

# Introducción a CSS para Principiantes

---

## ¿Qué es CSS?

CSS (Cascading Style Sheets o Hojas de Estilo en Cascada) es un lenguaje que determina cómo se presentan visualmente los elementos HTML en una página web. Si HTML es el esqueleto de una página web, CSS es el vestuario y maquillaje.

**Propósito principal:** Separar el contenido (HTML) de la presentación (CSS), permitiendo:

- Mayor control sobre la apariencia de los sitios web
- Cambios globales de estilo modificando un solo archivo
- Consistencia visual a través de múltiples páginas
- Adaptabilidad a diferentes dispositivos y tamaños de pantalla

**Historia breve:** CSS fue propuesto por primera vez en 1994 por Håkon Wium Lie mientras trabajaba en el CERN. La primera especificación oficial (CSS1) fue publicada en 1996, y desde entonces ha evolucionado constantemente, siendo CSS3 la versión más reciente.

## Formas de incluir CSS

Hay tres métodos para aplicar estilos CSS a un documento HTML:

### 1. CSS Inline

Se aplica directamente a un elemento HTML mediante el atributo `style`.

**Ejemplo:**

```
<p style="color: blue; font-size: 16px;">
  Este es un párrafo con estilo inline.
</p>
```

**Ventajas:**

- Aplicación inmediata y directa
- No requiere archivos adicionales

**Desventajas:**

- No es reutilizable
- Difícil de mantener cuando el sitio crece
- Mezcla contenido y presentación

### 2. CSS Interno (o Embedded)

Se define dentro del documento HTML utilizando la etiqueta `<style>` en la sección `<head>`.

### Ejemplo:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mi página web</title>
    <style>
      p {
        color: blue;
        font-size: 16px;
      }
      h1 {
        color: red;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <h1>Título principal</h1>
    <p>Este párrafo estará en color azul.</p>
  </body>
</html>
```

### Ventajas:

- Estilos aplicables a múltiples elementos en la misma página
- No requiere archivos externos

### Desventajas:

- No es reutilizable en otras páginas del sitio
- Puede aumentar el tiempo de carga de la página

## 3. CSS Externo (recomendado)

Los estilos se definen en un archivo separado con extensión **.css** que se vincula al documento HTML.

### Ejemplo de archivo **estilos.css**:

```
p {
  color: blue;
  font-size: 16px;
}

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

## Vinculación en el HTML:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mi página web</title>
    <link rel="stylesheet" href="estilos.css" />
  </head>
  <body>
    <h1>Título principal</h1>
    <p>Este párrafo estará en color azul.</p>
  </body>
</html>
```

## Ventajas:

- Separación completa entre contenido y presentación
- Reutilizable en múltiples páginas
- Facilita el mantenimiento del sitio
- El navegador puede almacenar en caché el archivo CSS
- Ideal para proyectos de cualquier tamaño

## Selectores CSS

Los selectores determinan qué elementos HTML serán afectados por las reglas de estilo. Son la base para aplicar estilos específicos a elementos concretos.

### Selectores básicos

#### Selector de elemento

Selecciona todos los elementos de un tipo específico.

```
p {
  color: blue;
}
```

Este código hará que todos los párrafos (<p>) sean de color azul.

#### Selector de clase

Selecciona elementos que tienen un atributo `class` específico. Se usa con un punto (.) seguido del nombre de la clase.

```
.destacado {
  font-weight: bold;
}
```

```
background-color: yellow;
}
```

```
<p class="destacado">Este párrafo estará en negrita con fondo amarillo.
</p>
```

## Selector de ID

Selecciona un elemento con un atributo `id` específico. Se usa con un signo de numeral (`#`) seguido del ID.

```
#encabezado-principal {
  font-size: 32px;
  text-align: center;
}
```

```
<h1 id="encabezado-principal">Este es el título principal</h1>
```

## Selectores combinados

### Selector descendiente

Selecciona elementos que son descendientes de otro elemento.

```
article p {
  text-indent: 20px;
}
```

Este código aplica una sangría a todos los párrafos que están dentro de un `<article>`.

### Selector hijo directo

Selecciona elementos que son hijos directos de otro elemento.

```
ul > li {
  list-style-type: square;
}
```

Este código cambia el marcador de lista a cuadrado, pero solo para los `<li>` que son hijos directos de un `<ul>`.

## Selectores de atributo

Seleccionan elementos basados en la presencia o valor de un atributo.

```
input[type="text"] {  
  border: 1px solid blue;  
}
```

Este código aplica un borde azul a todos los campos de entrada de texto.

## Pseudo-clases

Seleccionan elementos en un estado específico.

```
a:hover {  
  color: red;  
}  
  
li:first-child {  
  font-weight: bold;  
}
```

El primer ejemplo cambia el color de los enlaces al pasar el cursor sobre ellos. El segundo ejemplo pone en negrita el primer elemento de cada lista.

## Propiedades de estilo fundamentales

### Colores

CSS ofrece varias formas de especificar colores:

#### Nombres de colores

CSS tiene 140 nombres de colores predefinidos como **red**, **blue**, **green**, **purple**, etc.

```
h1 {  
  color: navy;  
  background-color: lightgray;  
}
```

#### Valores hexadecimales

Un código de 6 dígitos que representa los componentes RGB (rojo, verde, azul).

```
p {
  color: #ff0000; /* Rojo */
  background-color: #ffffff; /* Blanco */
}
```

También se puede usar la forma abreviada de 3 dígitos cuando los pares son iguales:

```
p {
  color: #f00; /* Equivalente a #FF0000 (rojo) */
}
```

## Valores RGB

Define colores según sus componentes rojo, verde y azul (valores de 0 a 255).

```
span {
  color: rgb(255, 0, 0); /* Rojo */
  background-color: rgb(200, 200, 200); /* Gris claro */
}
```

## Valores RGBA

Similar a RGB, pero con un cuarto valor para la transparencia (alpha), que va de 0 (completamente transparente) a 1 (completamente opaco).

```
div {
  background-color: rgba(255, 0, 0, 0.5); /* Rojo semi-transparente */
}
```

## Propiedades de color más comunes

- **color**: Color del texto
- **background-color**: Color de fondo
- **border-color**: Color del borde

## Ejemplo completo:

```
.caja {
  color: #333333;
  background-color: rgb(240, 240, 240);
  border: 2px solid rgba(0, 0, 255, 0.5);
}
```

---

## Tipografía

Las propiedades tipográficas controlan la apariencia del texto.

### Familia de fuente (font-family)

Define qué fuente se utilizará para mostrar el texto.

```
body {  
  font-family: Arial, Helvetica, sans-serif;  
}  
  
h1 {  
  font-family: "Times New Roman", Times, serif;  
}
```

Se recomienda especificar varias fuentes como respaldo, en caso de que el navegador no tenga la primera opción.

### Tamaño de fuente (font-size)

Controla el tamaño del texto. Puede especificarse en varias unidades:

```
p {  
  font-size: 16px; /* Píxeles */  
}  
  
h1 {  
  font-size: 2em; /* Relativo al tamaño del contenedor padre */  
}  
  
h2 {  
  font-size: 150%; /* Porcentaje del tamaño heredado */  
}
```

### Peso de fuente (font-weight)

Determina el grosor del texto.

```
p {  
  font-weight: normal; /* Peso normal */  
}  
  
strong {  
  font-weight: bold; /* Negrita */  
}
```

```
h1 {  
  font-weight: 700; /* Valores numéricos de 100 a 900 */  
}
```

## Estilo de fuente (font-style)

Controla si el texto se muestra en cursiva o normal.

```
p {  
  font-style: normal;  
}  
  
em {  
  font-style: italic;  
}
```

## Decoración de texto (text-decoration)

Añade líneas al texto (subrayado, tachado, etc.).

```
a {  
  text-decoration: none; /* Elimina el subrayado predeterminado */  
}  
  
h2 {  
  text-decoration: underline; /* Subrayado */  
}  
  
.precio-antiguo {  
  text-decoration: line-through; /* Tachado */  
}
```

## Alineación de texto (text-align)

Controla la alineación horizontal del texto.

```
p {  
  text-align: left; /* Izquierda (predeterminado) */  
}  
  
h1 {  
  text-align: center; /* Centrado */  
}  
  
.justificado {
```



```
    text-align: justify; /* Texto justificado */
}

.derecha {
    text-align: right; /* Derecha */
}
```

## Transformación de texto (text-transform)

Cambia las mayúsculas/minúsculas del texto.

```
.titulo {
    text-transform: uppercase; /* TODAS MAYÚSCULAS */
}

.subtitulo {
    text-transform: capitalize; /* Primera Letra En Mayúscula */
}

.texto-discreto {
    text-transform: lowercase; /* todas minúsculas */
}
```

## Espaciado

El espaciado controla la distancia entre elementos y dentro de ellos.

## Margen (margin)

Espacio fuera del elemento, entre el elemento y los elementos adyacentes.

```
p {
    margin: 10px; /* Margen de 10px en todos los lados */
}

h1 {
    margin-top: 20px; /* Solo margen superior */
    margin-right: 15px; /* Solo margen derecho */
    margin-bottom: 20px; /* Solo margen inferior */
    margin-left: 15px; /* Solo margen izquierdo */
}

div {
    /* Orden: arriba, derecha, abajo, izquierda */
    margin: 10px 15px 20px 25px;
}

section {
    /* Orden: arriba/abajo, izquierda/derecha */
}
```

```
margin: 10px 20px;
}
```

## Relleno (padding)

Espacio dentro del elemento, entre el contenido y el borde.

```
button {
  padding: 10px; /* Relleno de 10px en todos los lados */
}

.caja {
  padding-top: 20px;
  padding-right: 15px;
  padding-bottom: 20px;
  padding-left: 15px;
}

article {
  /* Sintaxis abreviada similar a margin */
  padding: 10px 15px 20px 25px; /* arriba, derecha, abajo, izquierda */
}
```

## Altura y anchura (height y width)

Define las dimensiones del elemento.

```
.cuadrado {
  width: 100px;
  height: 100px;
  background-color: red;
}

.contenedor {
  width: 80%; /* Porcentaje del ancho del contenedor padre */
  height: 300px;
}
```

## Altura de línea (line-height)

Controla el espacio vertical entre líneas de texto.

```
p {
  line-height: 1.5; /* 1.5 veces el tamaño de la fuente */
}
```

```
.texto-apretado {  
  line-height: 1; /* Sin espacio adicional */  
}  
  
.texto-espaciado {  
  line-height: 2; /* Doble espacio */  
}
```

## Espacio entre letras y palabras

Controla el espaciado horizontal del texto.

```
h1 {  
  letter-spacing: 2px; /* Espacio entre letras */  
}  
  
p {  
  word-spacing: 5px; /* Espacio entre palabras */  
}
```

## Bordes

Los bordes definen el contorno de un elemento.

### Propiedades básicas de borde

```
.caja {  
  border-width: 2px; /* Grosor del borde */  
  border-style: solid; /* Estilo del borde */  
  border-color: black; /* Color del borde */  
}  
  
/* Forma abreviada (ancho, estilo, color) */  
.caja-simple {  
  border: 2px solid black;  
}
```

### Estilos de borde

Existen varios estilos disponibles:

```
.borde-solido {  
  border: 2px solid black;  
}  
  
.borde-punteado {
```

```

    border: 2px dotted red;
}

.borde-discontinuo {
    border: 2px dashed blue;
}

.borde-doble {
    border: 4px double green;
}

.borde-acanalado {
    border: 5px groove gray;
}

.borde-cresta {
    border: 5px ridge orange;
}

.borde-hundido {
    border: 5px inset purple;
}

.borde-elevado {
    border: 5px outset brown;
}

```

## Bordes por lado

Se pueden definir bordes diferentes para cada lado del elemento:

```

.personalizado {
    border-top: 2px solid red;
    border-right: 3px dotted blue;
    border-bottom: 4px dashed green;
    border-left: 5px double black;
}

```

## Radio de borde (border-radius)

Crea esquinas redondeadas:

```

.esquinas-redondeadas {
    border: 2px solid black;
    border-radius: 10px; /* Todas las esquinas */
}

.esquinas-personalizadas {
    /* Orden: superior-izquierda, superior-derecha, inferior-derecha,

```

```

inferior-izquierda */
border-radius: 5px 10px 15px 20px;
}

.circulo {
width: 100px;
height: 100px;
border-radius: 50%; /* Crea un círculo */
background-color: blue;
}

```

## Ejercicios Prácticos

### Ejercicio 1: Personaliza un párrafo

Crea un párrafo con fondo azul claro, texto en negrita, bordes redondeados y algo de espacio interior.

```

<!DOCTYPE html>
<html>
  <head>
    <title>Ejercicio 1</title>
    <style>
      .parrafo-personalizado {
        background-color: #e0f0ff;
        color: #003366;
        font-weight: bold;
        padding: 15px;
        border: 2px solid #0066cc;
        border-radius: 10px;
        font-family: Arial, sans-serif;
      }
    </style>
  </head>
  <body>
    <p class="parrafo-personalizado">
      Este es un párrafo con estilo personalizado usando CSS. Observa cómo
      cambia su apariencia gracias a las propiedades CSS que hemos
      aplicado.
    </p>
  </body>
</html>

```

### Ejercicio 2: Tarjeta de presentación

Crea una tarjeta simple con un título, imagen y descripción.

```

<!DOCTYPE html>
<html>
  <head>

```

```

<title>Tarjeta de Presentación</title>
<style>
  .tarjeta {
    width: 300px;
    border: 1px solid #dddddd;
    border-radius: 8px;
    padding: 15px;
    margin: 20px auto;
    font-family: Arial, sans-serif;
  }

  .tarjeta-titulo {
    color: #333333;
    font-size: 22px;
    margin-bottom: 10px;
    text-align: center;
  }

  .tarjeta-imagen {
    width: 100%;
    border-radius: 5px;
    margin-bottom: 10px;
  }

  .tarjeta-descripcion {
    color: #666666;
    line-height: 1.4;
  }
</style>
</head>
<body>
  <div class="tarjeta">
    <h2 class="tarjeta-titulo">Mi Mascota</h2>
    
    <p class="tarjeta-descripcion">
      Esta es mi mascota. Le gusta jugar en el parque y dormir bajo el
      sol. Es
      muy cariñosa y juguetona.
    </p>
  </div>
</body>
</html>

```

### Ejercicio 3: Formulario con estilo

Crea un formulario básico con campos de entrada estilizados.

```

<!DOCTYPE html>
<html>
  <head>
    <title>Formulario con Estilo</title>
    <style>
      .formulario {
        width: 400px;
        margin: 0 auto;
        padding: 20px;
        background-color: #f8f8f8;
        border-radius: 10px;
        font-family: Arial, sans-serif;
      }

      .titulo-formulario {
        text-align: center;
        color: #333333;
        margin-bottom: 20px;
      }

      .campo {
        margin-bottom: 15px;
      }

      .etiqueta {
        display: block;
        margin-bottom: 5px;
        font-weight: bold;
        color: #555555;
      }

      .entrada {
        width: 100%;
        padding: 8px;
        border: 1px solid #dddddd;
        border-radius: 4px;
      }

      .boton {
        background-color: #4caf50;
        color: white;
        padding: 10px 15px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
        font-size: 16px;
      }

      .boton:hover {
        background-color: #45a049;
      }
    </style>
  </head>

```

```

<body>
  <form class="formulario">
    <h2 class="titulo-formulario">Regístrate</h2>

    <div class="campo">
      <label class="etiqueta">Nombre:</label>
      <input type="text" class="entrada" placeholder="Ingresa tu nombre"
    />
    </div>

    <div class="campo">
      <label class="etiqueta">Correo electrónico:</label>
      <input type="email" class="entrada" placeholder="Ingresa tu email"
    />
    </div>

    <div class="campo">
      <label class="etiqueta">Contraseña:</label>
      <input
        type="password"
        class="entrada"
        placeholder="Crea una contraseña"
      />
    </div>

    <button type="submit" class="boton">Registrarse</button>
  </form>
</body>
</html>

```

## Consejos para principiantes

1. **Experimenta constantemente:** Cambia los valores de las propiedades para ver cómo afectan al diseño.
2. **Usa las herramientas de desarrollo del navegador:** Chrome, Firefox y otros navegadores tienen herramientas para inspeccionar y modificar CSS en tiempo real.
3. **Mantén tu código organizado:** Agrupa las propiedades relacionadas y usa comentarios para documentar tu código.
4. **Comienza con proyectos simples:** No intentes diseños complejos de inmediato. Empieza con elementos básicos.
5. **Aprende de sitios web existentes:** Inspecciona el CSS de tus sitios web favoritos para entender cómo están contruidos.
6. **Practica la especificidad:** Comprende cómo los diferentes tipos de selectores afectan la precedencia de los estilos.

Estos temas proporcionan una base sólida para comenzar con CSS. Recuerda que la práctica constante es clave para dominar CSS.



