

Hojas de estilo en cascada

[3.0 Introducción](#)

[3.1 Selectores y declaraciones](#)

[3.2 Propiedades de la fuente](#)

[3.3 Propiedades del color](#)

[3.4 Listas](#)

[3.5 Alineación del texto](#)

[3.6 Imágenes de fondo](#)

[3.7 El modelo de caja](#)

[3.7.1 Bordes](#)

[3.7.2 Márgenes y espacio de relleno](#)

3.0 Introducción

POR QUÉ

CSS significa Cascading Style Sheets (Hojas de estilo en cascada). El lenguaje HTML se usa para añadir un significado y una estructura al contenido de una página web, mientras que el CSS describe *cómo* se debe mostrar dicho contenido. Nos solemos referir a este "cómo" con el nombre de *estilo*.

Aunque se puede establecer el estilo y diseño con lenguaje HTML (con tablas para el diseño o elementos de la fuente para el estilo, por ejemplo), con Cascading Style Sheets es mucho mejor, porque separan el código para añadir significado (HTML) del de la presentación (CSS).

CÓMO

Las CSS se pueden usar de tres formas:

- **Integradas**
 - Las style sheets (hojas de estilo) integradas se aplican al contenido de un único elemento. Se encuentran entre la apertura y el cierre de una etiqueta y se aplican solo al contenido de dicha etiqueta.
 - Las especificaciones de las hojas de estilo integradas se muestran como valores del atributo *style* en una etiqueta como esta:
`<p style="color: red;">Te damos la bienvenida a las`

CSS.</p>

Ejemplo

```
<!DOCTYPE html>
<html>

<head>
<title>Mi página</title>
</head>

<body>
<p style="color:red">texto</p>
</body>

</html>
```

- Esta forma no suele ser muy buena para aplicar estilos ya que mezcla el código CSS con HTML y puede llegar a ser difícil de mantener.

- **Nivel de documento**

- Las style sheets a nivel de documento se aplican a todo el documento.
- Añade un elemento de estilo a la etiqueta <head> de un documento como este:

```
<style>
p {
color: red;
}
</style>
```

Ejemplo

```
<!DOCTYPE html>
<html>

<head>
<title>Mi página</title>

<style type="text/css">
p {
color: red;
}
```

```
}
</style>
</head>

<body>
<p>texto</p>
</body>

</html>
```

- Esta acción separa el código CSS del HTML y mantiene todas las reglas de CSS en un único lugar, lo que va muy bien. Sin embargo, con este método el código CSS sigue estando en un archivo HTML y no se puede compartir con otros documentos.

- **Externas**

- Las style sheets externas se pueden aplicar a tantos documentos como quieras.
- La etiqueta <link> se usa para definir style sheets externas. Añade una etiqueta link a <head>:

```
<head>
...
<link rel="stylesheet" href="/css/global.css">
...
</head>
```

- Puedes enlazarlos a más de una style sheet:

```
<head>
...
<link rel="stylesheet" href="/css/global.css">
<link rel="stylesheet" href="css/local.css">
...
</head>
```

Ejemplo

```
<!DOCTYPE html>
<html>

<head>
```

```
<title>Mi página</title>
<link rel="stylesheet" href="my_styles.css">
</head>

<body>
<p>texto</p>
</body>

</html>
```

- Al usar código CSS de un archivo externo, este se mantiene separado del HTML, con lo que es más fácil de entender y mantener el código CSS y te permite compartir estilos entre documentos. También permite que los navegadores lo almacenen en la memoria caché para que los archivos CSS solo se tengan que cargar una vez.

CONSEJO: Todos los navegadores tienen una style sheet predeterminada. Por ejemplo, para definir el tamaño de los títulos o el margen predeterminado de los elementos anteriores y posteriores al párrafo. Aquí tienes una lista de las style sheets de los navegadores.

GLOSARIO

Las Cascading Style Sheets son muy útiles para especificar un orden de prioridad en caso de haber conflictos entre elementos de estilo.

3.1 Selectores y declaraciones

CÓMO

Todas las reglas de CSS están formadas por tres partes:

1. Selector
2. Propiedades
3. Valores

El selector indica los elementos a los que afecta la regla.

Una propiedad ('color') junto con un valor ('red') forman una declaración.

Por ejemplo, puedes hacer que el texto de todos los elementos `p` sea rojo con la siguiente regla de CSS:

```
p { color: red; }
```

En este ejemplo, `p` es el selector.

`color` es la propiedad, que siempre va seguida de dos puntos (:)

red es el valor, que siempre va seguido de un punto y coma (;)

Puedes añadir varias declaraciones a cada regla:

```
p {color: red; font-size: 12px;}
```

Coloca cada declaración en una línea diferente para que sea más fácil leerlas:

```
p {  
  color: red;  
  font-size: 12px;  
}
```

CONSEJO: Te recomendamos que introduzcas un punto y coma en todas las declaraciones, aunque solo haya una. Así te evitarás problemas si añades más posteriormente.

CÓMO

Los selectores pueden tener muchas formas:

INTÉNTALO

1. Nombre del elemento

Los valores de la propiedad se aplican a todas las repeticiones de los elementos mencionados.

```
h1 {color: red;}
```

Puedes aplicar los mismos valores a varios elementos:

```
h2, h3 {color: green;}
```

2. Selector de clase

Selecciona elementos con atributos de clase como este:

HTML:

```
<p class="highlight">¡Oh, oh!</p>
```

CSS:

```
p.highlight {  
  color: red;  
  background-color: yellow;  
}
```

No incluyas el nombre del elemento para aplicar los valores a varios elementos con la misma clase:

HTML:

```
<p class="highlight">¡Oh, oh!</p>
...
<blockquote class="highlight">Alguna citación
divertida.</blockquote>
```

CSS:

```
.highlight {
  color: red;
  background-color: yellow;
}
```

3. Selectores de ID

Aplican un estilo a un único elemento.

```
#intro {
  font-size: 18pt;
}

<p id="intro">El tamaño de este texto será 18. </p>
```

4. Selectores universales

El selector universal se aplica a todos los elementos del documento y se marca con un asterisco (*).

```
* {
  color: yellow;
}
```

ADVERTENCIA: El selector universal no es una opción recomendable. Puede empeorar el rendimiento y dificultar la depuración de las CSS, ya que la regla se aplica a todos los elementos.

GLOSARIO

Los selectores se usan para buscar los elementos HTML a los que quieres aplicar un estilo.

Puede tratarse del nombre del elemento, el ID o el nombre de la clase: `h1`, `#intro`, `.warning`, etc.

Las declaraciones tienen dos partes: la propiedad y el valor (`color: red`, por ejemplo).

CONSEJO: No te olvides de usar los comentarios para que tu código CSS sea más fácil de leer. Los comentarios se escriben con este formato:

```
/* Esto es un comentario, sin importar el formato. */
```

3.2 Propiedades de la fuente

POR QUÉ

En cualquier documento web, verás que algunos aspectos de la fuente pueden variar: el tamaño, el conjunto de fuentes (Arial, Helvetica etc.) o el estilo (**negrita**, *cursiva* o subrayado), entre otras propiedades de la fuente.

INTÉNTALO

1. Indica el orden de preferencia de las fuentes

```
font-family: Arial, Helvetica, Futura, sans-serif
```

2. Especifica el tamaño del texto

```
font-size: 14pt
```

3. Pon el texto en cursiva:

```
font-style: italic
```

4. Pon el texto en negrita:

```
font-weight: bold
```

Propiedades abreviadas

Puedes especificar más de un valor mediante la propiedad `font`:

```
font: bold 12pt Arial
```

GLOSARIO

Font-family se refiere al tipo de letra (Helvetica, Arial etc.).

Font-size se refiere al tamaño del texto.

Font-style se refiere al estilo del texto (*en cursiva*, **negrita** o normal).

3.3 Propiedades del color

POR QUÉ

Determina el color con uno de los dieciséis nombres de colores: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white y yellow (aguamarina, negro, azul, fucsia, gris, verde, lima, rojo oscuro, azul marino, verde oliva, morado, rojo, plateado, esmeralda, blanco y amarillo). En vez de usar los nombres, puedes usar valores numéricos para indicar la cantidad de rojo, verde y azul que quieres aplicar. Las tres reglas siguientes tienen el mismo efecto. La primera usa cifras decimales mientras que la segunda, hexadecimales: FF es igual a 255 en hexadecimales (base 16).

```
h1 {color: rgb(255, 0, 0);}
h1 {color: #FF0000;}
h1 {color: red;}
```

También puedes definir el color del fondo de un elemento. En el ejemplo siguiente, el texto estaría en rojo y el color del fondo del elemento H1 sería amarillo.

```
h1 {
  background-color: yellow;
  color: red;
}
```

INTÉNTALO

```
p.bright {color: red; background-color: yellow}
<p> Este texto sería rojo con el fondo amarillo. </p>

h1 {
    color: red;
}
<h1>Este texto sería rojo.</h1>
```

GLOSARIO

Color se refiere al color de un elemento. En los elementos que contienen texto, es el color de lo que hay en primer plano. El valor puede ser el nombre del color (p. ej. `black` o `red`, negro o rojo) o el valor hexadecimal o decimal del color (p. ej. `#ff0000` o `rgb(255, 0, 0)`).

Background-color se refiere al color del fondo de un elemento.

3.4 Alineación del texto y posición

Alineación del texto

POR QUÉ

Puedes alinear el texto a la izquierda (opción predeterminada) o a la derecha o centrarlo o justificarlo (con un borde recto en ambos lados). También puedes aplicar sangría a la primera línea de un bloque de texto.

El código CSS dispone de una serie de unidades para expresar la longitud que sirven para definir el aspecto del texto en los medios de salida, como la pantalla o el papel. Puedes usar píxeles (px), centímetros (cm) o incluso pulgadas (in).

CÓMO

Alinea el texto con la propiedad `text-align` y aplícale sangría con `text-indent`.

INTÉNTALO

```
<style>
    .body {
        text-align: justified;
        text-indent: 10px; }
</style>

<p class="body">
    Este párrafo empieza a 10 píxeles del margen. La
siguiente línea empezaría en el margen como de normal y todo el texto
del párrafo estaría justificado, es decir, tendría un borde recto
tanto a la izquierda como a la derecha.
</p>
```

Posición

POR QUÉ

La propiedad `position` determina el tipo de método de posicionamiento que se usa en un elemento. Hay cuatro tipos:

1. Static

El elemento se posiciona de acuerdo con el flujo normal de la página. Este es el valor predeterminado.

2. Fixed

El elemento se queda siempre en el mismo sitio, aunque el usuario se desplace por la página.

3. Relative

La posición del elemento es relativa a la posición normal.

4. Absolute

La posición del elemento es relativa al antecesor posicionado más cercano (elemento con un valor de `position` especificado). Si no tiene ningún antecesor posicionado, se usa el cuerpo del documento y sigue moviéndose cuando el usuario se desplaza por la página.

INTÉNTALO

```
<style>
    .box1 {
        position: relative;
        left: 10px; }
    .box2 {
        position: fixed;
        bottom: 20px; }
</style>

<p class="box1">
    Este párrafo empieza a 10 píxeles del
margen.
    </p>
    <p class="box2">
        Este párrafo empieza a 20 píxeles del margen y se
queda siempre en el mismo sitio, aunque el usuario se desplace por la
página.
    </p>
```

GLOSARIO

Text-indent se refiere a la cantidad de espacio en blanco que queda a la izquierda de la primera línea de un bloque de texto.

Text-align se refiere a la alineación de todo un bloque de texto.

3.5 Imágenes de fondo

POR QUÉ

La propiedad `background-image` se usa para establecer una imagen como fondo de un elemento.

CÓMO

Usa la propiedad `background-image` con una ruta de archivo:

```
h1 {background-image: url(/images/gradient.png);}
```

CONSEJO: La ruta de una imagen de fondo está relacionada con el documento que

contiene la regla. Así, pues, si usas `background-image: url(gradient.png)` en un archivo CSS en `/css/global.css`, el navegador buscará `gradient.png` en el mismo directorio CSS.

INTÉNTALO

Aplica una imagen de fondo al elemento del cuerpo de tu documento web.

3.6 La función Cascade

POR QUÉ

A veces, se puede aplicar más de una regla de hoja de estilo a un elemento CSS en concreto. Por ejemplo, una regla puede servir para que todos los elementos de una etiqueta `<p>` se vuelvan rosa y otra, para que se vuelvan azul. En casos como este, se necesita una forma para determinar la regla de hoja de estilo que se debe aplicar al elemento. Básicamente, cascade determina las declaraciones que se aplican a cualquier elemento.

CÓMO

Como es muy probable que se aplique una o más declaraciones a un elemento, el método "Cascading" se usa para definir la que debe aplicarse. Si existen conflictos entre las declaraciones, se usa la última y la más específica.

INTÉNTALO

```
<style>
    div {
        color: blue;
        font-style: italic; }
    p {
        color: pink; }
</style>

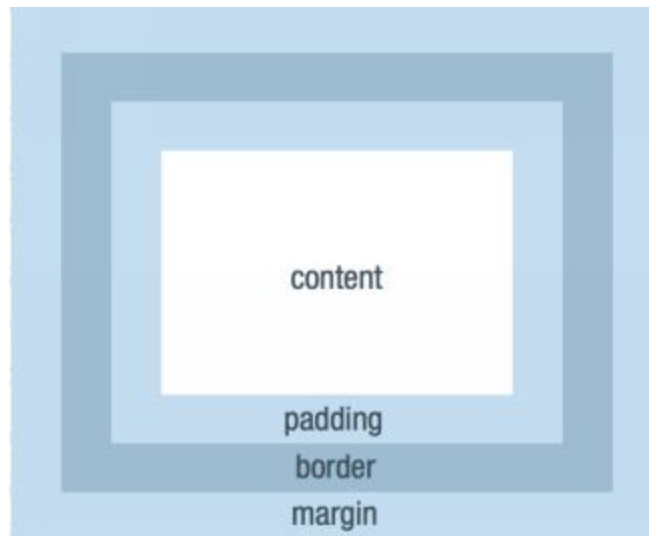
<div>
<p> Este texto estaría en rosa. </p>
</div>
```

3.7 El modelo de caja

QUÉ ES

En un documento, los elementos se representan con una caja rectangular. Es lo que llamamos "El modelo de caja". Básicamente se trata de una caja que envuelve cada uno de los elementos

HTML y tiene cuatro bordes: márgenes, bordes, espacio de relleno y contenido. La siguiente imagen es un ejemplo del modelo de caja:



Content es el área que contiene el contenido real del elemento HTML: el texto o la imagen, por ejemplo.

Padding es el área que rodea el contenido.

Border es el borde que rodea el área padding. Puede tener propiedades como border-style, border-width y border-color.

Margin es el espacio vacío que queda entre dos elementos adjuntos: en la parte superior e inferior, y a derecha e izquierda.

3.7.1 Bordes

POR QUÉ

Es la propiedad Borders controla si un elemento tiene bordes y el estilo de estos.

Los bordes tienen varias características, como el estilo, el color o el ancho, entre otros.

CÓMO

Cada uno de los lados del borde (superior, inferior, derecho e izquierdo) pueden tener un estilo, ancho o color diferente. Los bordes tienen tres propiedades clave:

- **border-style**
 - Esta propiedad define el estilo del borde, que puede ser *dotted* (de puntos), *dashed* (de guiones) o *double* (doble).
 - el valor predeterminado para border-style es *none* (ninguno).
 - Uno de los cuatro extremos de un elemento puede definirse con los estilos `border-top-style`, `border-bottom-style`, `border-left-style` y `border-right-style`.
- **border-width**

- Esta propiedad define lo grueso o estrecho que es el borde. Los valores pueden ser: *thin* (estrecho), *medium* (medio), *thick* (grueso) o un valor de grosor en píxeles.
- El valor predeterminado para `border-width` es *medium*.
- El grosor de uno de los cuatro extremos de un elemento puede definirse como `border-top-width`, `border-bottom-width`, `border-left-width` y `border-right-width`.
- `border-color`
 - Esta propiedad define el color del borde, y puedes usar el nombre del color o un valor hexadecimal.
 - El color predeterminado es *black* (negro).
 - El color de uno de los cuatro extremos de un elemento puede definirse como `border-top-color`, `border-bottom-color`, `border-left-color` y `border-right-color`.

INTÉNTALO

```
<head>
<link rel="stylesheet" type="text/css" href="boxmodel.css">
</head>
```

boxmodel.css:

```
table {
border-top-width: thin;
border-bottom-width: thick;
border-top-color: red;
border-bottom-color: yellow;
border-top-style: dotted;
}

p {
border-width: thick;
border-color: aqua;
border-style: dashed;
}
```

Ahora añade una tabla y un elemento para, guárdalo y ejecútalo. El resultado te mostrará las propiedades del borde.

GLOSARIO

Border es el área que se encuentra entre el espacio de relleno y el margen.

Border-style se refiere al aspecto del borde: una línea de puntos o de guiones, por ejemplo.

Border-width se refiere al ancho del borde.

Border-color se refiere al color del borde.

3.7.2 Márgenes y espacio de relleno

POR QUÉ

La cantidad de espacio entre el contenido de un elemento y su borde se llama espacio de relleno.

El espacio entre el borde de un elemento y el elemento inmediatamente junto a él (en la parte superior o inferior, o a la derecha o izquierda) se llama margen.

CÓMO

Las propiedades de margen se llaman *margin* y las de espacio de relleno, *padding*. Estas propiedades se pueden aplicar a los cuatro lados de una imagen: superior, inferior, izquierdo y derecho.

Así, pues, las propiedades de margen de un elemento pueden ser: `margin-top`, `margin-bottom`, `margin-left` y `margin-right`. Las del espacio de relleno pueden ser: `padding-top`, `padding-bottom`, `padding-left` y `padding-right`.

Los valores del margen y del espacio de relleno están en pulgadas o píxeles (o cualquier unidad de tamaño).

INTÉNTALO

```
<head>
<link rel="stylesheet" type="text/css" href="margins.css">
</head>
```

margins.css:

```
    p.one {
margin: 0.1in;
padding: 0.1in;
background-color: red;
    }
    p.two {
margin: 0.2in;
padding: 0.2in;
background-color: blue;
    }
    p.three {
margin: 20px;
padding: 10px;
background-color: yellow;
    }
    p.four {
```

```
margin: 40px;  
background-color: red;  
}
```

Ahora añade cuatro elementos para con nombres de clase como *one*, *two*, *three* y *four*, respectivamente, y ejecútalos. El resultado te mostrará las propiedades de margen y de espacio de relleno.

Recuerda guardar todo lo que hagas antes de pasar a la siguiente sección.

GLOSARIO

Margin se refiere al espacio entre el contenido del elemento y el elemento adjunto.

Padding se refiere al espacio entre el elemento y su borde.