601 - HTML Layout

HTML layouts are a technique used to divide a web page into multiple sections, making it easier to apply styles, organize content, and manage operations efficiently. This division improves readability, accessibility, and overall user experience.

HTML layout is achieved through elements
like <u><header></u>, <u><nav></u>, <u><main></u>, <u><article></u>, <u><section></u>, <u><aside></u>, and <u><footer></u> which help organize content, define the page's sections, and improve SEO.

Header Section
Navigation Bar

Index	Content section

Footer Section

Syntax:

```
<header> Content... </header>
<nav> Content... </nav>
<main> Content... </main>
<footer> Content... </footer>
```

Layout with Semantic Tags

```
<html>

<body>
 <header>
  <h1>My Website</h1>
 </header>
 <main>
  <p>Welcome to my website!</p>
```

```
  </main>
  <footer>
    <p> 2025 My Website</p>
  </footer>
</body>

</html>
```

Output:

- <header> contains the main heading of the page.
- <main> holds the primary content.
- <footer> includes the footer information.

Layout Components

Layouts	Descriptions
Header	The top section of a webpage, often containing the website title, logo, or navigation links. The <header> tag is used to define this part of the page.
Navigation bar	A menu that provides links to the main sections of the website. The <nav> tag is used to define navigation elements in the webpage.
Index / Sidebar	An optional section often found on the side, used for additional content such as ads, links, or other related information.
Content Section	The central area where the primary content of the page is displayed. The <main> tag is used to define the main content of a webpage.
Footer	The bottom section of the webpage, typically containing contact information, legal links, or additional site-related information. The <footer> tag is used to define this area.

More Examples:

Layout with Additional Semantic Tags

```
<html>
```

```
<body>
 <header>
  <h1>My Blog</h1>
 </header>
 <nav>
  <a href="#">Home</a> | <a href="#">About</a> | <a href="#">Contact</a>
 </nav>
 <section>
  <h2>Latest Posts</h2>
  <article>
   <h3>Post Title</h3>
   <p>This is a brief introduction to the blog post.</p>
  </article>
 </section>
 <aside>
  <h2>About Me</h2>
  <p>Short bio or profile information.</p>
 </aside>
 <footer>
  <p>&copy; 2025 My Blog</p>
 </footer>
</body>

</html>
```

Output:

- <nav> provides navigation links for the website.
- <section> groups related content, here titled "Latest Posts."
- <article> represents an individual blog post.
- <aside> contains supplementary information, such as an "About Me" section

Styled Layout with Semantic Tags

```
<html>

<head>
 <style>
  header {
    background-color: green;

    color: white;

    text-align: center;

    padding: 5px;

  }

  nav {

    background-color: black;

    overflow: hidden;

    padding: 10px;
```

```
    }

    nav a {
      color: white;
      padding: 10px;
    }

    main {
      padding: 20px;
      background-color: cyan;
    }

    footer {
      background-color: green;
      color: white;
      text-align: center;
      padding: 5px;
    }
  </style>
</head>

<body>
  <header>
    <h1>Styled Page</h1>
  </header>
  <nav>
    <a href="#">Home</a>
    <a href="#">Services</a>
    <a href="#">Contact</a>
  </nav>
  <main>
    <h2>Welcome!</h2>
    <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Iure odit sunt rerum accusamus architecto optio facere
      aliquid unde, omnis fuga voluptatum quod ad recusandae cum inventore perferendis qui consectetur esse!</p>

    <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Iure odit sunt rerum accusamus architecto optio facere
      aliquid unde, omnis fuga voluptatum quod ad recusandae cum inventore perferendis qui consectetur esse!</p>
  </main>
  <footer>
    <p>&copy; 2025 Styled Page</p>
```

```
  </footer>
</body>

</html>
```

Output:

- **Header:** The <header> element contains the main heading of the page, providing a clear introduction.
- **Main:** The <main> element holds the primary content, ensuring semantic clarity and improved accessibility.

602 - HTML header Tag

The **<header> tag** is a semantic HTML element that is used to define the introductory or navigational content of a webpage or a section. Typically, a header contains elements like:

- *The website or page title*
- *Logo or branding*
- *Navigation menus*
- *Search bar*
- *Any introductory information relevant to the page or section*

Note: Header tag cannot be placed within a <u><footer>, <address></u>, or another <u><header></u> element.

Syntax:

<header> ...</header>

HTML <header> tag Example

Example 1: HTML Header Tag

In this example, the <header> tag is used inside the <article> element to define introductory content for that article.

```
<!DOCTYPE html>
<html>

<body style="font-size: 25px;">
 <h1 style="color: green;">Prodip ALways</h1>
 <h3>HTML Header Tag</h3>
 <hr>
 <article>
  <header>
   <h3 style="color: green;">
    ProdipAlways Learning
   </h3>
   <p>Posted by Prodip</p>
   <p>
    A Computer Science portal for always.
    It contains well written, well thought <br>
    and well explained computer science and
    programming articles.
   </p>
  </header>
 </article>
```

```
</body>

</html>
```

Output:

- **The <header> tag here contains:**
  - <h3>: 'ProdipAlways Learning'
  - <p> (Paragraph): 'Posted by GFG'
  - Another <p> (Paragraph): 'A Computer Science portal for geeks' – along with more information.

Important: The <header> is not just for the main page header (like a site title), but it can also be used for each article or section to provide the essential context — such as the article's title, author, and a short description.

HTML Header Tag Attributes

The <header> tag itself doesn't have attributes unique to it. However, it supports most of the global attributes of HTML, such as id, class, style, role, lang, data-*, and tabindex.

Additionally, you can apply styling to control its behavior and appearance using CSS. The <header> tag also supports Event Attributes in HTML, allowing you to add interactivity, such as onclick, onmouseover, and others, to enhance user experience.

Example 2:

```
<!DOCTYPE html>
<html>

<body>
 <h1>Prodip Always</h1>
 <h3>HTML Header Tag</h3>


 <!--HTML header tag starts here-->
 <header>
   <h1>This is the heading.</h1>
   <h4>This is the sub-heading.</h4>


   <p>This is the metadata.</p>
 </header>
 <!--HTML header tag ends here-->
</body>

</html>
```
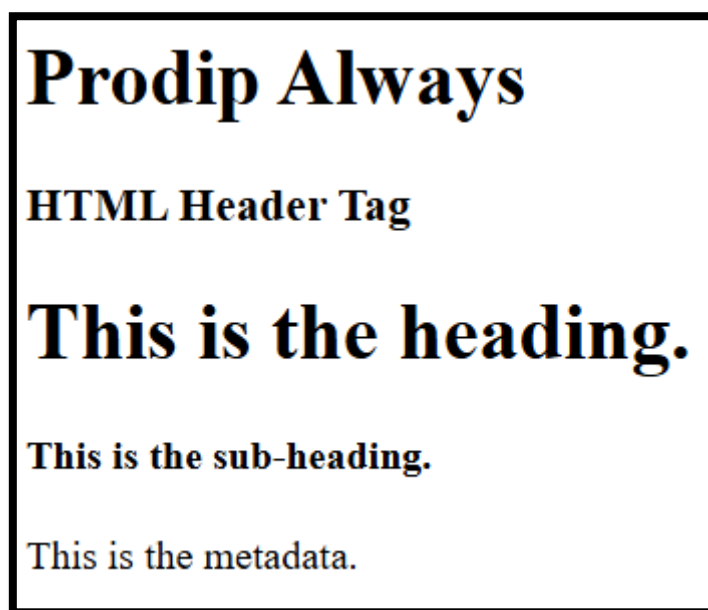
Output:

In this example:

- **Main content**: The document has a primary title (<h1>ProdipAlways) followed by a secondary heading (<h3>HTML Header Tag).

- **Header tag**: Inside the `<header>`, there's a primary heading (`<h1>`) and a subheading (`<h4>`) along with a paragraph (`<p>`).

Example 3:

This HTML code demonstrates the usage of the <header> tag to structure a navigation section of a webpage.

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

<body>
 <h1>Prodip Always</h1>
 <h3>HTML Header Tag</h3>

 <!--HTML header tag starts here-->
 <header>
   <a href="#">Algo</a> |
   <a href="#">DS</a> |
   <a href="#">Languages</a> |
   <a href="#">Interview</a> |
   <a href="#">Students</a> |
   <a href="#">Gate</a> |
   <a href="#">CS Subjects</a> |
   <a href="#">Quizzes</a><br><br>

   <div class="search-bar">
     <input type="text" placeholder="Search products...">
     <button>Search</button>
   </div>

 </header>
 <!--HTML header tag ends here-->

</body>

</html>
```

Output:

Prodip Always

HTML Header Tag

Algo | DS | Languages | Interview | Students | Gate | CS Subjects | Quizzes

Search products... | Search |

In this example:

- The navigation links (<a> tags) within the <header> allow users to quickly jump to different sections of the site. These links are separated by a pipe (|) symbol for readability.

Key Points to Remember About the <header> Tag

- **Multiple headers**: You can have multiple <header> tags on a page. For example, each section or article can have its own <header>, but there should only be one site-wide <header>.

- **Not just for titles**: While the <header> tag often contains a heading (<h1>, <h2>), it can also hold other elements like navigation links (<nav>), search bars, or introductory text.

- **Keep it clear**: The <header> tag should contain introductory content. It's not meant to hold all content; its purpose is to help introduce or navigate the content.

603 - HTML <nav> Tag

The <nav> tag in HTML is used to define navigation sections on a webpage. It typically contains links to important sections of the website, such as menus, tables of contents, or indexes, and is often structured within unordered lists (<ul>) or as standalone links.

How to Use the <nav> Tag

Links within the <nav> tag can either be standalone or structured within an unordered list (<ul>) for better organization. While it's common to use lists, it's not a strict requirement.

Syntax:

```
<nav>
  <!-- Your navigation links here -->
</nav>
```

Note: The *<nav>* tag also supports the Global Attribute and Event Attributes in HTML.

Example 1: Basic Implementation of <nav>

In this example, we will see the <nav> tag, containing a simple set of navigation links for different sections like Home, Interview, Languages, Data Structure, and Algorithm. The links are separated by vertical bars.

```
<!DOCTYPE html>
<html>


<head>
  <title>HTML nav Tag</title>
</head>


<body>


  <h2> HTML nav Tag</h2>


  <!-- nav tag starts -->
  <nav>
    <a href="#">Home</a> |
    <a href="#">Interview</a> |
    <a href="#">Languages</a> |
    <a href="#">Data Structure</a> |
    <a href="#">Algorithm</a>
  </nav>


  <!-- nav tag ends -->
```

```
</body>

</html>
```

Output:

Example 2: Styling the <nav> Tag Using CSS

In this example, we apply CSS styling to the <nav> section to enhance its appearance. We will style the navigation links with a green background, white text, and remove the underlines for a cleaner look.

```
<!DOCTYPE html>
<html>

<head>
  <title>HTML Nav Tag</title>
  <style>
    nav {
      border: 1px;
      background-color: green;
      color: white;
      padding: 10px;
    }

    a {
      text-decoration: none;
      color: white;
      font-size: 20px;
      padding: 10px;
    }
  </style>
</head>

<body>
```

```
<h2>HTML nav Tag</h2>

<!-- nav tag starts -->
<nav>
  <a href="#">Home</a> |
  <a href="#">Interview</a> |
  <a href="#">Gate</a> |
  <a href="#">Data Structure</a> |
  <a href="#">Algorithm</a>
</nav>
<!-- nav tag ends -->

</body>

</html>
```

Output:

604 - HTML <main> Tag

The **HTML <main> Tag** defines a document's main content, which should be unique. It excludes content like sidebars, navigation, logos, and copyright info, ensuring unique document-specific material within.

Note: The document must not contain more than one **<main>** element. The **<main>** element should not be a child element of an **<article>**, **<aside>**, **<footer>**, **<header>**, or **<nav>** element.

Syntax:

```
<main>
   // contents of main Element
</main>
```

HTML <main> Tag Examples

Example: In this example, we are using the <main> tag. It encapsulates main content, including headings and articles, conforming to HTML5 semantics.

```
<!DOCTYPE html>

<html>


<head>

 <title>The &lt;Main&gt; Tag</title>

</head>


<body>

 <h1>The &lt;Main&gt; Tag</h1>

 <main>

  <h1>Programming Languages</h1>

  <p>c programming, C++ Programming, Java Programming</p>


  <article>

   <h1>C Programming</h1>

   <p>C is a Procedural language</p>

  </article>


  <article>

   <h1>C++ Programming</h1>

   <p>C++ programming is a Object oriented Programming.</p>

  </article>


  <article>

   <h1>Java Programming</h1>
```

```
   <p>Java is a pure Object oriented Programming.</p>
  </article>
 </main>


</body>


</html>
```

Output:

The **<Main>** Tag

Programming Languages

c programming, C++ Programming, Java Programming

C Programming

C is a Procedural language

C++ Programming

C++ programming is a Object oriented Programming.

Java Programming

Java is a pure Object oriented Programming.

605 - HTML article Tag

The **HTML <article> tag** defines a self-contained, independent piece of content like a blog post, news article, or comment. It is designed for content that can be independently distributed, shared, or reused, providing semantic meaning to the content.

This tag is introduced in HTML5.

```
<!DOCTYPE html>
<html>


<body>
 <article>
   <h3>My First Blog Post</h3>
   <p>
     This is a brief introduction to my blog.
     I will share my thoughts on various topics.
   </p>
 </article>
</body>


</html>
```

Output:

My First Blog Post

This is a brief introduction to my blog. I will share my thoughts on various topics.

This tag is most often used in two contexts:

- On a page with a single piece of content, a single <article> element can be used to contain the main content and set it off from the rest of the page.

- On a page with multiple pieces of content (a blog index page, a search results page, a category page, news feed), multiple <article> elements can be used to contain each individual piece of content.

Either way, it is similar to the <div> element and displays the stylish work the same. However, using the <article> element instead of <div> provides more semantic information to screen readers, search engines, and third-party applications.

Note: This tag does not render as anything special in a browser, you have to use CSS for that.

Default CSS setting

Most browsers will display the Article element with the following values.

```css
article { display: block; }
```

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

<body>
  <article style="
      width: 300px;
      border: 2px solid gray;
      padding: 10px;
      border-radius: 10px;
      margin: 5px;">
    <img src="Logo.png" alt="" width="300" height="250" />

    <h1>Prodip Always</h1>
    <p>
      I am a Front End Web developer passionate about making error-free websites with 100% client satisfaction. I am passionate about learning and sharing my knowledge with others as publicly as possible. I love to solve real-world problems.
    </p>
  </article>
</body>

</html>
```

Output:

Prodip Always

I am a Front End Web developer passionate about making error-free websites with 100% client satisfaction. I am passionate about learning and sharing my knowledge with others as publicly as possible. I love to solve real-world problems.

606 - HTML <section> Tag

The Section tag defines the section of documents such as chapters, headers, footers, or any other sections. The section tag divides the content into sections and subsections.

The section tag is used when requirements of two headers or footers or any other section of documents are needed. Section tag grouped the generic block of related contents. The main advantage of the section tag is, it is a semantic element, that describes its meaning to both the browser and the developer.

Note: Section tag is used to distribute the content i.e., it distributes the sections and subsections.

Syntax

```
<section> Section Contents </section>
```

Example 1: The implementation of the section tag

```
<!DOCTYPE html>
<html>

<body>
 <!-- html section tag is used here -->
 <section>
  <h1>
   ProdipAlways section 1
  </h1>
  <p>
   Content of section 1
  </p>
 </section>
 <section>
  <h1>
   ProdipAlways section 2
  </h1>
  <p>
   Content of section 2
  </p>
 </section>
 <section>
  <h1>
   ProdipAlways section 3
  </h1>
  <p>
```
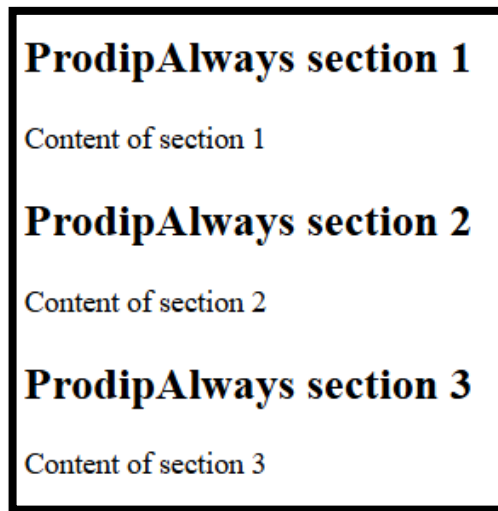
```
      Content of section 3

   </p>

 </section>

</body>


</html>
```

Output:

Nested Section tag:

The section tag can be nested. The font size of subsection is smaller than section tag if the text contains the same font property. The subsection tag is used for organizing complex documents. A rule of thumb is that section should logically appear in outline of the document.

Example 2: The implementation of the nested section tag

```html
<!DOCTYPE html>
<html lang="en">


<head>
 <title>Document</title>
 <style>
   body {
     text-align: center;
   }
 </style>
</head>


<body> <!-- html section tag is used here -->
 <section>
```

```html
<h1 style="color: green;">
 ProdipAlways: Section 1
</h1>
<p>
 Content of section 1
</p>
<section>
 <h1>Subsection</h1>
 <h1>Subsection</h1>
</section>
</section>
<section>
 <h1>ProdipAlways: Section 2</h1>
 <p>Content of section 2</p>
 <section>
  <h1>Subsection</h1>
  <h1>Subsection</h1>
 </section>
</section>
</body>

</html>
```

Output:

607 - HTML aside Tag

The **<aside>** tag is used to describe the main object of the web page more shortly like a highlighter. It identifies the content that is related to the primary content of the web page but does not constitute the main intent of the primary page. The <aside> tag contains mainly author information, links, related content, and so on.

Note: The <aside> tag supports **Global attributes** and **Event attributes** in HTML. The <aside> tag is new in <u>HTML5</u>. This tag does not render anything special in a browser you have to use <u>CSS</u> for that.

```
<!DOCTYPE html>
<html>

<body>
 <article>
  <h2>Learning HTML</h2>
  <p>
    HTML is the standard markup language for creating web pages.
  </p>
  <aside>
   <h3>Did You Know?</h3>
   <p>
     HTML stands for HyperText Markup Language.
   </p>
  </aside>
  <p>
   It is easy to learn and widely used in web development.
  </p>
 </article>
</body>

</html>
```

Output:

Learning HTML

HTML is the standard markup language for creating web pages.

Did You Know?

HTML stands for HyperText Markup Language.

It is easy to learn and widely used in web development.

Key Points:

- **Placement:** The <aside> tag is often placed

    o In a sidebar of a webpage.

    o Inside an <article> tag for content that provides additional context (e.g., a glossary, quote, or related links).

- **Common Use Cases:**

    o Sidebars with links to related content.

    o Advertisements or promotions.

    o Brief author bios or quotes.

Aside tag use as sidebar

```
<!DOCTYPE html>
<html>

<head>
 <style>
   article {
    width: 50%;
    padding: 10px;
    float: left;
   }

   aside {
    width: 40%;
    float: right;
    background-color: green;
    color: white;
    padding: 5px;
```

```
      margin: 10px;

      height: 100px;

     }

   </style>

 </head>


 <body>

  <article>

   <h1>Heading . . .</h1>

   <p>

     Aside tag is use to display important information about the primary page.

   </p>

  </article>


  <aside>

   <h1>Aside tag example</h1>

   <p>Aside tag content. . .</p>

  </aside>

 </body>


 </html>
```

Output:

608 - HTML footer Tag

The **<footer>** tag in HTML is used to define the footer section of an HTML document.

- The footer section typically contains information such as contact information, sitemap, back-to-top links, related documents, copyright, etc.

- The footer tag is a <u>semantic tag</u> included in HTML (in 2014) along with other tags like article, nav, header, etc.

- It is not mandatory, but adds to clear structure to the document and useful for SEO.

HTML Footer Code Example:

```
<!DOCTYPE html>
<html>


<body>
 <footer>
   <a href="#">About Us</a> |
   <a href="#">Privacy Policy</a> |
   <a href="#">Careers</a>
   <p>@prodipalways, Some rights reserved</p>
 </footer>
</body>


</html>
```
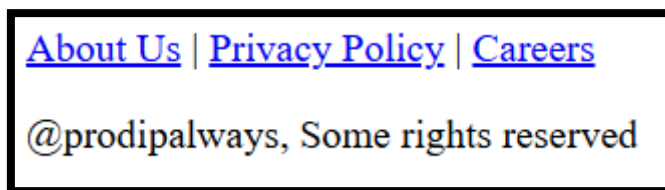
Output:

About Us | Privacy Policy | Careers

@prodipalways, Some rights reserved

Key Points:

- The <footer> tag is typically used to wrap content at the bottom of a page or section.

- The <footer> tag also supports the <u>Global Attributes</u> and <u>Event Attributes</u> in HTML.

- Contact information inside a **<footer>** element should go inside an <address> tag.

- We can have several **<footer>** elements in one document.

Styling the <footer> Tag with CSS

By default, the <footer> tag has only the **display: block property**. You can customize its style using CSS.

```
<!DOCTYPE html>
<html>

<head>
 <style>
  body {
    font-family: Arial, sans-serif;
  }

  footer {
    display: flex;
    justify-content: space-around;
    background-color: #333;
    color: #fff;
    padding: 20px;
  }

  .column {
    width: 27%;
  }

  p {
    font-size: 20px;
    font-weight: bold;
    margin-bottom: 10px;
  }

  ul {
    list-style-type: none;
    padding: 0;
  }

  li {
    margin-bottom: 5px;
  }
 </style>
</head>
```

```html
<body>
  <footer>
    <div class="column">
      <p>Company</p>
      <ul>
        <li>About Us</li>
        <li>Careers</li>
        <li>Privacy Policy</li>
        <li>Contact Us</li>
      </ul>
    </div>

    <div class="column">
      <p>Learn</p>
      <ul>
        <li>Algorithms</li>
        <li>Data Structures</li>
        <li>Languages</li>
        <li>CS Subjects</li>
        <li>Video Tutorials</li>
      </ul>
    </div>

    <div class="column">
      <p>Practice</p>
      <ul>
        <li>Company-wise</li>
        <li>Topic-wise</li>
        <li>Contests</li>
        <li>Subjective Questions</li>
      </ul>
    </div>
  </footer>
</body>

</html>
```

Output:

Best Practices for Using the HTML <footer> Tag

- **Avoid Overloading with Content**: While the footer is meant for additional or supplementary content, avoid overloading it with too much information. Stick to key links and essential details to prevent clutter.
- **Ensure Mobile Responsiveness**: When designing a footer, make sure that the content is accessible and looks good on all screen sizes. Footer content should be arranged in a user-friendly manner, especially on mobile devices.
- **Use Clear and Descriptive Links**: In the footer, links to external sites, social media, or important pages (like privacy policy or terms of service) should be clearly labeled so users know what to expect when clicking.