



Posición relativa, absoluta y fija en CSS

Última actualización: 29 de octubre de 2024

El **posicionamiento CSS** es un concepto fundamental en el diseño y desarrollo web que permite un control preciso sobre cómo se organizan los elementos en una página web. Existen tres tipos principales de posicionamiento CSS: relativo, absoluto y fijo.

¿Qué es el posicionamiento relativo?

[El posicionamiento relativo](#) es una [técnica CSS](#) que permite ajustar un elemento desde su posición normal. La sintaxis para el posicionamiento relativo es **position: relative;** Cuando se establecen las propiedades top, right, bottom e left de un elemento con **posicionamiento relativo**, este se mueve desde su ubicación original. Sin embargo, es importante tener en cuenta que otros elementos de la página no llenarán el espacio que deja el elemento movido.

Sintaxis

```
posición: relativa;
```

Ejemplo: Es un ejemplo de la propiedad position: relative;. El elemento se desplaza 50 px hacia la derecha mediante la propiedad left, pero aún ocupa su espacio original en el flujo del documento.

```
1  <!DOCTYPE html>
2  <html>
3
4  < cabeza >
5    < estilo >
6      div .relativo {
7        posición : relativa ;
8        izquierda : 50px ;
9        borde : 3px sólido #73AD21 ;
10     }
11    </ estilo >
12  </cabeza>
13
14  < cuerpo >
15    < h1 > posición: relativa; </ h1 >
16
17    < div class = "relativo" >
18      Este elemento tiene posición:relativa;
19    </div>
20  </ cuerpo >
21
22  </html>
```

Producción

position: relative;

This element has position:relative;

¿Qué es el posicionamiento absoluto?

¿Que es el posicionamiento absoluto:

Posición absoluta es otra técnica CSS que ajusta la posición de un elemento en relación con su elemento padre. Si no hay ningún elemento padre presente, se utiliza el cuerpo del documento como elemento padre. La sintaxis para el posicionamiento absoluto es **position: absolute ;**

Sintaxis

```
posición: absoluta;
```

Ejemplo: Es un ejemplo de propiedad position: absolute;. El elemento absolute está ubicado a 80 px de la parte superior y derecha de su antecesor más cercano (el contenedor relativo).

```
1  <!DOCTYPE html>
2  <html>
3
4  < cabeza >
5    < estilo >
6      div .relativo {
7        posición : relativa ;
8        ancho : 400px ;
9        height: 200px;
10       border: 3px solid #73AD21;
11     }
12
13     div.absolute {
14       position: absolute;
15       top: 80px;
16       right: 80px;
17       width: 200px;
18       height: 100px;
19       border: 3px solid #73AD21;
20     }
21   </style>
22 </head>
23
24 <body>
25   <h1>position: absolute;</h1>
26
27   <div class="relative">
28     This element has position: relative;
29     <div class="absolute">
30       This element has position: absolute;
31     </div>
32   </div>
33 </body>
34
35 </html>
```

Output

What is Fixed Positioning?

Fixed Position is a CSS technique that keeps an element in the same place even when the page is scrolled. The syntax for fixed positioning is **position: fixed;**. To position the element, we use top, right, bottom, and left properties.

Syntax

```
position: fixed;
```

Example: It is an example of position: fixed; property. The fixed element will remain in the same position (bottom-right corner) of the browser window, even when the page is scrolled.

```

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <style>
6          div.fixed {
7              position: fixed;
8              bottom: 0;
9              right: 0;
10             width: 300px;
11             border: 3px solid #73AD21;
12         }
13
14         div.absolute {
15             position: absolute;
16             top: 150px;
17             right: 80;
18             width: 200px;
19             height: 100px;
20             border: 3px solid #73AD21;
21         }
22     </style>
23 </head>
24
25 <body>
26     <h1>position: absolute;</h1>
27
28     <h2>position: fixed;</h2>
29     <div class="absolute">
30         This element has position: absolute;
31     </div>
32 </body>
33
34 </html>

```

Output

Differences Between Relative, Absolute, and Fixed Positioning

Property	Relative	Absolute	Fixed
Position Reference	Positioned relative to its normal location in the flow.	Positioned relative to the closest positioned ancestor or the document body if no ancestor is found.	Positioned relative to the browser window (viewport).
Affects Other Elements	No, it doesn't affect the position of other elements.	Yes, it is removed from the document flow, and other elements may shift.	No, it is removed from the document flow.
Scroll Behavior	Moves along with the page when scrolling.	Moves along with the page when scrolling.	Stays fixed in place even when the page is scrolled.
Original Space	Retains its original space in the document layout.	Does not retain its original space; other elements may occupy it.	Does not retain any space in the document layout.
Use Case	Used to move an element slightly from its original position.	Used for exact positioning within a container or page.	Used for sticky elements like headers, footers, or buttons.
Stacking Context	Can create a new stacking context if z-index is applied.	Creates a new stacking context based on the ancestor's stacking order.	Always creates a new stacking context.
Common Usage	Minor adjustments like nudging elements.	Precise positioning within a specific container or overlay effects.	Fixed navigation bars, banners, or call-to-action buttons.

CSS Positioning – FAQs

What is the main difference between relative and absolute positioning in CSS?

Relative positioning moves an element from its normal position, while absolute positioning places an element relative to its closest positioned ancestor.

Can an element have both relative and absolute positioning?

Yes, an element can have relative positioning while its child elements can use absolute positioning, making the child elements move relative to the parent.

How does fixed positioning differ from absolute positioning?

Fixed positioning keeps an element fixed in place relative to the browser window, even when scrolling, while absolute positioning scrolls with the page.

Does relative positioning affect the layout of other elements?

No, relative positioning does not affect the layout of other elements, as the element still occupies its original space.

Is fixed positioning supported in all browsers?

Yes, fixed positioning is supported in all modern browsers, but older versions of Internet Explorer may have partial support.

Dive deep into the world of JavaScript with our [JavaScript Full Course Online](#), offering everything from the basics to advanced techniques. Learn how to build interactive websites, optimize your code, and earn a certification to showcase your skills.

Take the **Three 90 Challenge**! Complete **90% of the course in 90 days**, and **earn a 90% refund**. Push your limits and stay motivated as you work towards mastering JavaScript. Join now and get started on your **JavaScript learning journey!**

[Comment](#)[More info](#)[Advertise with us](#)[Next Article](#)[CSS bottom Property](#)

Similar Reads

CSS Tutorial

CSS stands for Cascading Style Sheets. It is a stylesheet language used to style and enhance website presentation. CSS is one of the main three components of a webpage along with HTML and JavaScript. HTML adds Structure to a Webpage, JavaScript adds logic to it and CSS makes it visually appealing.

7 min read

CSS Introduction

CSS (Cascading Style Sheets) is a language designed to simplify the process of making web pages presentable. It allows you to apply styles to HTML documents by prescribing colors, fonts, spacing, and positioning. The main advantages are separation of content (in HTML) and styling (in CSS) and the same.

5 min read

CSS Syntax

CSS (Cascading Style Sheets) is a stylesheet language used to describe the presentation of a document written in HTML. Understanding CSS syntax is fundamental for creating visually appealing and well-structured web pages. Basic CSS Syntax CSS is written as rulesets. A ruleset consists of a selector and a declaration.

6 min read

CSS Selectors

CSS selectors are used to target HTML elements on your pages, allowing you to apply styles based on their ID, class, type attributes, and more. There are mainly 5 types of selectors. Basic CSS Selectors: These are used to target elements by tag, .class, or #id for fundamental styling needs. Combinators

9 min read

CSS Comments

CSS comments are used to add notes or explanations to your code, helping you and others understand it better. They start with `/*` and end with `*/` and can be used for both single-line and multi-line comments. Note: Comments are ignored by browsers, so they won't affect how your webpage looks or works.

2 min read

CSS Colors

CSS colors are used to set the color of different parts of a webpage, like text, background, and borders. This helps make the page look more attractive and easier to read. You can define colors using names, hex codes, RGB values, and more. You can try different formats of colors here- [#content-iframe](#)

6 min read

CSS Borders

Borders in CSS are used to create a visible outline around an element. They can be customized in terms of Width: The thickness of the border.Style: The appearance of the border (solid, dashed, dotted, etc.).Color: The color of the border.You can try different types of borders here- [#custom-iframe](#) h

6 min read

CSS Margins

CSS margins are used to create space around an element, separating it from neighboring elements and the edges of the webpage. They control the layout by adjusting the distance between elements, providing better organization and readability. Syntax:body { margin: value;}[GFGTABS] HTML <!--Driver C

4 min read

CSS Height and Width

Height and Width in CSS are used to set the height and width of boxes. Their values can be set using length, percentage, or auto. Width and HeightThe width and height properties in CSS are used to define the dimensions of an element. The values can be set in various units, such as pixels (px), centi

4 min read

CSS Outline

CSS outline is a property used to draw a line around an element's border. It does not affect the layout, unlike borders. It's often used to highlight elements, providing a visual emphasis without altering the dimensions of the element. Syntaxselector{ outline: outline-width outline-type outline-colo

4 min read

CSS Fonts

CSS fonts control how text appears on a webpage. With CSS, you can specify various properties like font family, size, weight, style, and line height to create visually appealing and readable typography [GFGTABS] HTML <!--Driver Code Starts--> <html> <head> <!--Driver Code Ends

5 min read

CSS Icons

CSS Icons are an essential part of modern web design, offering a visual representation of actions, ideas, and objects. CSS icons offer a compelling way to achieve this by seamlessly integrating scalable and customizable icons into your web design. CSS Icons from various libraries can be effortlessly

5 min read

CSS Links

A link is a connection from one web page to another web page. CSS property can be used to style the links in various different ways.States of Link: Before discussing CSS properties, it is important to know the states of a link. Links can exist in different states and they can be styled using pseudo-

5 min read

CSS Lists

CSS Lists are used to display items in a clear and organized manner, either with bullets (unordered) or numbers (ordered). They help keep content neat and structured on a webpage. With CSS, you can customize the look of lists to improve the design and layout of your content. Try It: .custom-item { b

5 min read

CSS Tables

Tables in CSS are used to style HTML table elements, allowing data to be presented in a structured, organized format with rows and columns. CSS provides a variety of properties that can be applied to tables to enhance their appearance and functionality. CSS Table Properties1. BorderThe border proper

6 min read

CSS Display Property

The CSS display property specifies an element's display behaviour (the type of rendering box). It defines how an element is rendered in the layout, determining its positioning and interaction within the document's flow and structure. Syntax display: value;Try CSS Display Property #iframe{ height:550

5 min read

CSS max-width Property

The max-width property in CSS defines the maximum width an element can occupy, ensuring that it doesn't exceed a specified width. This property plays a crucial role in responsive design, allowing the content to adjust within a defined limit. It helps in creating flexible layouts that look great on v

4 min read

CSS Positioning Elements

CSS positioning defines how elements are placed within a web page. It allows you to control the layout, stacking order, and alignment of elements. The primary positioning types in CSS are: Position PropertyDescriptionFixedAn element with position: fixed property remains in the same position relative

4 min read

CSS - Table Properties

CSS z-index Property

CSS z-index is used to control the stacking order of overlapping elements, which decides whether an element appears on top or behind others based on their assigned value. It works only on positioned elements (relative, absolute, or fixed). Default stacking order applies if no z-index is defined. The z

4 min read

CSS Overflow

The CSS overflow controls the big content. It tells whether to clip content or to add scroll bars. Try It: .custom-item { border: 1px solid gray; font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif; border-radius: 3px; margin: auto; margin-top: 5px; max-width: 320px; min-wi

4 min read

CSS Float

The CSS float property is used to move an element out of the normal document flow and position it to the left or right of its container. For example, float: left moves the element to the left, and float: right moves it to the right. Other content will wrap around the floated element which helps to c

3 min read

CSS Align

CSS alignment techniques are essential for positioning items and distributing space between content. These techniques include horizontal and vertical alignment using various properties such as margin: auto, position: absolute, text-align, and padding. This article covers multiple methods to achieve

4 min read

CSS Combinators

CSS combinators define the relationship between two selectors. CSS selectors are patterns used to select elements for styling. A CSS selector can be simple or complex, consisting of more than one selector connected using combinators. Understanding these combinators is essential for precise and effic

4 min read

CSS Pseudo-classes

A pseudo-class is a keyword added to a CSS selector, prefixed by a colon (:), to define a specific state or condition of an element. It is used to style elements like a hovered button, the first child of a container, or checked input fields. Syntax selector:pseudo-class { /* styles */}Interactive/Us

11 min read

CSS Pseudo Elements

A pseudo-element is a keyword added to a selector that lets you style specific parts of an element. For example, you can style the first line of a paragraph, add content before or after an element, or create complex effects with minimal code. Pseudo-elements are denoted by a double colon (::) (or :

5 min read

CSS Opacity / Transparency

The opacity in CSS is the property of an element that describes the transparency of the element. It is the opposite of transparency & represents the degree to which the content will be hidden behind an element. Try It: .custom-item { border: 1px solid gray; font-family: 'Gill Sans', 'Gill Sans M

6 min read

CSS DropDowns

Dropdown menus are essential for interactive websites, allowing users to access multiple links from a single menu. Using CSS, you can design stylish and functional dropdown menus that enhance the user experience. This article will cover various types of dropdowns and how to implement them using CSS.

6 min read

CSS Image Gallery

Creating a responsive image gallery is a great way to showcase a collection of pictures on your website. In this article, we'll walk you through the steps to build a responsive image gallery using HTML and CSS. This guide will help you create a beautiful gallery that looks great on all devices. Step

4 min read

CSS Image Sprites

CSS image sprites are a single image that combines multiple images, used to reduce page loading time and bandwidth consumption. Why Use Image Sprites?Image sprites are used for two main reasons: Faster Page Loading: By using a single image, page loading time is significantly reduced.Reduced Bandwidt

3 min read

CSS Attribute Selector

CSS attribute Selector allows you to select elements based on the presence, value, or specific characteristics of their attributes. They are particularly useful for dynamic or structured content where attributes play a key role, such as in forms or data tables. Types of CSS Attribute Selectors1. [at

5 min read

CSS Counters

CSS counters allow you to number elements like lists or sections automatically. They are "variables" maintained by CSS, and their values can be incremented with CSS rules, tracking how many times they are used. To work with CSS counters, we use a set of properties: counter-reset: Creates or resets a

3 min read

CSS Website Layout

CSS website layout divides a webpage into multiple sections like header, navigation bar, content area, and footer, making it easier to organize content and build the site. 1. Header SectionThe header is the top section of a webpage, typically containing the website's name or logo. [GFGTABS] html

4 min read

CSS Units

CSS Units

CSS units define the size of elements, with absolute units (like px, cm) having fixed values and relative units (like em, rem, %, vh) depending on factors like the viewport or parent elements. There are two types of units Absolute and Relative units. Absolute unitsAbsolute units in CSS, such as px,

9 min read

CSS Specificity

CSS Specificity is a fundamental concept in CSS that determines the order of style application. It is calculated based on the types of selectors used, including inline styles, IDs, classes, attributes, and element types. Understanding CSS Specificity is important for: Avoiding styling conflictsEnsuring

3 min read

How to apply !important in CSS?

The !important rule in CSS is used to add more importance to a property/value than normal. It forces a style to override any other declarations, ensuring the specified property value is applied, regardless of specificity. It helps resolve conflicts but should be used sparingly to avoid complicating

2 min read

CSS Text Formatting

CSS Backgrounds

CSS Lists

CSS Box model

CSS Positioning Elements

CSS Positioning Elements

CSS positioning defines how elements are placed within a web page. It allows you to control the layout, stacking order, and alignment of elements. The primary positioning types in CSS are: Position PropertyDescriptionFixedAn element with position: fixed property remains in the same position relative

4 min read

Relative vs Absolute vs Fixed Position in CSS

CSS positioning is a fundamental concept in web design and development that allows precise control over how elements are arranged on a webpage. There are three main types of CSS positioning: relative, absolute, and fixed positioning. What is Relative Positioning?Relative Positioning is a CSS techniq

4 min read

CSS bottom Property

The CSS bottom property affects the vertical position of a positioned element. This property does not affect non-positioned elements. It controls an element's vertical distance from the bottom edge of its containing element using units like pixels, percentages, or keywords (auto, initial, inherit).

4 min read

CSS flexbox

CSS Grid Layout

Responsive Web Design

CSS Transitions

CSS Animations

CSS Exercises, Practice Questions and Solutions

Are you eager to master CSS or looking to sharpen your skills? Dive into our CSS Exercises, Practice Questions, and Solutions designed for both beginners and experienced developers. With our interactive platform, you can engage in hands-on coding challenges, monitor your progress, and enhance your w

3 min read



Corporate & Communications Address:

A-143, 7th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305)

Registered Address:

K 061, Tower K, Gulshan Vivante
Apartment, Sector 137, Noida, Gautam
Buddh Nagar, Uttar Pradesh, 201305

Advertise with us

Company

About Us
Legal
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
GeeksforGeeks Community

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Tecnologías web

HTML
CSS
JavaScript
Mecanografiado
ReactJS
SiguienteJS
Oreja
Diseño web

Ciencias de la Computación

Sistemas operativos
Red de computadoras
Sistema de gestión de bases de datos
Ingeniería de software
Diseño de lógica digital
Matemáticas de ingeniería
Desarrollo de software
Pruebas de software

Diseño de sistemas

Diseño de alto nivel
Diseño de bajo nivel
Diagramas UML
Guía de entrevistas
Patrones de diseño
OAD (Oportunidad de aprendizaje)
Campamento de entrenamiento sobre diseño de sistemas
Preguntas de la entrevista

Asignaturas escolares

Matemáticas
Física
Química
Biología
Ciencia social
Gramática inglesa
Comercio
GK mundial

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

Data Science & ML

Data Science With Python
Ciencia de datos para principiantes
Aprendizaje automático
Matemáticas ML
Visualización de datos
Pandas
NumPy
PNL
Aprendizaje profundo

Tutorial de Python

Ejemplos de programación en Python
Proyectos de Python
Intercambio de Python
Extracción de datos web
Tutorial de OpenCV
Pregunta de entrevista sobre Python
Django

DevOps

Git
Linux
AWS
Estibador
Kubernetes
Azur
PCG
Hoja de ruta de DevOps

Preparación para la entrevista

Programación competitiva
DS superior o Algo para CP
Proceso de selección por empresa
Preparación por empresa
Preparación de aptitud
Rompecabezas

Vídeos de GeeksforGeeks

DSA
Pitón
Java
C++
Desarrollo web
Ciencia de datos
Asignaturas de informática