

HTML - Layouts

HTML Layouts specifies the arrangement of components on an HTML web page. A good layout structure of the webpage is important to provide a user-friendly experience on our website. It takes considerable time to design a website's layout with a great look and feel.



HTML Layout Elements

The below-listed elements are used to create the HTML layout's structure:

Elements	Description
header	The header tag is used to add a header section in an HTML web page. All the content inside this tag will be on top of the webpage.
nav	It represents a section of a page within the webpage, where it has hyperlinks to other pages or parts within the page (just like the menu bar).
section	It defines a major part of the web page where all the important content will be displayed.
footer	The footer tag defines the footer section of the webpage. This section contains copyright information and other important details. It will always be at the bottom of the webpage.
article	It specifies independent self-containing content such as a forum post, magazine, any blog post, and so on.

aside	It specifies a section of content that is directly or indirectly related to the main content (like a sidebar).
summary	It specifies a caption for the <details> element.
details	It specifies a tag for additional information. It requires the <summary> element.

Visit this chapter to learn about the various elements with examples that are used for creating HTML layouts: [HTML Layout Elements](#)

Example of HTML Layouts

Here are some examples that illustrate HTML layout designs. CSS and CSS frameworks are used to design the layout. The above-mentioned elements are used to create layout structure.

Define an HTML Layout

We can achieve HTML layout by simply using tags like **<header>**, **<footer>**, and **<nav>**. The following code shows how to make an HTML layout:

</>

Open Compiler

```
<!DOCTYPE html>
<html lang="en">
<head>
  <style>
    .header {
      background-color: #b3e0f2;
      text-align: center;
      padding: 20px;
      font-size: 2em;
      font-weight: bold;
    }
    .container {
      display: flex;
    }
    .sidebar {
      width: 20%;
      background-color: #f4f4f4;
      padding: 20px;
    }
    .content {
      width: 80%;
      background-color: #f9f9f9;
      padding: 20px;
    }
    .footer {
```

```

        background-color: #b3e0f2;
        text-align: center;
        padding: 10px;
        position: fixed;
        width: 100%;
        bottom: 0;
    }
</style>
</head>
<body>
    <div class="header">
        Title
    </div>
    <div class="container">
        <div class="sidebar">
            <a href="#">Home</a>
            <a href="#">Learn HTML</a>
            <a href="#">About Us</a>
        </div>
        <div class="content">
            <h1>Home</h1>
            <p>This is a home page.</p>
        </div>
    </div>
    <div class="footer">
        Footer
    </div>
</body>
</html>

```

Define an HTML Layout Using Bootstrap

The CSS Bootstrap library can be used to make layouts in HTML. The following code shows how it's done:

```

</>
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Bootstrap CDN Link -->
    <link href=
"https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
        rel="stylesheet">
</head>

```

Open Compiler

```
<body>
  <!-- Header -->
  <header class="bg-info text-white text-center py-4">
    <h1>Title</h1>
  </header>

  <!-- Main Container -->
  <div class="container-fluid">
    <div class="row">
      <!-- Sidebar -->
      <nav class="col-md-3 col-lg-2 d-md-block bg-light sidebar">
        <div class="sidebar-sticky">
          <ul class="nav flex-column">
            <li class="nav-item">
              <a class="nav-link active" href="#">
                Home
              </a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="#">
                Learn HTML
              </a>
            </li>
            <li class="nav-item">
              <a class="nav-link" href="#">
                About Us
              </a>
            </li>
          </ul>
        </div>
      </nav>

      <!-- Content -->
      <main role="main" class="col-md-9 ml-sm-auto col-lg-10 px-4">
        <h2>Home</h2>
        <p>This is a home page.</p>
      </main>
    </div>
  </div>

  <!-- Footer -->
  <footer class="bg-info text-white text-center py-3">
    Footer
  </footer>
```

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

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Different Techniques to Create HTML Layouts

There are four different techniques (ways) to create and design HTML layouts; you can create simple and multicolumn layouts using these techniques:

- **CSS Float Property:** A classic way to control position and formatting on a webpage.
- **CSS Flexbox Property:** Used for dynamic layout and to position elements in different screen sizes.
- **CSS Grid Property:** Create a complex grid-like layout
- **CSS frameworks:** The framework, like [Bootstrap](#), allows to create dynamic HTML layouts.

To learn how to use CSS for making HTML layouts, visit the chapter **[Layouts Using CSS](#)**