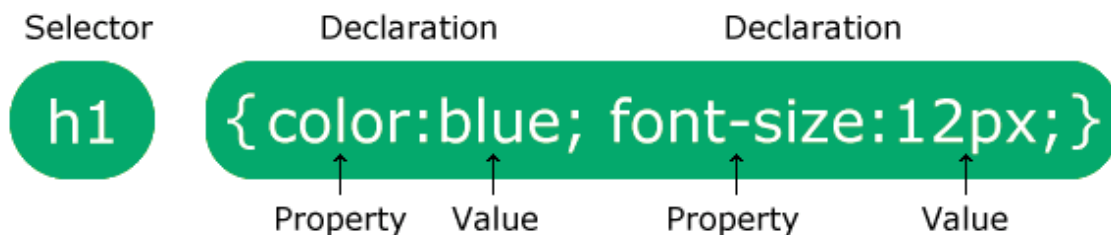


## Introducción a css

### ¿Qué es CSS?

CSS son las siglas de Cascading Style Sheets, describe cómo se mostrarán los elementos HTML en la pantalla, el papel o en otros medios. CSS ahorra mucho trabajo. Puede controlar el diseño de varias páginas web a la vez y las hojas de estilo externas se almacenan en archivos **.css**.

### Sintaxis CSS



El selector apunta al elemento HTML al que desea aplicar estilo. El bloque de declaración contiene una o más declaraciones separadas por punto y coma. Cada declaración incluye un nombre de propiedad CSS y un valor, separados por dos puntos. Varias declaraciones CSS se separan con punto y coma y los bloques de declaración están rodeados por llaves.

### Ejemplo

En este ejemplo, todos los elementos `<p>` estarán alineados al centro, con un color de texto rojo:

```
p {  
  color: red;  
  text-align: center;  
}
```

### Ejemplo explicado

- **p** es un selector en CSS (apunta al elemento HTML que desea aplicar estilo: `<p>`).
- **color** es una propiedad y **red** el valor de la propiedad.
- **text-align** es una propiedad y **center** es el valor de la propiedad.

### Selectores CSS

Un selector de CSS selecciona los elementos HTML a los que desea aplicar estilo.

## Selectores

Los selectores CSS se utilizan para "buscar" (o seleccionar) los elementos HTML que desea aplicar estilo. Podemos dividir los selectores de CSS en cinco categorías:

- Selectores simples (seleccione elementos según el nombre, la identificación, la clase).

### *El selector de elementos CSS*

El selector de elementos selecciona elementos HTML en función del nombre del elemento.

#### Ejemplo

Aquí, todos los elementos **<p>** de la página estarán alineados al centro, con un color de texto **rojo**:

```
p {  
  color: red;  
  text-align: center;  
}
```

### *El selector de id de CSS*

El selector de id utiliza el atributo **id** de un elemento HTML para seleccionar un elemento específico. La identificación de un elemento es única dentro de una página, por lo que el selector de identificación se utiliza para seleccionar un elemento único. Para seleccionar un elemento con una identificación específica, escriba un carácter de almohadilla (#), seguido de la identificación del elemento.

#### Ejemplo

La siguiente regla CSS se aplicará al elemento HTML con **id = "para1"**:

```
#para1 {  
  text-align: center;  
  color: red;  
}
```

**Nota:** ¡Un nombre de identificación no puede comenzar con un número!

## El selector de clases CSS

El selector de clases selecciona elementos HTML con un atributo de clase específico. Para seleccionar elementos con una clase específica, escriba un punto (.), seguido del nombre de la clase.

## Ejemplo

En este ejemplo, todos los elementos HTML con **class = "center"** serán **rojos** y alineados al **centro**:

```
.center {  
    text-align: center;  
    color: red;  
}
```

También puede especificar que solo los elementos HTML específicos deben verse afectados por una clase.

## Ejemplo

En este ejemplo, solo los elementos `<p>` con `class = "center"` serán rojos y alineados al centro:

```
p.center {  
    text-align: center;  
    color: red;  
}
```

Los elementos HTML también pueden hacer referencia a más de una clase.

## Ejemplo

En este ejemplo, el elemento `<p>` se diseñará de acuerdo con **class = "center"** y **class = "large"**:

```
<p class="center large">This paragraph refers to two classes.</p>
```

**Nota:** ¡El nombre de una clase no puede comenzar con un número!

## El selector universal de CSS

El selector universal (\*) selecciona todos los elementos HTML de la página.

## Ejemplo

La siguiente regla CSS afectará a todos los elementos HTML de la página:

```
* {  
    text-align: center;  
    color: blue;  
}
```

## El selector de agrupación CSS

El selector de agrupación selecciona todos los elementos HTML con las mismas definiciones de estilo.

Mire el siguiente código CSS (los elementos **h1**, **h2** y **p** tienen las mismas definiciones de estilo):

```
h1 {  
    text-align: center;  
    color: red;  
}  
  
h2 {  
    text-align: center;  
    color: red;  
}  
  
p {  
    text-align: center;  
    color: red;  
}
```

Será mejor agrupar los selectores para minimizar el código. Para agrupar selectores, separe cada selector con una coma.

### Ejemplo

En este ejemplo, hemos agrupado los selectores del código anterior:

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

## Cómo agregar o enlazar CSS con nuestro documento HTML

### *Tres formas de insertar CSS*

Hay tres formas de insertar una hoja de estilo:

- CSS externo
- CSS interno

- CSS en línea

## *CSS externo*

Con una hoja de estilo externa, puede cambiar el aspecto de un sitio web completo cambiando solo un archivo. Cada página HTML debe incluir una referencia al archivo de hoja de estilo externo dentro del elemento **<link>**, dentro de la sección del encabezado.

## Ejemplo

Los estilos externos se definen dentro del elemento **<link>**, dentro de la sección **<head>** de una página HTML:

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Una hoja de estilo externa se puede escribir en cualquier editor de texto y debe guardarse con una extensión **.css**. El archivo **.css** externo no debe contener etiquetas HTML.

Así es como se ve el archivo "mystyle.css":

## "mystyle.css"

```
body {
  background-color: lightblue;
}

h1 {
  color: navy;
  margin-left: 20px;
}
```

Nota: No agregue un espacio entre el valor de la propiedad y la unidad (como **margin-left: 20 px;**). La forma correcta es: **margin-left: 20px;**

## *CSS interno*

Se puede utilizar una hoja de estilo interna si una sola página HTML tiene un estilo único. El estilo interno se define dentro del elemento **<style>**, dentro de la sección de cabecera.

## **Ejemplo**

Los estilos internos se definen dentro del elemento **<style>**, dentro de la sección **<head>** de una página HTML:

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
    background-color: linen;
}

h1 {
    color: maroon;
    margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

## *CSS en línea*

Se puede usar un estilo en línea para aplicar un estilo único a un solo elemento. Para usar estilos en línea, agregue el atributo de estilo al elemento relevante. El atributo de estilo puede contener cualquier propiedad CSS.

## Ejemplo

Los estilos en línea se definen dentro del atributo "**style**" del elemento relevante:

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

<h1 style="color:blue;text-align:center;">This is a heading</h1>
<p style="color:red;">This is a paragraph.</p>

</body>
</html>
```

**Consejo:** un estilo en línea pierde muchas de las ventajas de una hoja de estilo (al mezclar contenido con presentación). Utilice este método con moderación.

## ¿Cómo añadir color con CSS?

Los colores se especifican utilizando nombres de colores predefinidos o valores RGB, HEX, HSL, RGBA, HSLA.

### *Nombres de colores CSS*

En CSS, se puede especificar un color mediante el uso de un nombre de color predefinido: Tomato, Orange, DodgerBlue, MediumSeaGreen, Gray, SlateBlue, Violet, LightGray, etc.

### *Color de fondo CSS*

Puede establecer el color de fondo de los elementos HTML:

## Ejemplo

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>
<p style="background-color:Tomato;">Lorem ipsum...</p>
```

### *Color de texto CSS*

Puede establecer el color del texto:

## Ejemplo

```
<h1 style="color:Tomato;">Hello World</h1>
<p style="color:DodgerBlue;">Lorem ipsum...</p>
<p style="color:MediumSeaGreen;">Ut wisi enim...</p>
```

---

## Color del borde CSS

Puede establecer el color de los bordes:

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

## Valores de color CSS

En CSS, los colores también se pueden especificar usando valores RGB, valores HEX, valores HSL, valores RGBA y valores HSLA:

Igual que el nombre del color "Tomate":

**rgb (255, 99, 71)**

**# ff6347**

**hsl (9, 100%, 64%)**

Igual que el nombre del color "Tomate", pero 50% transparente:

**rgba (255, 99, 71, 0,5)**

**hsla (9, 100%, 64%, 0,5)**

## Ejemplo

```
<h1 style="background-color:rgb(255, 99, 71);">...</h1>
<h1 style="background-color:#ff6347;">...</h1>
<h1 style="background-color:hsl(9, 100%, 64);">...</h1>

<h1 style="background-color:rgba(255, 99, 71, 0.5);">...</h1>
<h1 style="background-color:hsla(9, 100%, 64%, 0.5);">...</h1>
```

## Color de fondo CSS

La propiedad **background-color** especifica el color de fondo de un elemento.



## Ejemplo

El color de fondo de una página se establece así:

```
body {  
    background-color: lightblue;  
}
```

## Ejemplo

Aquí, los elementos `<h1>`, `<p>` y `<div>` tendrán diferentes colores de fondo:

```
h1 {  
    background-color: green;  
}  
  
div {  
    background-color: lightblue;  
}  
  
p {  
    background-color: yellow;  
}
```

## *Opacidad / Transparencia*

La propiedad **opacity** especifica la opacidad / transparencia de un elemento. Puede tomar un valor de 0,0 a 1,0. Cuanto menor sea el valor, más transparente:

## Ejemplo

```
div {  
    background-color: green;  
    opacity: 0.3;  
}
```

**Nota:** Cuando se usa la propiedad **opacity** para agregar transparencia al fondo de un elemento, todos sus elementos secundarios heredan la misma transparencia. Esto puede hacer que el texto dentro de un elemento completamente transparente sea difícil de leer.

Imagen de fondo CSS

La propiedad **background-image** especifica una imagen para usar como fondo de un elemento.

De forma predeterminada, la imagen se repite para que cubra todo el elemento.

## Ejemplo

Establecer la imagen de fondo para una página:

```
body {  
    background-image: url("paper.gif");  
}
```

## Bordes CSS

Las propiedades del borde CSS le permiten especificar el estilo, el ancho y el color del borde de un elemento.

### Estilo de borde CSS

La propiedad **border-style** especifica qué tipo de borde mostrar:

## Ejemplo

Demostración de los diferentes estilos de bordes:

```
p.dotted {border-style: dotted;}  
p.dashed {border-style: dashed;}  
p.solid {border-style: solid;}  
p.double {border-style: double;}  
p.groove {border-style: groove;}  
p.ridge {border-style: ridge;}  
p.inset {border-style: inset;}  
p.outset {border-style: outset;}  
p.none {border-style: none;}  
p.hidden {border-style: hidden;}  
p.mix {border-style: dotted dashed solid double;}
```

## Márgenes CSS

Los márgenes se utilizan para crear espacio alrededor de los elementos, fuera de los bordes definidos.

Con CSS, tienes control total sobre los márgenes. Existen propiedades para establecer el margen para cada lado de un elemento (superior, derecha, inferior e izquierda).

## *Margen: lados individuales*

CSS tiene propiedades para especificar el margen para cada lado de un elemento:

- margin-top
- margin-right
- margin-bottom
- margin-left

## Ejemplo

Establezca diferentes márgenes para los cuatro lados de un elemento **<p>**:

```
p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}
```

## *Margen - Propiedad de taquigrafía*

Para acortar el código, es posible especificar todas las propiedades de los márgenes en una propiedad.

La propiedad margin es una propiedad abreviada para las siguientes propiedades de margen individuales:

- margin-top
- margin-right
- margin-bottom
- margin-left

Pues así es como funciona:

Si la propiedad margin tiene cuatro valores:

**margin: 25px 50px 75px 100px;**

- El margen superior es de 25 px.
- El margen derecho es 50px.

- El margen inferior es de 75 px.
- El margen izquierdo es 100px.

### Ejemplo

Utilice la propiedad abreviada de margen con cuatro valores:

```
p {  
  margin: 25px 50px 75px 100px;  
}
```

Si la propiedad margin tiene tres valores:

**margin: 25px 50px 75px;**

- El margen superior es de 25 px.
- Los márgenes derecho e izquierdo son 50px.
- El margen inferior es de 75 px.

### Ejemplo

Utilice la propiedad abreviada de margen con tres valores:

```
p {  
  margin: 25px 50px 75px;  
}
```

Si la propiedad margin tiene dos valores:

**margin: 25px 50px;**

- Los márgenes superior e inferior son de 25 px.
- Los márgenes derecho e izquierdo son 50px.

### Ejemplo

Utilice la propiedad abreviada de margen con dos valores:

```
p {  
  margin: 25px 50px;  
}
```

## Relleno CSS

Las propiedades padding CSS se utilizan para generar espacio alrededor del contenido de un elemento, dentro de los bordes definidos.

Con CSS, tienes control total sobre el relleno. Existen propiedades para configurar el relleno para cada lado de un elemento (superior, derecha, inferior e izquierda).

### *Relleno - Lados individuales*

CSS tiene propiedades para especificar el relleno para cada lado de un elemento:

- `padding-top`
- `padding-right`
- `padding-bottom`
- `padding-left`

### Ejemplo

Establezca un relleno diferente para los cuatro lados de un elemento `<div>`:

```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}
```

### *Padding - Propiedad de taquigrafía*

Para acortar el código, es posible especificar todas las propiedades de relleno en una propiedad.

### Ejemplo

Utilice la propiedad abreviada de relleno con cuatro valores:

```
div {  
  padding: 25px 50px 75px 100px;  
}
```

**relleno: 25px 50px 75px;**

- el relleno superior es de 25 px
- los rellenos derecho e izquierdo son 50px

- el acolchado inferior es de 75 px

## Ejemplo

Utilice la propiedad abreviada de relleno con tres valores:

```
div {  
  padding: 25px 50px 75px;  
}
```

**relleno: 25px 50px;**

- los acolchados superior e inferior son de 25 px
- los rellenos derecho e izquierdo son 50px

## Ejemplo

Utilice la propiedad abreviada de relleno con dos valores:

```
div {  
  padding: 25px 50px;  
}
```