HTML Structure and Core Tags

Learn to structure the web with HTML.

ABSTRACT

HTML is the foundation of web development, used to structure and present content on the web. This module introduces the basics of HTML page structure, core tags, and editors, followed by semantic and non-semantic elements, document structure, layout, text, and formatting tags. Learners will gain a strong foundation for creating well-structured and meaningful web pages.

❖ Basics of HTML

- ✓ HTML stands for **Hyper Text Markup Language**
- ✓ It is used to design the web pages.
- ✓ With the help of HTML, you can create a complete website structure.
- ✓ HTML is the combination of Hypertext and Markup language. Hypertext defines the link between the web pages and markup language defines the text document within the tag that define the structure of web pages.
- ✓ HTML consists of a series of elements
- ✓ HTML elements tell the browser how to display the content

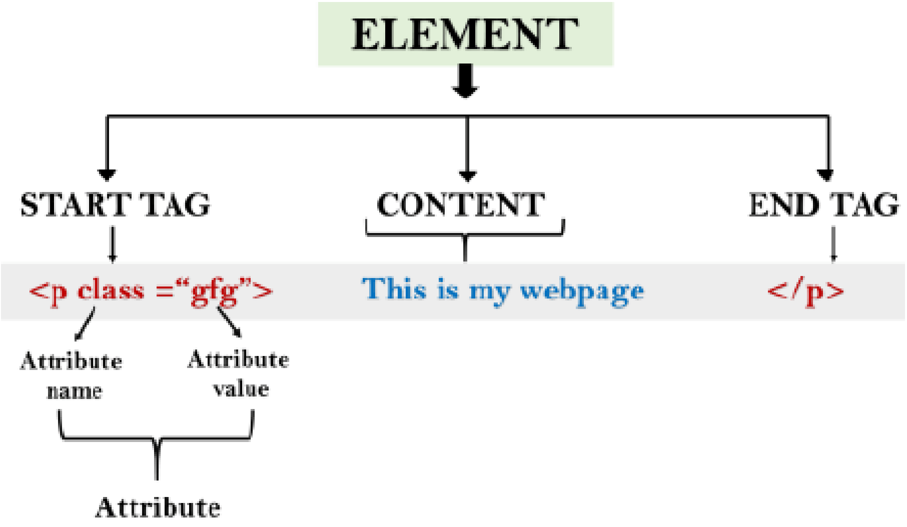
HTML Tags: Tags are the starting and ending parts of an HTML element. They begin with < symbol and end with > symbol. Whatever written inside < and > are called tags.

HTML elements: Elements enclose the contents in between the tags. They consist of some kind of structure or expression. It generally consists of a start tag, content and an end tag.

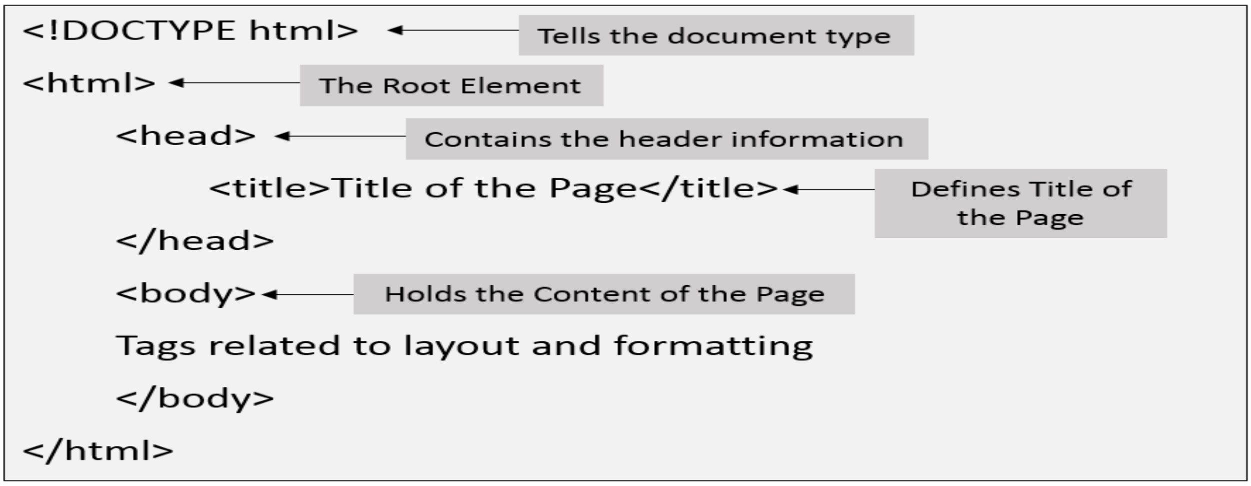
Example: <b>This is the content.</b>

Where, **<b>** is the starting tag and **</b>** is the ending tag.

HTML Attributes: It is used to define the character of an HTML element. It always placed in the opening tag of an element. It generally provides additional styling (attribute) to the element.



❖ Page Structure

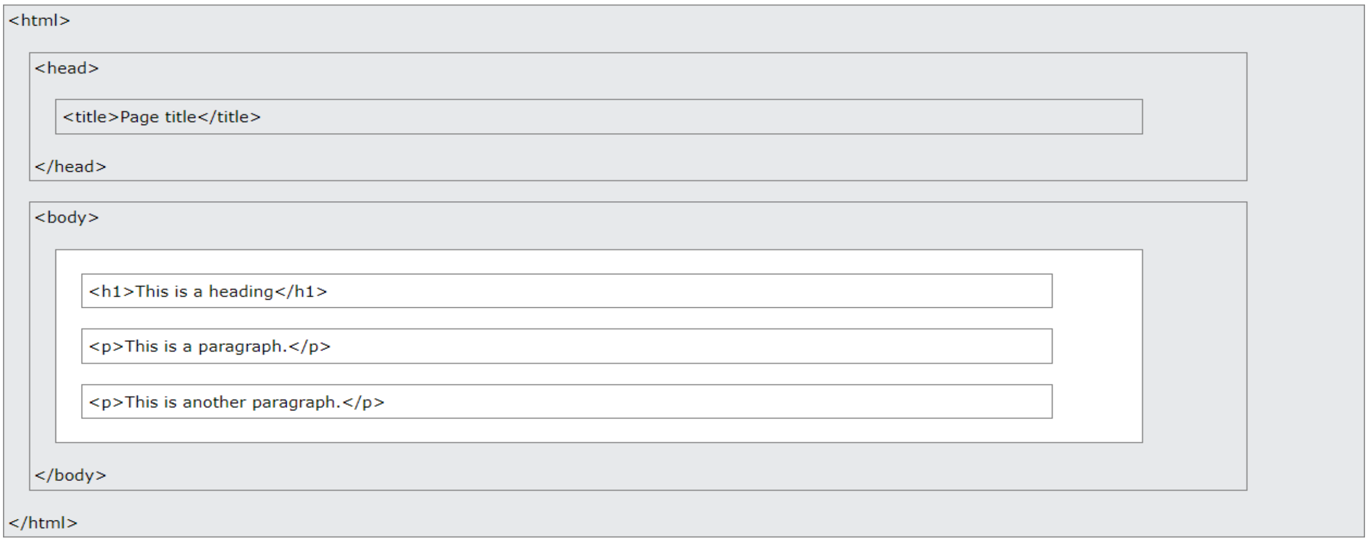


```
<!DOCTYPE html>
<html>
        <head>
                <title>Title of the Page</title>
        </head>
        <body>
        Tags related to layout and formatting
        </body>
</html>
```

Tells the document type
The Root Element
Contains the header information
Defines Title of the Page
Holds the Content of the Page

❖ Building Blocks of HTML

1. **<!DOCTYPE html>** – A doctype or document type declaration is an instruction that tells the web browser about the markup language in which the current page is written. The doctype declaration is not case-sensitive. The declaration is not an HTML tag. It is an "information" to the browser about what document type to expect.
2. **<html>** – This tag is used to define the root element of HTML document. This tag tells the browser that it is an HTML document.
3. **<head>** – This tag is used to define the head portion of the HTML document that contains information related to the document. Elements within the head tag are not visible on the front-end of a webpage.
4. **<title>** -- This element specifies a title for the HTML page (which is shown inthe browser's title bar or in the page's tab)
5. **<body>** – The body tag is used to enclose all the visible content of a webpage. In other words, the body content is what the browser will show on the front end. For example, headings, paragraphs, images, hyperlinks, tables, lists, etc.



```
<html>
  <head>
    <title>Page title</title>
  </head>
  <body>
      <h1>This is a heading</h1>
      <p>This is a paragraph.</p>
      <p>This is another paragraph.</p>
  </body>
</html>
```

❖ Editors

Web pages can be created and modified by using editors (sublime,notepad++,vs code etc). However, for learning HTML it is recommended to use simple text editor like Notepad/TextEdit (Mac). It is good way to learn HTML.

First Example:

- **Open Notepad**
- **Write HTML**

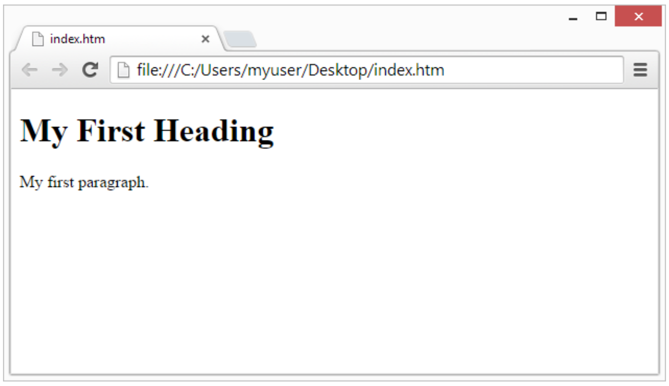
```
<!DOCTYPE html>
<html>
<head><title>Test</title></head>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

- **Save the HTML Page**

Save the file on your computer. Select **File > Save as** in the Notepad menu. Name the file **"index.html" or “index.htm”**

- **View the HTML Page in Your Browser**

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").
The result will look much like this:



My First Heading

My first paragraph.

❖ HTML Comments

- In HTML, comments are written using <!-- and -->.
- The browser ignores everything inside the comment.
- Comments are useful to explain code, organize sections, or hide code for testing

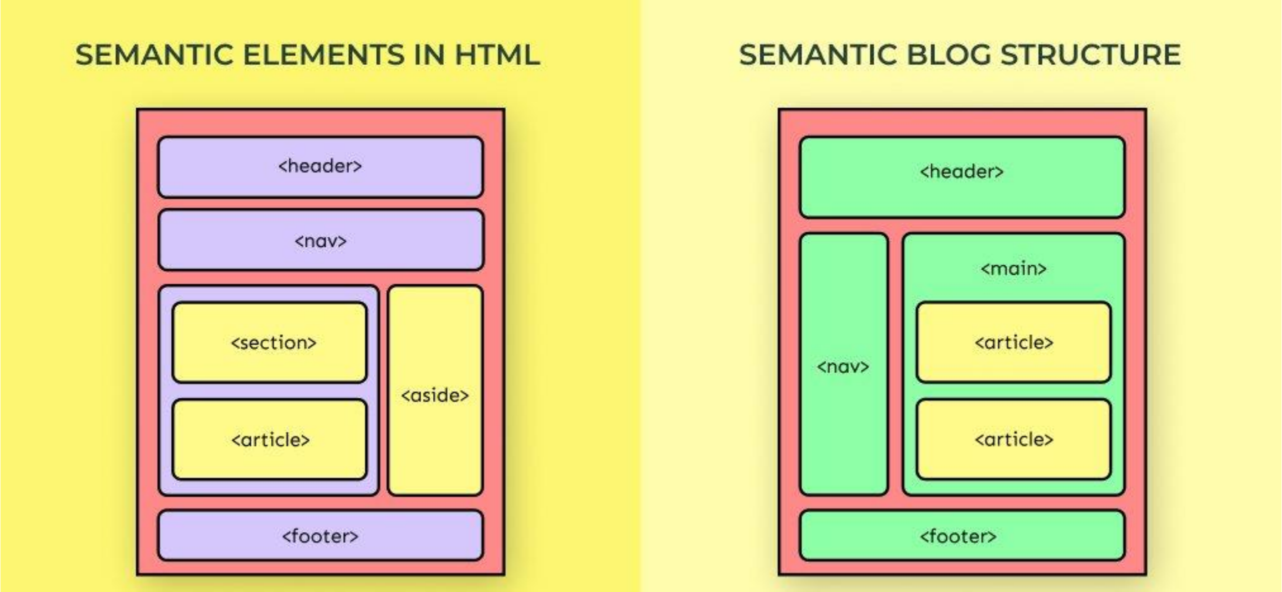
Syntax:

```
<!-- This is a comment -->
```

Example:

```
<!DOCTYPE html>
<html>
<head>
  <!-- This is the head section -->
  <title>My First Webpage</title>
</head>
<body>
  <!-- Main heading -->
<h1>Welcome</h1>
</body>
</html>
```

❖ Semantic and Non-Semantic HTML Tags



- **Semantic Tags** → clearly describe their meaning and purpose both to the browser and developer. They make code **more readable, SEO-friendly, and accessible**.
    - Examples:
        - <header> → defines header section
        - <footer> → defines footer section
        - <article> → represents an independent article
        - <section> → defines a thematic section
        - <aside> → tangential/side content (ads, related links, tips)
        - <nav> → navigation links
        - <main> → main content area
- **Non-Semantic Tags** → do not describe their meaning. They are used mainly for styling or grouping. They don't tell what the content is, only **how it should look**.
    - Examples:
        - <div> → division/container
        - <span> → inline container for text

❖ Block and Inline elements

Block Elements

Block elements always start on a new line and stretch to the full width of their parent container. They are used to structure the main layout of a page.

<div>, <p>, <h1> to <h6>, <section>, <article>, <aside>, <header>, <footer>, <table>, <ul>, <ol>, <li>, <form>, <main>, <nav>

Inline Elements

Inline elements do not start on a new line. They only take up as much width as their content requires. They are often used for formatting, styling, or embedding small chunks of content within block elements.

<a>, <span>, <b>, <strong>, <i>, <em>, <img>, <br>, <label>, <input>, <textarea>, <select>,<button>, <sub>, <sup>, <abbr>

❖ Head tags : <link>, <script>, <style>

- <link> → Connects external resources (CSS, favicon).
  - A **favicon** (short for **"favorite icon"**) is the **small icon that represents a website**.
- <script> → Adds JavaScript for interactivity.
- <style> → Adds CSS for styling.
- <meta> → It provides metadata (information about the webpage) for browsers and search engines.

```
<!DOCTYPE html>
<html>
<head>
  <title>Head Tags Example</title>
  <!-- Link tag: external CSS file and favicon -->
  <link rel="stylesheet" href="styles.css">
  <link rel="icon" type="image/png" href="favicon.png">
  <!-- Style tag: internal CSS -->
  <style>h1 { color: blue; }</style>
  <!-- Script tag: external JavaScript -->
  <script src="s1.js"></script>
</head>
<body><h1>Head Tags Demonstration</h1></body>
</html>
```

Meta Tags in HTML

The <meta> tag is placed inside the <head> section of an HTML page.

1. charset

Purpose: Defines character encoding (commonly **UTF-8** – Unicode Transformation Format-8).

Real-world scenario: Ensures that special characters, accented letters, emojis, or non-English text display correctly.

```
<meta charset="UTF-8">
```

Examples:

- **Hindi news website:** The Times of India or BBC Hindi
  - Headline: भारत में नई शिक्षा नीति लागू
- **Gujarati news website:** Divya Bhaskar or Sandesh
  - Headline: અમદાવાદમાં નવાં વર્ષ માટે તહેવારની તૈયારી

Without UTF-8:

Text may appear as **random symbols or question marks**.

Example:

- Wrong: ������ ������ ���� �����
- Correct with UTF-8: અમદાવાદમાં નવાં વર્ષ માટે તહેવારની તૈયારી

2. viewport

Purpose: Makes a website responsive on mobile devices.

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

Example: Flipkart or **Myntra**: On a mobile phone, the layout adjusts so product images, buttons, and menus fit perfectly, making online shopping easy on smartphones.

3. description

Purpose: Short summary of the page, used by search engines.

```
<meta name="description" content="Learn the best street food recipes from India including
chaat, vada pav, and golgappa.">
```

Example: When someone searches “Indian street food recipes,” Google may show this description under the page title on **Sanjeev Kapoor’s recipe website** or **Tarla Dalal recipes**.

4. keywords

Purpose: List of keywords relevant to the page content.

```
<meta name="keywords" content="Indian recipes, street food, vada pav, chaat, golgappa">
```

Example: Helps search engines understand the page topic. For instance, a blog about **Indian street food** will rank better when users search for “vada pav recipe” or “chaat recipes.”

5. author

Purpose: Indicates the creator of the webpage.

```
<meta name="author" content="Arjun Mehta">
```

Example: A food blog post written by **Arjun Mehta**, an Indian food blogger. Some browsers or CMS platforms may display the author’s name, giving proper credit.

---

6. http-equiv

Purpose: Provides HTTP header-like info. Can be used for refresh, content type, or caching.

```
<meta http-equiv="refresh" content="10;url=https://indiatimes.com">
```

Example: A page on **IndiaTimes** that says “Thank you for subscribing to our newsletter” and then redirects to the homepage after 10 seconds.

Example:

```
<head>
   <!-- Character encoding for Indian languages -->
   <meta charset="UTF-8">
   <!-- Responsive design for mobile devices -->
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <!-- Short description for search engines -->
```

```
    <meta name="description" content="Explore the best Indian street food recipes including chaat, vada pav, pani puri, and more.">
    <!-- Relevant keywords -->
    <meta name="keywords" content="Indian recipes, street food, vada pav, chaat, pani puri, Mumbai food">
    <!-- Author of the webpage -->
    <meta name="author" content="Arjun Mehta">
    <!-- Page refresh (redirects after 10 seconds) -->
    <meta http-equiv="refresh" content="10;url=https://en.wikipedia.org/wiki/Meta_element">
</head>
```

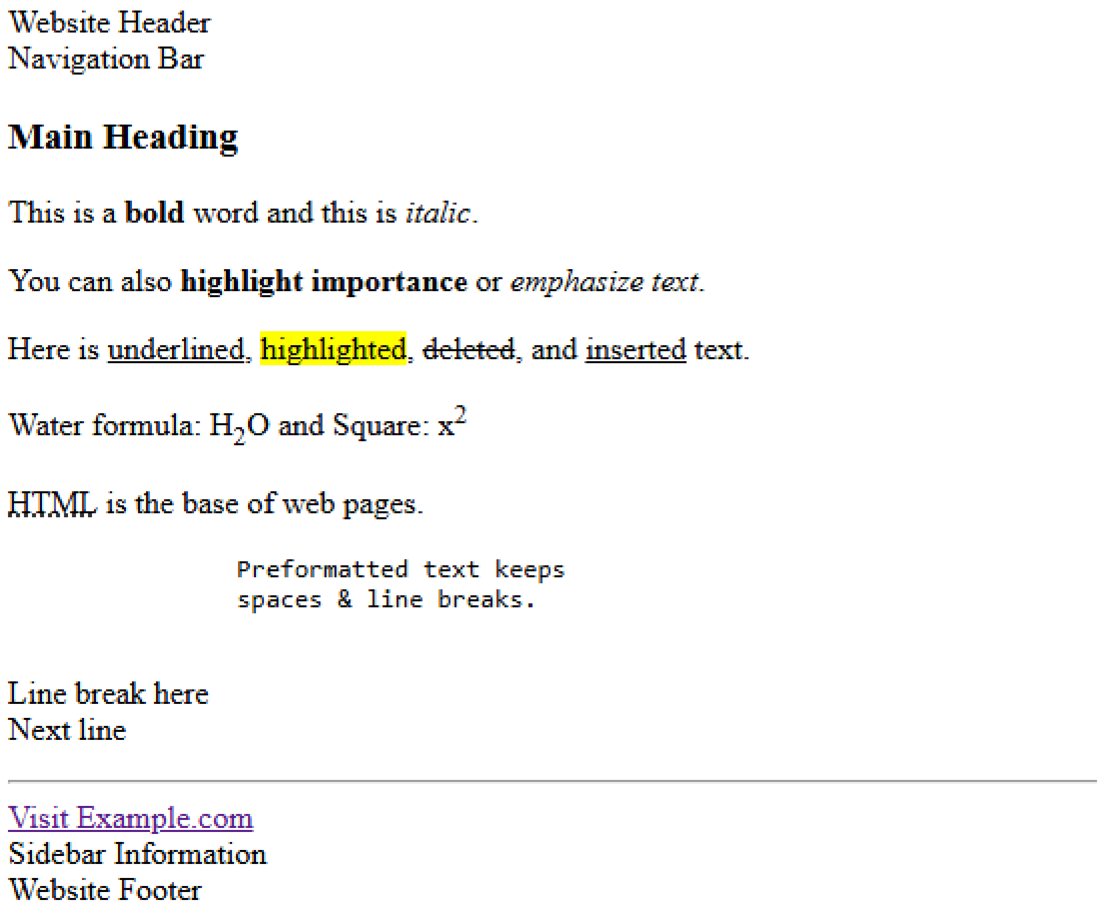
❖ TAGS

Category	Tag	Description
Layout & Sectioning Tags	<div>	Generic container, used for grouping elements.
	<span>	Inline container for styling a small portion of text.
	<header>	Represents the top section of a webpage or article.
	<footer>	Bottom section of a webpage (credits, copyright).
	<main>	Main content of the webpage.
	<section>	Defines sections in a webpage.
	<article>	Represents self-contained content (e.g., blog post).
	<aside>	Sidebar content, not the main focus.
	<nav>	Navigation links/menu.
Content & Text Tags	<h1> to <h6>	Headings (h1 is largest, h6 is smallest).
	<p>	Paragraph of text.
	<pre>	Preformatted text (preserves spaces, line breaks).
	<br>	Line break.
	<hr>	Horizontal line (divider).
Hyperlink Tag	<a>	Anchor tag used for hyperlinks.
Text Formatting Tags	<b>	Bold text (visual only).
	<strong>	Important text (semantic meaning).
	<i>	Italic text (visual only).
	<em>	Emphasized text (semantic meaning).
	<u>	Underlined text.
	<mark>	Highlighted text.
	<del>	Deleted/strikethrough text.
	<ins>	Inserted/underlined text.
	<sub>	Subscript (below text).
	<sup>	Superscript (above text).
	<abbr>	Abbreviation (shows full form on hover).

Example:

```
<html>
  <head>
  <title>HTML Tags Example</title>
 </head>
 <body>
  <!-- Layout & Sectioning Tags -->
  <header>Website Header</header>
  <nav>Navigation Bar</nav>
  <main>

   <section>
    <article>
     <h1>Main Heading</h1>
     <p>This is a <b>bold</b> word and this is <i>italic</i>.</p>
     <p>You can also <strong>highlight importance</strong> or <em>emphasize
text</em>.</p>
     <p>
      Here is <u>underlined</u>, <mark>highlighted</mark>, <del>deleted</del>,
      and <ins>inserted</ins> text.
     </p>
     <p>Water formula: H<sub>2</sub>O and Square: x<sup>2</sup></p>
     <p>
      <abbr title="HyperText Markup Language">HTML</abbr> is the base of web
      pages.
     </p>
     <pre>
        Preformatted text keeps
        spaces & line breaks.
     </pre>
     <p>Line break here<br>Next line</p>
     <hr>
     <a href="https://example.com" target="_blank">Visit Example.com</a>
    </article>
   </section>
   <aside>Sidebar Information</aside>
  </main>
  <footer>Website Footer</footer>
 </body>
</html>
```

Output:



Website Header
Navigation Bar

Main Heading

This is a **bold** word and this is *italic*.

You can also **highlight importance** or *emphasize text*.

Here is underlined, highlighted, ~~deleted~~, and inserted text.

Water formula: H_2O and Square: x^2

HTML is the base of web pages.

```
                Preformatted text keeps
                spaces & line breaks.
```

Line break here
Next line

Visit Example.com
Sidebar Information
Website Footer

❖ id and class attributes

- **id attribute**
  - A unique identifier for a single element on a page.
  - Used to **target that element** in CSS or JavaScript.
  - Each id must be unique (only one per page).
- **class attribute**
  - A reusable label that can be assigned to multiple elements.
  - Used to **apply the same styles or behaviors** to groups of elements.

Why use them?

- To **style elements** with CSS.
- To **manipulate elements** with JavaScript.
- To provide **structure and meaning** in HTML for easier maintenance and reusability.

In short:

- **id = unique identifier (one element).**

- **class = reusable category (many elements).**

❖ HTML Link: Anchor tag (<a>)

Defines a hyperlink, which is used to link from one page to another. The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

Attributes:

- ✓ **href:** Specifies the URL of the page the link goes to
- ✓ **target:** Specifies where to open the linked document
- ✓ **name:** is used to specify the name for an <a> element. It is used to reference the form data after submitting the form or to reference the element in a JavaScript.

<a href="url" target="_blank | _self" id="a_link">Test</a>

_blank = Opens the linked document in a new window or tab
_self = Opens the linked document in the same frame as it was clicked (this is default)

Example:

<a **href**=”https://www.example.com” **target**=”_blank”>Click here</a>

- ✓ When rendered in a browser, it would look like this: **Click here**
- ✓ If you click it, it will open **https://www.example.com** in a new tab.
- ✓ When target="_blank" is used, it tells the browser to open the link in a **new browser tab** or window (depending on browser settings).
- ✓ This is helpful when you want to keep the current page open while navigating to another site.

Example: Create a webpage that contains a long block of text and a link at the bottom which, when clicked, scrolls the user back to the top of the page. The solution should demonstrate how to use the id attribute to define an anchor point (<a id="...">) and the href="#..." link to navigate back to that point.

```
<html>
   <body>
      <a id="test"></a>
      <h1>
         <pre>
            a
            b
            c
```

```
        d
        e
        f
        g
        h
        i
        j
        k
        l
        m
        n
        o
        p
        q
        r
        s
        t
        u
        v
        w
        x
        y
        z
        a
        b
        c
        d
        e
        f
        g
        h
        i
        j
      </pre>
    </h1>
    <a href="#test">Scroll to top</a>
  </body>
</html>
```

1. **<a id="test"></a>**: A unique identifier used to reference or target elements. It is the preferred method in modern HTML. Using id is more current and widely supported.

2. <a href="#test">Scroll to top</a>

This creates a hyperlink that, when clicked, will scroll the user back to the point in the page where the named anchor (<a name="test"></a>) is located. The #test in the **href** attribute reference.

Note:

name: Used in older versions of HTML for named anchors, but mostly for form elements now. It's deprecated for anchor elements in HTML5.

❖ Media Tags

<img>

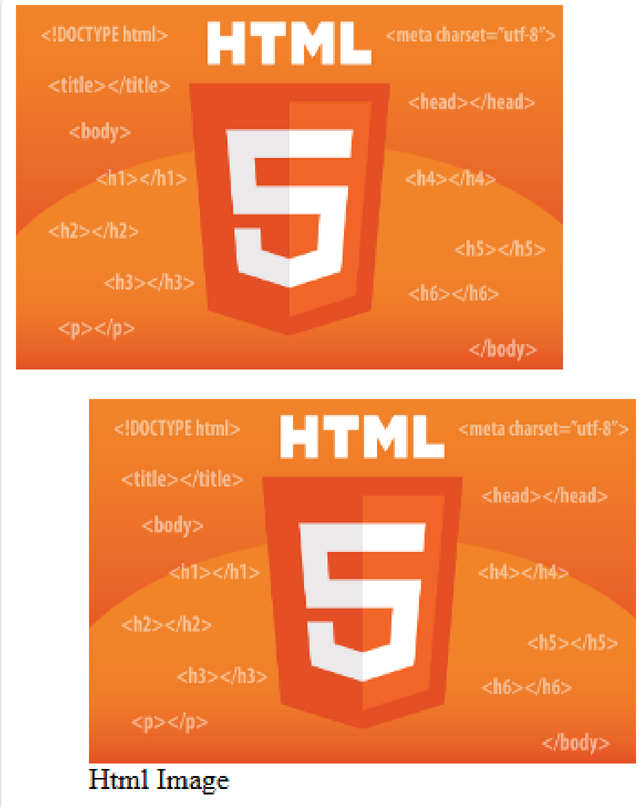
- Used to display images.
- Attributes: src (image source), alt (alternative text), width, height.

<figure> and <figcaption>

- <figure> groups media content (like images or videos) with an optional caption.
- <figcaption> provides a caption for the media inside <figure>.

Example: Understand the concept of <img>,<figure>,<figcaption>.

```
<img src="html.jpg" alt="A beautiful image" width="300" height="200">

<figure>
   <img src="html.jpg" alt="A beautiful image" width="300" height="200">
   <figcaption>Html Image</figcaption>
</figure>
```



Html Image

Using <figure> with <img> is useful because it **semantically groups an image with its caption**, making the content more meaningful and accessible. The <figcaption> inside <figure> provides a clear description or context for the image, which improves **accessibility, SEO, and content clarity**.

In short: **<figure> + <img> = image + optional caption, grouped meaningfully.**

Information:

Absolute File Paths

An absolute file path is the full URL to a file:
<img src="**https://www.lju.com/images/picture.jpg**" alt="css">

Relative File Paths

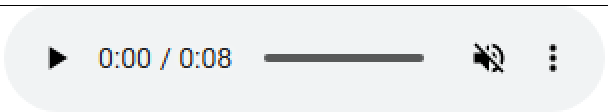
A relative file path points to a file relative to the current page.
In the following example, the file path points to a file in the images folder located at the root of the current web:
<img src="**/images/picture.jpg**" alt="css">

<audio>

- Embeds audio files.
- Attributes: controls (show play/pause controls), autoplay, loop, muted.
  - **controls** → shows play/pause/volume UI.
  - **autoplay** → tries to start audio automatically.
  - **muted** → starts muted (browsers allow muted autoplay).
  - **loop** → plays continuously.

Example:

```
<audio controls autoplay muted loop>
  <source src="1.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```



To allow play audio automatically on muted mode

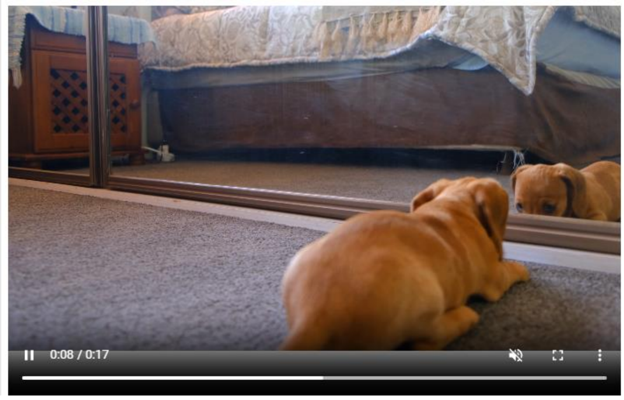
1. Open Chrome.
2. Go to **Settings** → **Privacy and Security** → **Site Settings**.
3. Scroll to **Additional content settings** → **Sound**.
4. Here you can:
   - Allow all sites to play sound automatically, OR
   - Add specific sites to **Allow** list (so they can autoplay with sound).

<video>

- Embeds video files.
- Attributes: controls, autoplay, loop, width, height, muted.

Example:

```
<video  autoplay controls muted loop width="700" height="500">
  <source src="2.mp4" type="video/mp4" >
  Your browser does not support the video element.
</video>
```



<source>

- Specifies multiple media sources for <audio> or <video> so the browser can choose a supported format.

Example:

```
<video width="400" controls>
  <source src="video.mp4" type="video/mp4">
  <source src="video.ogg" type="video/ogg">
</video>
```

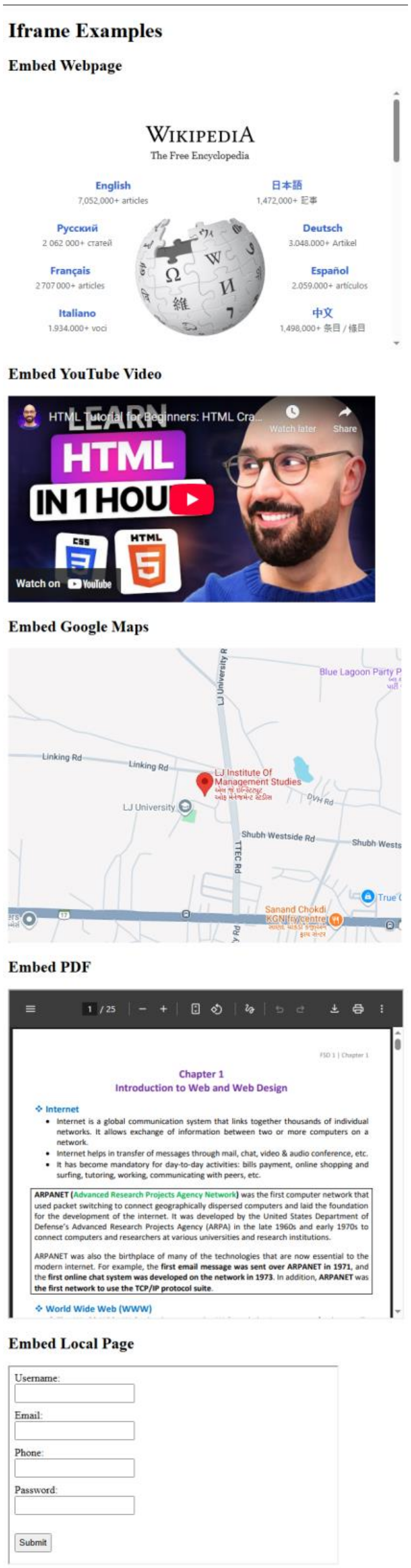
<iframe>

- Embeds another webpage, map, video (like YouTube), pdf etc inside your page.

Attribute	Purpose	Example from your code
src	URL or file path of embedded resource	src="https://www.wikipedia.org"
width	Sets width in pixels	width="600"
height	Sets height in pixels	height="400"
frameborder	Shows/hides border (deprecated)	frameborder="0"
style	Inline CSS (styling)	style="border:0;"
allowfullscreen	Allows fullscreen (mainly for video)	allowfullscreen (YouTube iframe)
loading	Lazy/eager load for performance	loading="lazy" (Google Maps iframe)
title	Accessibility/SEO description	title="YouTube video player"

Example:

```
<body>
  <h1>Iframe Examples</h1>
  <!-- 1. Another Webpage -->
  <h2>Embed Webpage</h2>
  <iframe src="https://www.wikipedia.org" width="600" height="400" frameborder="0"></iframe>

  <!-- 2. YouTube Video -->
  <h2>Embed YouTube Video</h2>
  <iframe width="560" height="315"
src="https://www.youtube.com/embed/qz0aGYrrlhU?si=eNLgc597jsbAK9-2" title="YouTube video
player" frameborder="0"  allowfullscreen></iframe>

  <!-- 3. Google Maps -->
  <h2>Embed Google Maps</h2>
  <iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3672.8940065186866!2d72.4852
8877603572!3d22.990924817507025!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x395e9a
ee615392b1%3A0x30d4b8160142adf3!2sL%20J%20Institute%20Of%20Management%20Studies!5e0
!3m2!1sen!2sin!4v1756790030904!5m2!1sen!2sin" width="600" height="450" style="border:0;"
allowfullscreen="" loading="lazy"></iframe>

  <!-- 4. PDF Document -->
  <h2>Embed PDF</h2>
  <iframe src="Chapter 1-Introduction.pdf" width="600" height="500"></iframe>

  <!-- 5. Local HTML File -->
  <h2>Embed Local Page</h2>
  <iframe src="fv.html" width="500" height="300" frameborder="1"></iframe>
</body>
```



Iframe Examples

Embed Webpage

Embed YouTube Video

Embed Google Maps

Embed PDF

FSD 1 | Chapter 1

Chapter 1
Introduction to Web and Web Design

❖ Internet

- Internet is a global communication system that links together thousands of individual networks. It allows exchange of information between two or more computers on a network.
- Internet helps in transfer of messages through mail, chat, video & audio conference, etc.
- It has become mandatory for day-to-day activities: bills payment, online shopping and surfing, tutoring, working, communicating with peers, etc.

ARPANET (Advanced Research Projects Agency Network) was the first computer network that used packet switching to connect geographically dispersed computers and laid the foundation for the development of the internet. It was developed by the United States Department of Defense's Advanced Research Projects Agency (ARPA) in the late 1960s and early 1970s to connect computers and researchers at various universities and research institutions.

ARPANET was also the birthplace of many of the technologies that are now essential to the modern internet. For example, the **first email message was sent over ARPANET in 1971**, and the **first online chat system was developed on the network in 1973**. In addition, **ARPANET** was **the first network to use the TCP/IP protocol suite**.

❖ World Wide Web (WWW)

Embed Local Page

Username:

Email:

Phone:

Password:

❖ HTML Lists:

A list is a record of short pieces of related information or used to display the data orany information on web pages in the ordered or unordered form. For instance, to purchase the items, we need to prepare a list that can either be ordered or unorderedlist which helps us to organize the data & easy to find the item.

There are two types of lists:

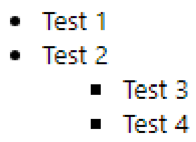
1. **Unordered list:** An unordered list starts with the <ul> tag. Each list item startswith the <li> tag. The list items will be marked with disc (small black circles) by default.

Attribute:

type: type = ”disc/square/circle/none”

```
<ul>
          <li>Test 1</li>
          <li>Test 2</li>
          <ul type="square">
             <li>Test 3</li>
             <li>Test 4</li>
          </ul>
</ul>
```

Output:



```
•  Test 1
•  Test 2
       ▪  Test 3
       ▪  Test 4
```

type="disc":
Items are marked with **filled circles** as bullet points. This is default.

● Item 1
● Item 2
● Item 3

type="square":
Items are marked with **squares**.

■ Item 1
■ Item 2
■ Item 3

type="circle":
Items are marked with **hollow circles**.

○ Item 1
○ Item 2
○ Item 3

type="none":
No bullet points are shown for the list items.

Item 1
Item 2
Item 3

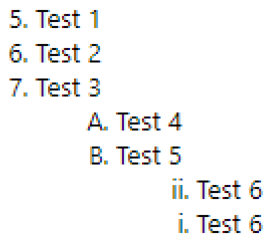
2. **Ordered list:** An ordered list starts with the <ol> tag. Each list item startswith the <li> tag. The list items will be marked with numbers by default:

Attribute:

type: type=” 1/ i/ I/ a/ A”
start: Specifies the start value of an ordered list
reversed: Specifies that the list order should be reversed (9,8,7...)

```
    <ol start="5">
            <li>Test 1</li>
            <li>Test 2</li>
            <li>Test 3</li>
            <ol type="A">
                <li>Test 4</li>
                <li>Test 5</li>
                <ol reversed type="i">
                    <li>Test 6</li>
                    <li>Test 6</li>
                </ol>
            </ol>
        </ol>
```

Output:

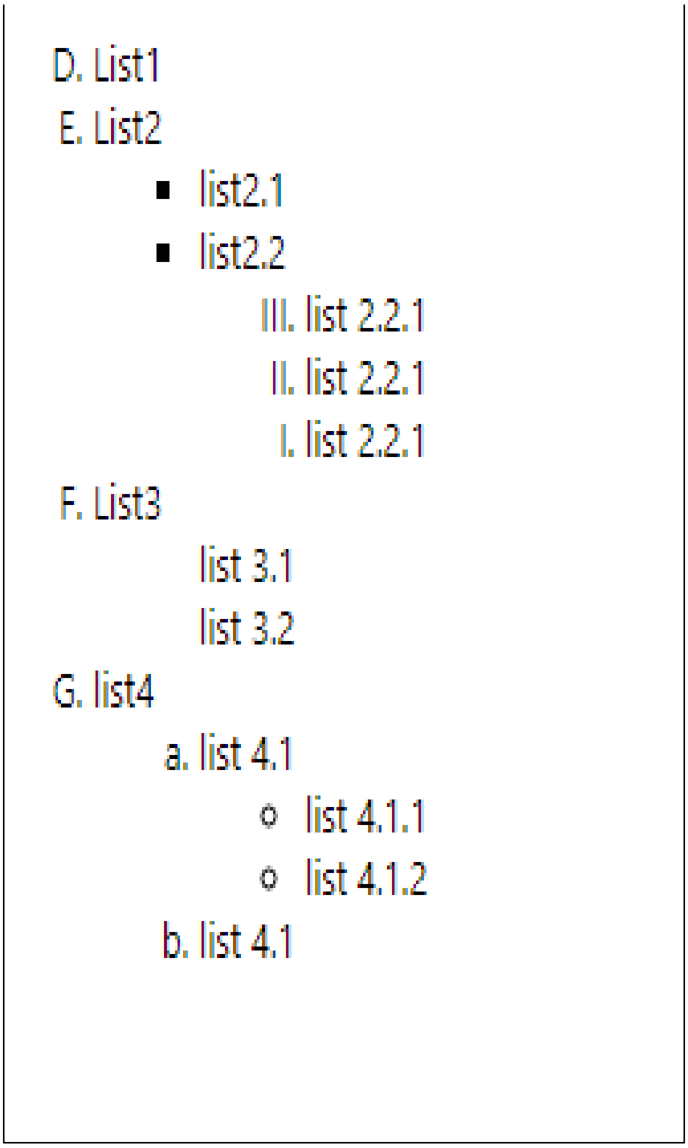


```
5. Test 1
6. Test 2
7. Test 3
        A. Test 4
        B. Test 5
                ii. Test 6
                 i. Test 6
```

For type ‘a’ 27,28,29.. letters are aa,ab,ac… respectively.

Nested List Example:

```
<html>
   <head><title>Nested List</title></head>
   <body>
      <ol type="A" start="4">
         <li>List1</li>
         <li>List2</li>
         <ul type="square">
            <li>list2.1</li>
            <li>list2.2</li>
               <ol type="I" reversed >
                  <li>list 2.2.1</li>
                  <li>list 2.2.1</li>
                  <li>list 2.2.1</li>
               </ol>
         </ul>
         <li>List3</li>
         <ul type="none">
            <li>list 3.1</li>
            <li>list 3.2</li>
         </ul>
         <li>list4</li>
         <ol type="a">
            <li>list 4.1</li>
               <ul type="circle">
                  <li>list 4.1.1</li>
                  <li>list 4.1.2</li>
               </ul>
            <li>list 4.1</li>
         </ol>
      </ol>
   </body>
</html>
```



```
D. List1
E. List2
      ■  list2.1
      ■  list2.2
            III. list 2.2.1
             II. list 2.2.1
              I. list 2.2.1
F. List3
         list 3.1
         list 3.2
G. list4
      a. list 4.1
            ○  list 4.1.1
            ○  list 4.1.2
      b. list 4.1
```

❖ HTML Table:

- ✓ The <table> tag defines HTML table.
- ✓ An HTML table consists of one <table> element and one or more <tr>, <th>, and <td> elements.
- ✓ The <tr> element defines a table row, the <th> element defines a table header, and the <td> element defines a table cell.
- ✓ An HTML table may also include <caption>
- ✓ , <thead>, <tfoot>, and <tbody> elements.

HTML tags for table:

<caption>	The <caption> tag defines a table caption. The <caption> tag must be inserted immediately after the <table> tag. **Tip:** By default, a table caption will be center-aligned above a table. However, the CSS properties text-align and caption-side can be used to align and place the caption.
<table>	Defines a table.
<th>	Defines a header cell in a table
<tr>	Defines a row in a table
<td>	Defines a cell in a table
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table

HTML <table> tag attributes:

Attribute	Value	Description
align	right left center justify char	Deprecated − Visual alignment.
bgcolor	rgb(x,x,x) #hexcode colorname	Deprecated − Specifies the backgroundcolor of the table.

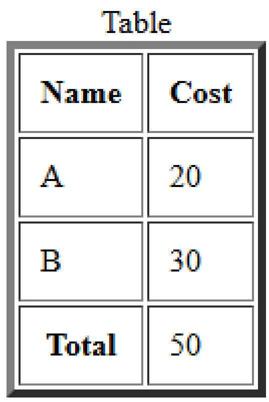
border	pixels	Deprecated − Specifies the border width.A value of "0" means no border.
cellpadding	pixels or %	Deprecated − Specifies the space betweenthe cell borders and their contents.
cellspacing	pixels or %	Deprecated − Specifies the space betweencells.
rules	none groups rows cols all	Deprecated – The **HTML <table> rules Attribute** is used to *specify which parts of the inside borders that should be visible*. **Syntax:** <table rules="value"> **Attribute Values:** • **none:** It does not create any lines. • **groups:** It create lines between row and column groups. • **rows:** It creates line between the rows. • **cols:** It creates line between the columns. • **all:** It creates line between the rows and columns. **Note:** The <table> rules Attribute is not supported by HTML 5.
width	pixels or %	Deprecated .Specifies the width of the table.

<td><th><tr> tags attributes:

Attribute	Value	Description
align	right left center justify char	Deprecated – Visual alignment.
bgcolor	rgb(x,x,x) #hexcode colorname	Deprecated − Specifies the backgroundcolor of the cell.

colspan	Number of columns to merge	Number of columns a header cell shouldspan
rowspan	Number of rows to merge	Set the number of rows a header cell shouldspan.

Example 1:



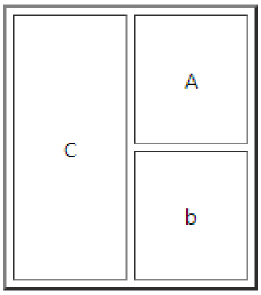
Table

Name	**Cost**
A	20
B	30
Total	50

```
<table border="4" cellpadding="10">
        <caption>Table</caption>
        <thead>
          <tr>
            <th>Name</th>
            <th>Cost</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>A</td>
            <td>20</td>
          </tr>
          <tr>
            <td>B</td>
            <td>30</td>
          </tr>
        </tbody>
        <tfoot>
          <tr>
            <th>Total</th>
            <td>50</td>
          </tr>
```

```
      </tfoot>
    </table>
```

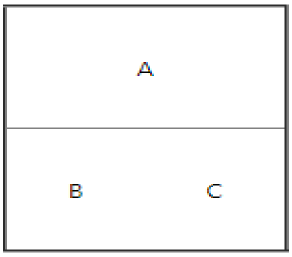
Example 2



C	A
	b

```
 <table border="2" cellspacing="4" cellpadding="30">
      <tr>
        <td rowspan="2">C</td>
        <td>A</td>
      </tr>
      <tr>
        <td>b</td>
      </tr>
    </table>
```

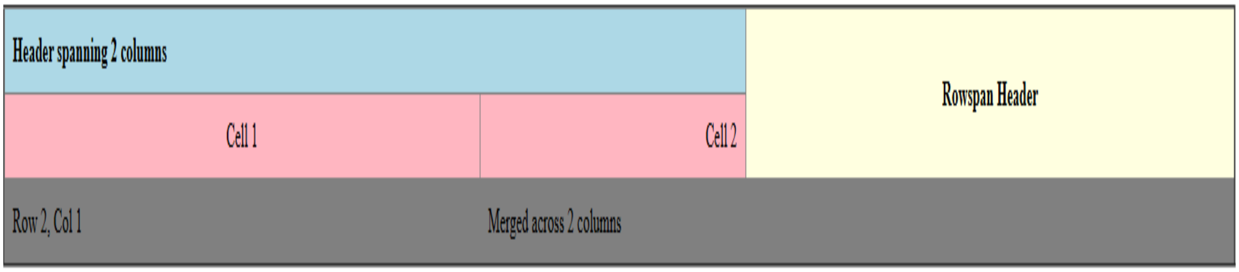
Example3:



A	
B	C

```
<table border="2" rules="rows" cellpadding="30">
      <tr>
        <td colspan="2" align="center">A</td>
      </tr>
      <tr>
        <td>B</td>
        <td>C</td>
      </tr>
</table>
```

Example 4

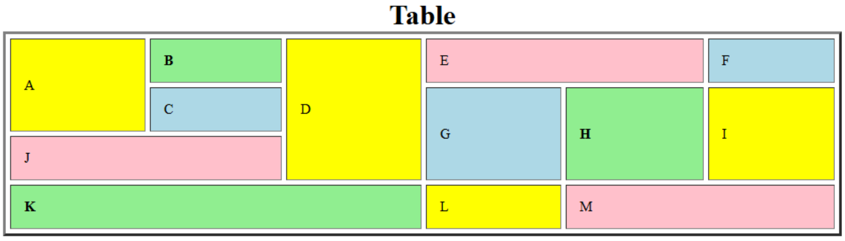


Header spanning 2 columns		Rowspan Header
Cell 1	Cell 2	
Row 2, Col 1	Merged across 2 columns	

```
<!DOCTYPE html>
<html>
<head>
  <title>Deprecated Table Attributes Example</title>
</head>
<body>

<!-- Using deprecated attributes just for demonstration -->
<table align="center" bgcolor="lightyellow" border="2" cellpadding="10" cellspacing="5" rules="all"
width="80%">

  <tr bgcolor="lightblue">
    <th align="left" colspan="2">Header spanning 2 columns</th>
    <th rowspan="2" bgcolor="lightyellow">Rowspan Header</th>
  </tr>

  <tr bgcolor="lightpink">
    <td align="center">Cell 1</td>
    <td align="right">Cell 2</td>
  </tr>

  <tr bgcolor="gray">
    <td>Row 2, Col 1</td>
    <td colspan="2" align="justify">Merged across 2 columns</td>
  </tr>
</table>
</body>
</html>
```

- ✓ If you want **visible grid lines** → use rules="all" and set cellspacing="0".
- ✓ If you want **spacing between cells** → use cellspacing, but then the effect of rules is lost.

Example 5



Table

<table>
<tr><td rowspan="2">A</td><td><b>B</b></td><td rowspan="3">D</td><td colspan="2">E</td><td>F</td></tr>
<tr><td>C</td><td rowspan="2">G</td><td rowspan="2"><b>H</b></td><td rowspan="2">I</td></tr>
<tr><td colspan="2">J</td></tr>
<tr><td colspan="3"><b>K</b></td><td>L</td><td colspan="2">M</td></tr>
</table>

```
<html>
   <head><title> Table </title></head>
   <body align='center'>
      <table border='3px' width='50%' align="center" cellpadding='15' cellspacing="5">
         <caption><strong><font size="6"> Table</font></strong></caption>
         <tr>
            <td rowspan='2' bgcolor="yellow"> A</td>
            <td bgcolor="lightgreen"> <b>B</b> </td>
            <td rowspan='3' bgcolor="yellow"> D </td>
            <td colspan='2' bgcolor="pink"> E </td>
            <td bgcolor="lightblue"> F </td>
         </tr>
         <tr>
            <td bgcolor="lightblue"> C </td>
            <td rowspan='2' bgcolor="lightblue"> G </td>
            <td rowspan='2' bgcolor="lightgreen"> <b>H</b> </td>
            <td rowspan='2' bgcolor="yellow"> I </td>
         </tr>
         <tr>
            <td colspan='2' bgcolor="pink"> J </td>
         </tr>
         <tr>
            <td colspan='3' bgcolor="lightgreen"> <b>K</b> </td>
            <td bgcolor="yellow"> L </td>
            <td colspan='2' bgcolor="pink"> M </td>
         </tr>
      </table>
   </body></html>
```

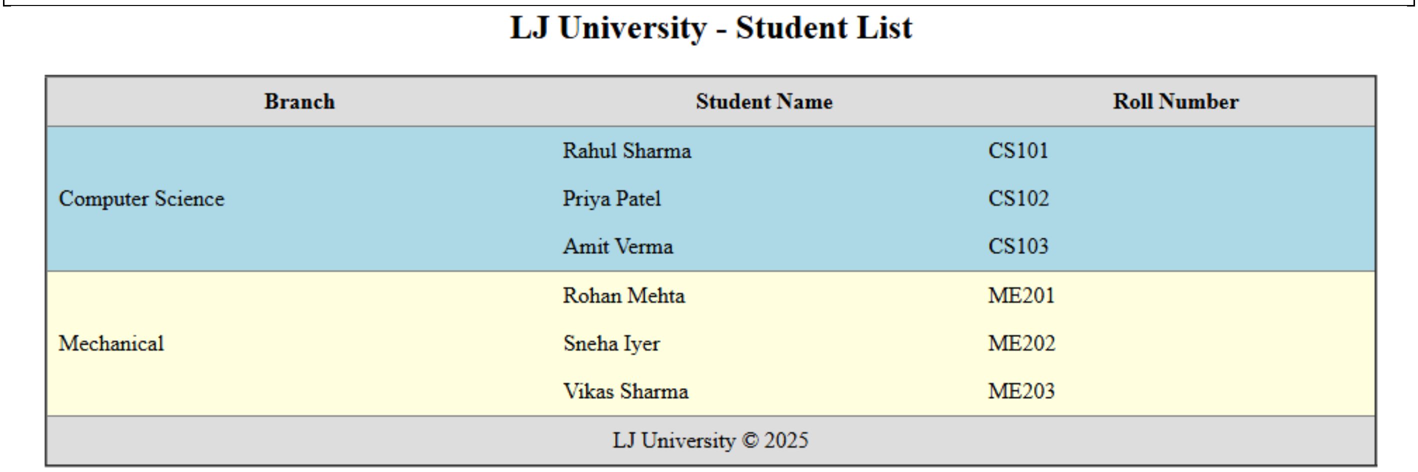
Example 6

Design an HTML table for **LJ University - Student List** that displays the **Branch, Student Name, and Roll Number** of students.

The table should:

1. Use the **rules="groups" attribute** so that horizontal lines are drawn **only between row groups** (different branches).
2. Include at least **two different branches** (e.g., Computer Science, Mechanical).
3. Each branch should have **multiple students**, and the **branch name should span multiple rows** using the rowspan attribute.
4. Add a **table header** (<thead>) with column titles (Branch, Student Name, Roll Number).
5. Add a **footer row** (<tfoot>) with copyright information for LJ University.
6. Show background color for different row groups (<tbody>) so that each branch is visually distinct.

```
<!DOCTYPE html>
<html>
<head>
  <title>College Table - Rules Groups Example</title>
</head>
<body>

<h2 align="center">LJ University - Student List</h2>

<table border="2" rules="groups" cellspacing="0" cellpadding="8" align="center" width="50%">


  <thead bgcolor="#ddd">
    <tr>
      <th>Branch</th>
      <th>Student Name</th>
      <th>Roll Number</th>
    </tr>
  </thead>

  <tbody bgcolor="lightblue">
    <tr>
      <td rowspan="3">Computer Science</td>
      <td>Rahul Sharma</td>
      <td>CS101</td>
    </tr>
    <tr>
      <td>Priya Patel</td>
      <td>CS102</td>
    </tr>
    <tr>
      <td>Amit Verma</td>
      <td>CS103</td>
    </tr>
  </tbody>

  <tbody bgcolor="lightyellow">
    <tr>
      <td rowspan="3">Mechanical</td>
      <td>Rohan Mehta</td>
      <td>ME201</td>
    </tr>
    <tr>
      <td>Sneha Iyer</td>
      <td>ME202</td>
    </tr>
    <tr>
      <td>Vikas Sharma</td>
      <td>ME203</td>
    </tr>
  </tbody>

  <tfoot bgcolor="#ddd">
    <tr>
      <td colspan="3" align="center">LJ University © 2025</td>
    </tr>
  </tfoot>
</table>

</body>
</html>
```



LJ University - Student List

Branch	Student Name	Roll Number
Computer Science	Rahul Sharma	CS101
	Priya Patel	CS102
	Amit Verma	CS103
Mechanical	Rohan Mehta	ME201
	Sneha Iyer	ME202
	Vikas Sharma	ME203
LJ University © 2025		