

Lesson:

Introduction to CSS Layout



Topics Covered

- What is CSS Layout?
- Layout techniques.
- What is CSS Positioning?
- Positioning techniques

What is CSS Layout?

CSS layout refers to the process of positioning and arranging elements on a web page using Cascading Style Sheets (CSS). The layout of a web page is an essential aspect of its design, as it affects the visual hierarchy of the content and its usability.

Layout techniques

CSS provides a range of layout techniques, including the use of normal layout, float, flexbox, grid and positioning.

These techniques allow web designers and developers to create complex layouts that adapt to different screen sizes and devices.

In this module we will give overview of common layouts,

- Normal Layout
- Float
- Flex
- Grid

In later modules, we will cover them in depth.

Normal Layout:

The normal layout is the default layout that web browsers use to display web pages. In this layout, elements are placed in the order in which they appear in the HTML document, and their positions are determined by their default positioning properties.

These properties can be adjusted using CSS to create custom layouts, but the options are limited compared to other layout techniques.

Normal Layout

Elements are placed in the order in which they appear in the HTML document, and their positions are determined by their default positioning properties.

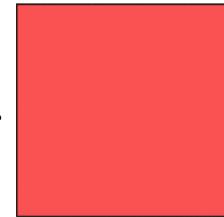
1. Item 1
2. Item 2
3. Item 3

Float Layout:

The float layout technique involves positioning elements by "floating" them to one side of their container. This is typically used to position images or other elements next to text or other content.

Floats can be used to create complex layouts, but they can also be tricky to work with, as they can cause unexpected behaviour when used improperly.

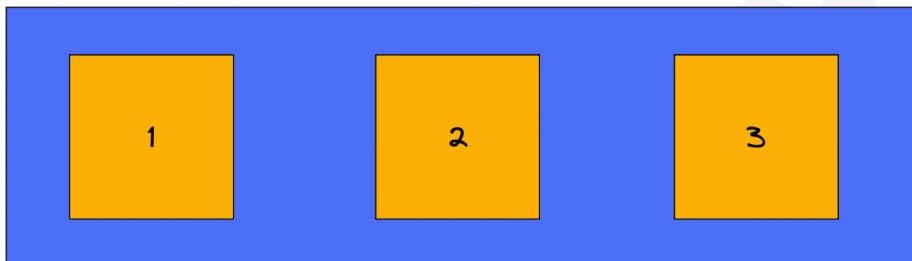
*Lorem ipsum dolor sit amet, consectetur adipiscing elit.
 Phasellus imperdiet, nulla et dictum interdum, nisi lorem
 egestas odio, vitae scelerisque enim ligula venenatis dolor.
 Maecenas nisl est, ultrices nec congue eget, auctor vitae
 massa. Fusce luctus vestibulum augue ut aliquet.*



Flex Layout:

The flex layout technique is based on a flexible box model, which allows elements to be arranged in rows or columns. This technique is ideal for creating responsive layouts that can adapt to different screen sizes and device orientations.

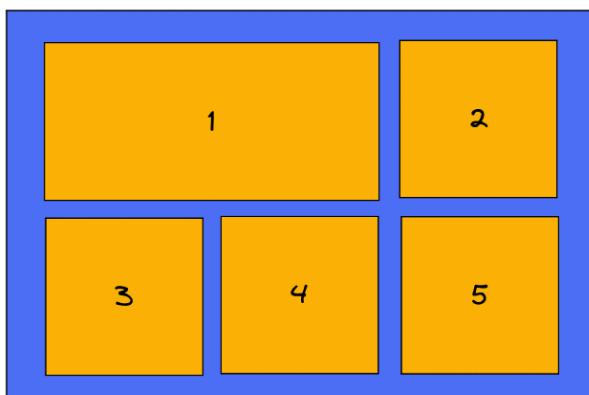
Flexbox offers a lot of control over the position and sizing of elements, making it a popular choice for modern web design.



Grid Layout:

The grid layout technique is similar to the flex layout, but with more advanced features for creating complex, multi-dimensional layouts. With grid layout, elements are positioned on a grid of rows and columns, and can span multiple rows or columns.

This allows for precise control over the layout of content, making it a powerful tool for web designers. Grid layout is also ideal for creating responsive layouts, as elements can be repositioned and resized automatically based on the screen size.



What is CSS Positioning?

Apart from layout, we sometimes need to position HTML elements. CSS Positioning refers to the process of controlling the position of elements on a webpage using CSS.

Positioning is a fundamental aspect of web design, as it allows designers to create custom layouts and control the visual hierarchy of content.

Positioning Techniques

1. Static Positioning:

The element is positioned based on its default position within the document's flow.

Here we will give an overview of positioning techniques, and explain in detail in later lessons.

2. Relative Positioning:

Relative positioning allows elements to be positioned relative to their normal position in the document flow.

It allows you to move an element up, down, left, or right relative to its normal position. This positioning technique is often used to create small adjustments to an element's position.

3. Absolute Positioning:

Absolute positioning allows elements to be positioned anywhere on the webpage, regardless of their position in the document flow.

It allows you to specify the exact position of an element using the top, right, bottom, and left properties. Absolute positioning is often used to create overlays, tooltips, and other types of dynamic content.

4. Fixed Positioning:

Fixed positioning is similar to absolute positioning, but the position of the element is fixed relative to the viewport, rather than the containing element. This means that the element will remain in the same position even when the user scrolls the webpage. This technique is often used for fixed headers, footers, or navigation bars.

5. Sticky Positioning:

Sticky positioning is a hybrid of relative and fixed positioning. This technique allows an element to behave as if it is positioned statically until it reaches a certain threshold, after which it becomes fixed. This technique is often used for creating sticky headers, sidebars, or navigation menus.