
<map name="deskmap">
<area shape="rect" coords="175,242,420,358" alt="Keyboard" target="_blank" href="https://en.wikipedia.org/wiki/Computer_keyboard">
<area shape="rect" coords="444,251,481,357" alt="Mouse" target="_blank" href="https://en.wikipedia.org/wiki/Computer_mouse">
<area shape="rect" coords="375,14,481,357" alt="Diary" target="_blank" href="https://en.wikipedia.org/wiki/Book">
</map>
</body>
</html>
```

Chapter 6 HTML5 Audio and Video Tag

Audio files are used to store audio data on a variety of devices, including computers, MP3 players, and cell phones. You must convert audio data into a digital format before storing it.

Encoding the raw audio data is the process of transforming audio data into a digital file. It includes extracting audio data samples and storing them in a compressed way to reduce file size.

Syntax: The basic syntax of the <audio> element

<audio> … </audio>

Embedding Audio in HTML Document

Previously, embedding audio into a web page was difficult due to the lack of a universal standard for specifying embedded media assets like audio in web browsers. Below we'll show you how to integrate audio in your website using the latest HTML5 <audio> element.

Using the HTML5 audio Element

The HTML5 <audio> element, which was recently introduced, provides a standard way to embed audio in web pages. The audio element, on the other hand, is very new, but still, it works in most modern web browsers. The example below simply adds audio into an HTML5 document using the browser's default controls and a single source defined by the src attribute.

```
<!DOCTYPE html>
<html>
<body>
<audio controls>
<source src="birds.mp3" type="audio/mpeg">
</audio>
</body>
</html>
```

HTML <audio> Autoplay:

We can add autoplay inside the audio element to let our audio file start playing automatically.

Example:

```
<!DOCTYPE html>
<html>
<body>
<audio controls autoplay>
<source src="birds.mp3" type="audio/mpeg">
</audio>
</body>
</html>
```

Audio controls, such as play, pause, and volume, are added by the controls attribute. we can specify alternative audio files for the browser to choose from using the <source> element. The first recognized format will be used by the browser. Only browsers that do not support the <audio> element will display the text between the <audio> and </audio> tags.

Tag-Specific Attributes

The following table shows the attributes that are specific to the <audio> tag.

Attribute	Value	Description
autoplay	autoplay	This Boolean attribute indicates that the audio will begin playing automatically.
controls	controls	The browsers will display controls for controlling audio playback, such as play/pause, volume, and so on.
loop	loop	When the audio reaches the end, this Boolean attribute specifies that it will automatically start over.
muted	muted	The audio will be muted if this Boolean property is true.
src	URL	The audio file's location is specified using src

File Format	Media Type
MP3	audio/mpeg
OGG	audio/ogg
WAV	audio/wav

Videos in HTML with Examples

A video file is a collection of images displayed in a sequence to represent moving scenes. Different video codecs, such as DivX and QuickTime, are often used to encode and decode video files. A video is displayed on a web page using the HTML <video> element.

Syntax: The basic syntax of the <video> element

<video> … </video>

Web browsers commonly support three different formats: mp4, Ogg, and WebM.

Video Controls in HTML

The browser does not display any controls for the video element by default; only the video is displayed.

<video src="file.mp4" >

This means that the audio will only play if the video is set on autoplay, and the user is unable to stop, pause, control the volume, or jump to a specific point in the video. Because the control attribute is not defined inside the video element. We can use the controls attribute inside the video element to display the built-in controls.

<video src="file.mp4" controls >

Example:

```
<!DOCTYPE html>
<html>
<style>
body{
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
background-color:mediumseagreen;
}
video{
border-radius: 8px;
border: 5px solid transparent;
}
</style>
<body>
<video width="400" controls>
<source src="/parrot.mp4" type="video/mp4">
</video>
</body>
</html>
```

It is always a good practice to include width and height attributes. If the height and width are not specified, the page may flicker while the video is loading.

Video controls, such as play, pause, and volume, are added by the controls attribute. Only browsers that don't support the <video> element will display the text between the <video> and </video> tags.

Tag-Specific Attributes

Attribute	Value	Description
load	load	Reloads the current playing video.
play	play	Start playing the video.
pause	pause	Pauses playing the current video.
autoplay	autoplay	This Boolean attribute indicates that the video will begin playing automatically.
src	URL	The video file's location is specified using src.
height	height	It is used to specify the height of the video player.
width	width	It is used to specify the width of the video player.
muted	muted	The audio will be muted if this Boolean property is true.

Chapter 7 HTML Form Design

How to Create HTML Forms:

An HTML or a Web form helps collect user input. HTML forms can have different fields, such as text areas, text boxes, radio buttons, checkboxes, drop-down lists, file uploads, etc.

The collected data can be validated on the client browser using JavaScript. After validation of form data on the client-side, the user clicks on the Submit button in the form. After this, data is sent to the server for further processing.

Attributes for the <form> tag:

accept - This is only used when you have an <input> tag with a "file" attribute. It restricts the type of files uploaded to the server when the form is submitted. The values for this attribute should be one or more MIME types. For example: class="example"accept="text/html, image/gif" would only allow an HTML file or a GIF file to be uploaded.

action - This tells the form where the CGI application is located. If you are using cgiwrap, then this attribute value should be action="http://www.engr.colostate.edu/usr-bin/cgiwrap/your username/your script name"

enctype - This sets the MIME type for the data being submitted. This is only relevant when the method attribute is "POST". The default is "application/x-www-form-urlencoded" and you will rarely need to change it. You will, however, want to set this to "multipart/form-data" if you are uploading files via your form.

method - Form are submitted with two possible methods: GET and POST. The default is GET. There are some CGI programs that require that the form be submitted via only the POST or only the GET method. So what's the difference?

GET - The data is submitted and encoded into the URL of the resulting web page. The URL takes the value of the "action" attribute, appends a "?" to it and then appends the form data set, such as name=arthur&color=red&quest=grail You are limited to a total of 256 bytes with this method and the result of your submission is visible in the browser's URL location bar.

POST - This method submits the data "behind the scenes" via standard input. There is no limit to the amount of bytes submitted as there is with the GET method and the data is not displayed in the browser's URL location bar.

name & id - These assign an identifier name to the form for reference within a CGI application. Newer browsers support the preferred "id" attribute, but the "name" attribute is still needed with a few older CGI programs.

target - This specifies which browser window the resultant information will be displayed. You can use a value of "_blank" to open a new browser window.

Example: <form id="emailform" action="http://www.engr.colostate.edu/usr-bin/cgiwrap/joesmith/email.pl" method="post">

This tag creates a form with the id of "emailform" that will submit to the program "email.php" with a "post" method.

Process to Create Forms Using HTML

HTML <form> Element

To create an HTML form, we will use the HTML <form> element. It starts with the <form> tag and ends with the </form> tag. We can add the input elements within the form tags for taking user input.

Syntax:

```
<form>

    form elements, such as text box, textarea, etc.

</form>
```

Here are the different input types you can use in HTML:

- <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 <input> Element

We use the HTML <input> element to create form fields and receive input from the user. We can use various input fields to take different information from the user. Using different **Type attributes**, we can display an **<input> element** in various ways.

Example:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Text Input</title>

  </head>

<body>

 <form>

   Enter your first name  <br>

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

 </form>

</body></html>
```

Common HTML Form Tags

The following are some of the commonly used HTML tags to create web forms:

Tag	Description
<form>	This tag defines an HTML form to receive input from the user.
<input>	Defines an input control.
<textarea>	Defines a multi-line input control.
<label>	Creates a caption for input elements.
<select> & <option>	Create a drop-down menu with various options. The <select> tag defines the selection or the drop-down, while the <option> tag represents all the options in the list.
<optgroup>	Creates a drop-down list of related options.
<fieldset>	Groups the related element in a web form.
<legend>	It defines a caption for the <fieldset> element.
<button>	Creates a clickable button.
<output>	It is used for defining the result of a calculation.

Password in an HTML Form

We use the type value as 'password' to take the password as input. The password entered by the user will not be visible in the password field control.

Example:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Password Input Control</title>

  </head>

<body>

<form>

<p>

      <label>Username : <input type="text" /></label>

</p>

<p>

      <label>Password : <input type="password" /></label>

</p>

</form>

</body>

</html>
```

Radio Button Control

Radio buttons let users select one option out of many options. We can create a Radio Button control using the HTML <input> tag and the type attribute will be radio. <input type="radio"> defines a radio button.

Example:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Radio Button Control</title>

  </head>

<body>

    <form>

      <input type = "radio" name = "Gender" value = "Male"> Male

      <input type = "radio" name = "Gender" value = "Female"> Female

    </form>

  </body>

</html>
```

Submit Button Control

We can create clickable buttons in HTML using the <input>tag and setting its type attribute to Submit. Using the type="submit", we can add a submit button on the web page.

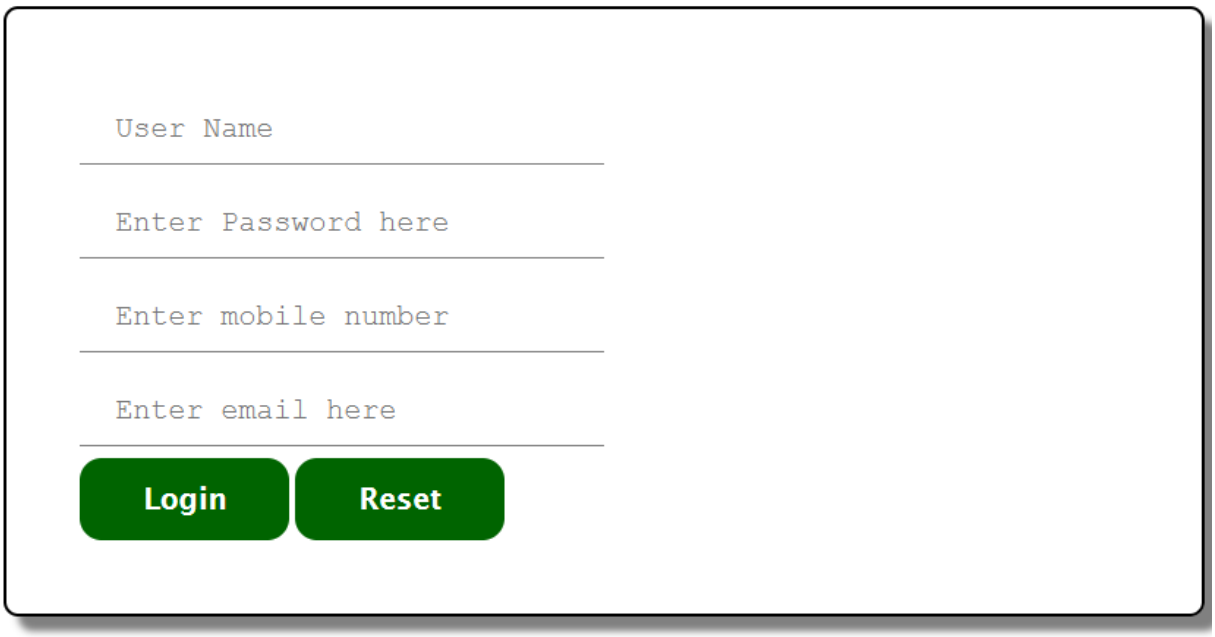
Users can submit the details provided in the form by clicking on the Submit button in an HTML form. These details are submitted to the form handler – a file on the server with a script for processing input data.

Example:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Submit Button</title>
  </head>
<body>
  <form>
<p>
      <label>Username : <input type="text" /></label>
</p>
<p>
      <label>Password : <input type="password" /></label>
</p>
<p>
    <input type = "submit" name = "submit" value = "Submit" />
   </p>
 </form>
</body>
</html>
```

Form Design:

Create the form as bellow



Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>
   <style>
      *
      {
         margin: 0px;
         padding: 0px;
         box-sizing: border-box;
      }
      #wrapper
      {
         width: 60%;
         margin: auto;
         border: 2px solid black;
         border-radius: 10px;
         margin-top: 40px;
         padding: 50px;
         box-shadow: 10px 10px 5px gray;
```

```
        }
        input[type=text], input[type=password],input[type=email],
        input[type=number]
        {
            width: 50%;
            padding: 12px 24px;
            margin: 8px 0;
            font-family: 'Courier New', Courier, monospace;
            font-size: 20px;
            /*border: 1px solid darkred;
            border-radius: 10px;*/
            border: none;
            border-bottom: 1px solid gray;
           /* background-color: cadetblue;*/

        }
        input[type=button], input[type=Submit], input[type=Reset]
        {
            background-color: darkgreen;
            color: white;
            width: 20%;
            padding: 16px 32px;
            font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida
Sans Unicode', Geneva, Verdana, sans-serif;
            font-size: 20px;
            font-weight: bold;
            border: none;
            border-radius: 15px;
            text-decoration: none;

        }
    </style>
</head>
<body>
    <div id="wrapper">
    <form>
      <input type="text" placeholder="User Name" id="user"><br>
      <input type="password" placeholder="Enter Password here"
id="password"><br>
       <input type="number" placeholder="Enter mobile number" id="mob"><br>
       <input type="email" placeholder="Enter email here" id="email"><br>
       <input type="button" name="Login" value="Login" id="btnLogin">
       <input type="Reset" name="Reset" value="Reset" id="btnLogin">
    </form>
</div>
</body>
</html>
```

2. Create the form as bellow.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <style>
        *
        {
            margin: 0px;
            padding: 0px;
            box-sizing: border-box;
        }
        #wrapper
        {
            width: 60%;
            margin: auto;
            border: 2px solid black;
            border-radius: 10px;
            margin-top: 40px;
            padding: 50px;
            box-shadow: 10px 10px 5px gray;

        }
        input[type=text]
        {
            width: 56%;
            padding: 12px 24px;
            margin: 8px 0;
            font-family: 'Courier New', Courier, monospace;
            font-size: 20px;
            border: 1px solid darkred;
            border-radius: 10px;
```

```
        background-image: url("imgs/sr.png");
        background-position: 10px 10px;
        background-repeat: no-repeat;
        padding-left: 40px;

      }
      input[type=button], input[type=Submit], input[type=Reset]
      {
         background-color: darkgreen;
         color: white;
         width: 40%;
         padding: 16px 32px;
         font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida
Sans Unicode', Geneva, Verdana, sans-serif;
         font-size: 20px;
         font-weight: bold;
         border: none;
         border-radius: 15px;
         text-decoration: none;

      }
   </style>
</head>
<body>
   <div id="wrapper">
      <form>
        <input type="text" placeholder="Enter product name for search"
id="user"><br>
        <br>
        <input type="button" name="Login" value="Login">
      </form>
   </div>
</body>
</html>
```