

CSS básico

CSS (*Hojas de Estilo en Cascada*) es el código que usas para dar estilo a tu página web. *CSS Básico* te lleva a través de lo que tú necesitas para empezar. Contestará a preguntas del tipo: ¿Cómo hago mi texto rojo o negro? ¿Cómo hago que mi contenido se muestre en tal y tal lugar de la pantalla? ¿Cómo decoro mi página web con imágenes de fondo y colores?

Entonces ¿qué es CSS, realmente?

Como HTML, CSS (*Cascading Style Sheets*) u Hojas de estilo en cascada en español, no es realmente un lenguaje de programación, tampoco es un lenguaje de marcado. Es un *lenguaje de hojas de estilo*, es decir, te permite aplicar estilos de manera selectiva a elementos en documentos HTML. Por ejemplo, para seleccionar **todos** los elementos de párrafo en una página HTML y volver el texto dentro de ellos de color rojo, has de escribir este CSS:

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

Copy to Clipboard

Vas a probarlo: pega estas tres líneas de CSS en un nuevo archivo en tu editor de texto y guarda este archivo como `style.css` en tu directorio `styles` (estilos).

Pero aún debes aplicar el CSS a tu documento HTML, de otra manera el estilo CSS no cambiará cómo tu navegador muestra el documento HTML. (Si no has seguido nuestro proyecto, lee [Manejo de archivos](#) y [HTML básico](#) para averiguar qué necesitas hacer primero.)

1. Abre tu archivo `index.html` y pega la siguiente línea en algún lugar dentro del `<head>`, es decir, entre las etiquetas `<head>` y `</head>`:

```
<link href="styles/style.css" rel="stylesheet" type="text/css">
```

Copy to Clipboard

2. Guarda el archivo `index.html` y cárgalo en tu navegador. Debes ver algo como esto:

My test page

file:///Users/chrisfills/Dropbc Google

Most Visited MDN Mozilla general FirefoxOS ecosy... Meeting details

Mozilla is cool



At Mozilla, we're a global community of

- technologists
- thinkers
- builders

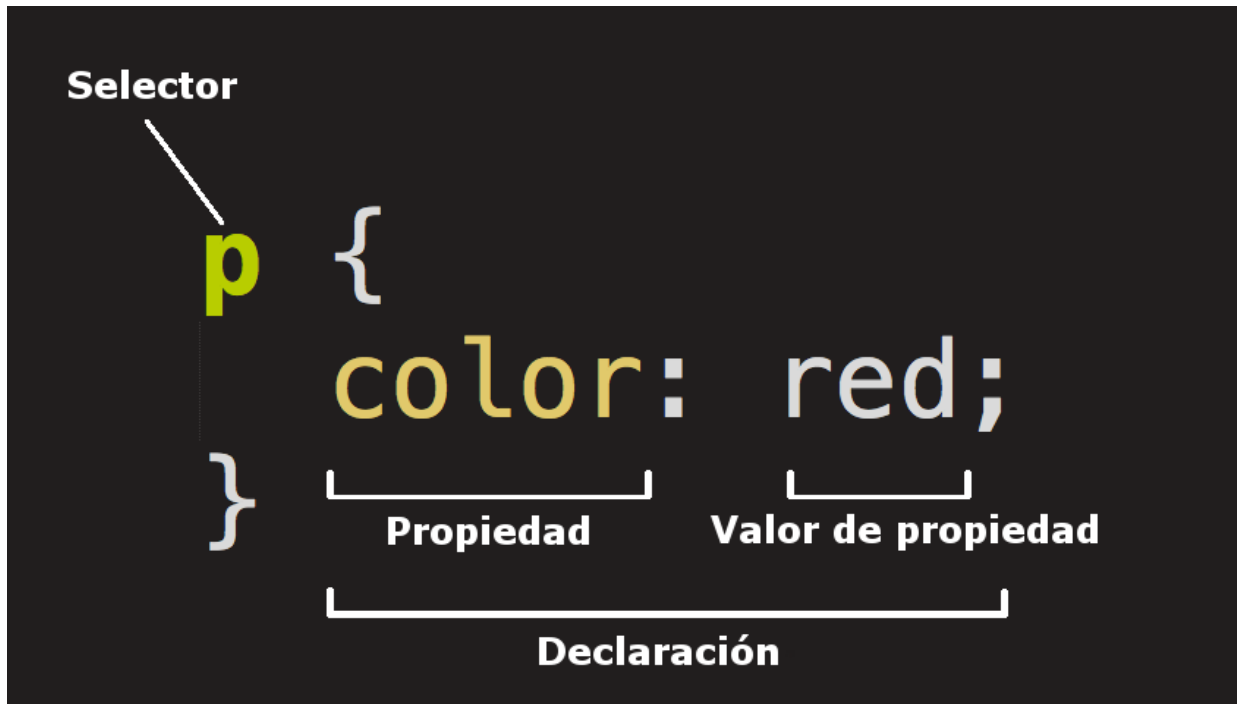
working together to keep the Internet alive and accessible, so people worldwide can be informed contributors and creators of the Web. We believe this act of human collaboration across an open platform is essential to individual growth and our collective future.

Read the [Mozilla Manifesto](#) to learn even more about the values and principles that guide the pursuit of our mission.

ahora es rojo, ¡felicitaciones, ya has escrito tu primer CSS de forma exitosa!

Anatomía de una regla CSS

Observa el código CSS de arriba, un poco más a detalle:



La estructura completa es llamada **regla predeterminada** (pero a menudo «regla» para abreviar). Nota también los nombres de las partes individuales:

Selector

El elemento HTML en el que comienza la regla. Esta selecciona el(los) elemento(s) a dar estilo (en este caso, los elementos `<p>`). Para dar estilo a un elemento diferente, solo cambia el selector.

Declaración

Una sola regla como `color: red;` especifica a cuál de las **propiedades** del elemento quieres dar estilo.

Propiedades

Maneras en las cuales puedes dar estilo a un elemento HTML. (En este caso, `color` es una propiedad del elemento `<p>`). En CSS, seleccionas qué propiedad quieres afectar en tu regla.

Valor de la propiedad

A la derecha de la propiedad, después de los dos puntos (:), tienes el **valor de la propiedad**, para elegir una de las muchas posibles apariencias para una propiedad determinada (hay muchos valores para `color` además de `red`).

Nota las otras partes importantes de la sintaxis:

- Cada una de las reglas (aparte del selector) deben estar encapsuladas entre llaves ({}).
- Dentro de cada declaración, debes usar los dos puntos (:) para separar la propiedad de su valor.
- Dentro de cada regla, debes usar el punto y coma (;) para separar una declaración de la siguiente.

De este modo para modificar varios valores de propiedad a la vez, solo necesitas escribirlos separados por punto y coma (;), así:

```
p {  
  
  color: red;  
  
  width: 500px;  
}
```

```
border: 1px solid black;  
  
}
```

Copy to Clipboard

Seleccionar varios elementos

También puedes seleccionar varios elementos y aplicar una sola regla a todos ellos. Incluye varios selectores separados por comas (.). Por ejemplo:

```
p,li,h1 {  
  
  color: red;  
  
}
```

Copy to Clipboard

Diferentes tipos de selectores

Existen muchos tipos diferentes de selectores. Antes, solo viste los **selectores de elementos**, los cuales seleccionan todos los elementos de un tipo dado en los documentos HTML. Sin embargo puedes hacer selecciones más específicas que esas. En seguida están algunos de los tipos de selectores más comunes:

Nombre del selector	Qué selecciona	Ejemplo
Selector de elemento (llamado algunas veces selector de etiqueta o tipo)	Todos los elementos HTML del tipo especificado.	p Selecciona <p>
Selector de identificación (ID)	El elemento en la página con el ID especificado (en una página HTML dada, solo se permite un único elemento por ID).	#mi-id Selecciona <p id="mi-id"> y
Selector de clase	Los elementos en la página con la clase especificada (una clase puede aparecer varias veces en una página).	.mi-clase Selecciona <p class="mi-clase"> y
Selector de atributo	Los elementos en una página con el atributo especificado.	img[src] Selecciona pero no

Nombre del selector	Qué selecciona	Ejemplo
Selector de pseudoclase	Los elementos especificados, pero solo cuando esté en el estado especificado, por ejemplo cuando el puntero esté sobre él.	a: hover Selecciona <a>, pero solo cuando el puntero esté sobre el enlace.

Existen muchos más selectores para explorar, y podrás encontrar una lista más detallada en la [guía de Selectores](#).

Fuentes y texto

Ahora que has explorado lo básico de CSS, empieza por añadir información y algunas reglas más a tu archivo `style.css` para que tu ejemplo se vea bonito. Primero, haz que tus fuentes y texto luzcan un poco mejor.

1. Antes que nada, regresa y busca las [fuentes de Google Fonts](#) que guardaste en un lugar seguro. Agrega el elemento `<link>...` en algún lugar del `head` de tu archivo `index.html` (de nuevo, en cualquier lugar entre las etiquetas `<head>` y `</head>`). Debe verse algo así:

```
<link href="https://fonts.googleapis.com/css2?family=Open+Sans" rel="stylesheet" type="text/css">
```

Copy to Clipboard

2. Luego, borra la regla existente en tu archivo `style.css`. Fue una buena prueba, pero el texto en rojo en realidad no se ve muy bien.
3. Añade las siguientes líneas (que se muestran a continuación), sustituyendo la asignación de `font-family` por tu selección de `font-family` que obtuviste en [¿Cuál será la apariencia de tu sitio Web?](#) La propiedad `font-family` se refiere a la(s) fuente(s) que deseas usar en tu texto. Esta regla define una fuente base global y un tamaño de fuente

para usar en toda la página. Dado que `<html>` es el elemento primario (o padre) de toda la página, todos los elementos contenidos dentro de él heredan las propiedades `font-size` y `font-family`):

```
4. html {  
5.   font-size: 10px; /* px quiere decir 'píxeles': el tamaño de la fuente base es ahora de 10 píxeles de altura */  
6.   font-family: "Open Sans", sans-serif; /* Este debe ser el resto del resultado que obtuviste de Google fonts */  
7.  
8. }
```

Copy to Clipboard

Nota: se ha añadido un comentario para explicar qué significa «px». Todo lo que está en un documento de CSS entre `/*` y `*/` es un **comentario en CSS**, el cual el navegador descarta cuando carga el código. Este es un espacio donde puedes escribir notas útiles sobre lo que estás haciendo.

7. Ahora escoge el tamaño de fuente para los elementos que contienen texto dentro del cuerpo del HTML (`<h1>`, ``, y `<p>`). También centra el texto del título, escoge un ancho de línea y espaciado entre letras en el contenido del texto para hacerlo un poco más legible:

```
8. h1 {  
9.   font-size: 60px;  
10.  text-align: center;  
11. }  
12.  
13. p, li {
```

```
14. font-size: 16px;  
15. line-height: 2;  
16. letter-spacing: 1px;
```

```
}
```

Copy to Clipboard

Puedes ajustar estos valores en px para lograr que tu diseño luzca como deseas, pero por lo general tu diseño debe verse así:

My test page

file:///Users/chrisfills/Dropbox/Work/git/web-projects/tes Google

Most Visited MDN Mozilla general FirefoxOS ecosy... Meeting details Other Moz depar... W3C

Mozilla is cool



At Mozilla, we're a global community of

- technologists
- thinkers
- builders

working together to keep the Internet alive and accessible, so people worldwide can be informed contributors and creators of the Web. We believe this act of human collaboration across an open platform is essential to individual growth and our collective future.

Read the [Mozilla Manifesto](#) to learn even more about the values and principles that guide the pursuit of our mission.

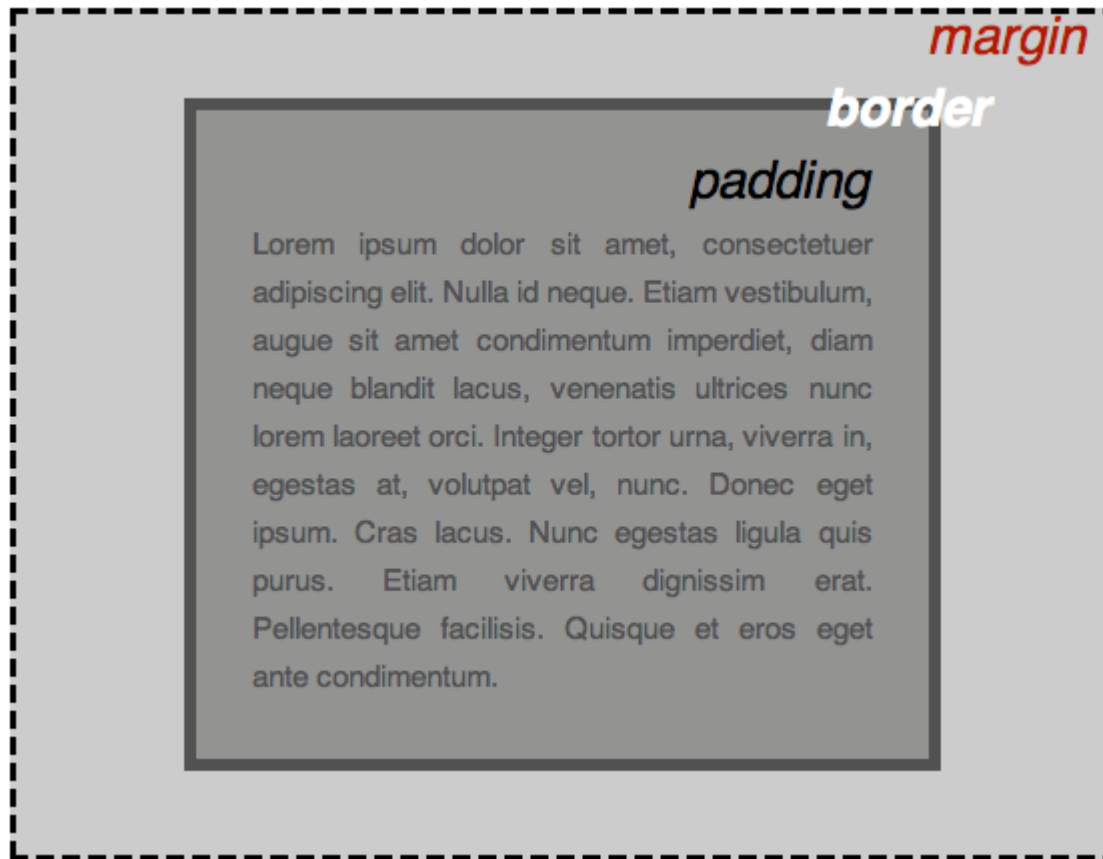
Cajas, cajas, todo se trata de cajas

Una cosa que notarás sobre la escritura de CSS es que trata mucho sobre cajas —ajustando su tamaño, color, posición, etc—. Puedes pensar en la mayoría de los elementos HTML de tu página como cajas apiladas una sobre la otra.



No es de extrañar que el diseño de CSS esté basado principalmente en el *modelo de caja*. Cada una de las cajas que ocupa espacio en tu página tiene propiedades como estas:

- `padding` (relleno), el espacio alrededor del contenido. En el ejemplo siguiente, es el espacio alrededor del texto del párrafo.
- `border` (marco), la línea que se encuentra fuera del relleno.
- `margin` (margen), el espacio fuera del elemento que lo separa de los demás.



En esta sección también se utiliza:

- `width` (ancho del elemento)
- `background-color`, el color de fondo del contenido y del relleno
- `color`, el color del contenido del elemento (generalmente texto)

- `text-shadow`: coloca una sombra difuminada en el texto dentro del elemento
- `display`: selecciona el modo de visualización para el elemento (no te preocupes de esto por ahora)

Bien, ¡continúa y agrega más código CSS a la página! Sigue añadiendo estas reglas nuevas al final de la página, y no temas experimentar cambiando los valores para ver cómo resulta.

Cambiar el color de la página

```
html {  
  
  background-color: #00539F;  
  
}
```

Copy to Clipboard

Esta regla asigna un color de fondo a la página entera. Puedes cambiar el código de color por cualquiera [como el que elegiste usar en tu proyecto](#).

Dar estilo al cuerpo del documento

```
body {  
  
  width: 600px;  
  
  margin: 0 auto;  
  
  background-color: #FF9500;  
  
  padding: 0 20px 20px 20px;
```

```
border: 5px solid black;
}
```

Copy to Clipboard

Ahora tienes varias declaraciones en el elemento `body`. Revisa una por una:

- `width: 600px;` — esto hará que el cuerpo siempre tenga 600 píxeles de ancho.
- `margin: 0 auto;` — cuando seleccionas dos valores dentro de propiedades como `margin` o `padding`, el primer valor afectará los lados superior (top) e inferior (bottom) (en este caso haciéndolo en 0), y el segundo valor los lados izquierdo (left) y derecho (right) (aquí, `auto` es un valor especial que divide el espacio disponible entre derecha e izquierda). Puedes usar esta propiedad con uno, dos, tres o cuatro valores como se explica en la [sintaxis de padding](#).
- `background-color: #FF9500;` — como antes, este selecciona el color de fondo de un elemento. Se ha usado un naranja rojizo para el elemento `body` en contraste con el azul oscuro del elemento `<html>`. Sigue y experimenta. Siéntete libre de usar `white` o cualquiera que sea de tu agrado.
- `padding: 0 20px 20px 20px;` — tienes 4 valores puestos en el relleno, para dar un poco de espacio alrededor del contenido. Esta vez no pondrás relleno en la parte de arriba de `body`, 20 píxeles a la izquierda, abajo y derecha. Los valores se ponen: arriba, derecha, abajo e izquierda, en ese orden. Como con `margin` usar esta propiedad con uno, dos, tres o cuatro valores como se explica en la [sintaxis de padding](#).
- `border: 5px solid black;` — este simplemente pone un borde de 5 píxeles de ancho, continuo y de color negro alrededor del elemento `body`.

Posicionar y dar estilo al título principal de la página

```
h1 {
  margin: 0;
  padding: 20px 0;
```



```
color: #00539F;

text-shadow: 3px 3px 1px black;

}
```

Copy to Clipboard

Puedes haber notado que hay un hueco horrible en la parte superior de *body*. Esto sucede porque los navegadores vienen con estilos por defecto, ¡incluso cuando aún no se ha aplicado ningún archivo CSS! Esto podría parecer una mala idea, pero se quiere que aun una página sin estilizar sea legible. Para deshacerte de este espacio elimina el estilo por defecto, agregando `margin: 0;`.

Enseguida, se ha puesto un relleno arriba y abajo del título de 20 píxeles, y se hizo que el color del texto sea el mismo que el color de fondo de `html`.

Una propiedad muy interesante que se ha usado aquí es `text-shadow`, que aplica una sombra al texto del elemento. Sus cuatro valores son como sigue:

- El primer valor en píxeles asigna el **desplazamiento horizontal** de la sombra desde el texto —qué tan lejos la mueve a la derecha—. Un valor negativo la moverá a la izquierda.
- El segundo valor en píxeles asigna el **desplazamiento vertical** de la sombra desde el texto —qué tan lejos la mueve hacia abajo—. En este ejemplo, un valor negativo la desplazaría hacia arriba.
- El tercer valor en píxeles asigna **radio de desenfoque** de la sombra —un valor grande es igual a una sombra borrosa—.
- El cuarto valor asigna el color base de la sombra.

Una vez más, trata de experimentar con diferentes valores para ver cómo resulta.

Centrar la imagen

```
img {  
  
  display: block;  
  
  margin: 0 auto;  
  
}
```

Copy to Clipboard

Finalmente, centra la imagen para hacer que luzca mejor. Puedes usar nuevamente el truco de `margin: 0 auto` que usaste antes para `body`, pero existen diferencias que requieren que hagas algo más para que el código CSS funcione.

El elemento `<body>` es un elemento en nivel de bloque (**block-level**), lo que significa que tomará espacio en la página y que puede tener otros valores de espacio aplicables como margen. Las imágenes, por otra parte, son elementos **inline**, lo que quiere decir que no puedes aplicarles márgenes, debes dar a la imagen un comportamiento de *block-level* usando `display: block`.

Nota: las instrucciones anteriores asumen que estás usando una imagen más pequeña que el ancho establecido en `body` (600 píxeles). Si tu imagen es más grande, desbordará el cuerpo, derramándose en el resto de la página. Para solucionar esto, puedes hacer lo siguiente: 1) reducir el ancho de la imagen usando un editor gráfico, o 2) usar CSS para dimensionar la imagen estableciendo la propiedad `width` en el elemento `` con un valor menor.

Nota: no te preocupes si aún no entiendes `display: block`; y la diferencia entre un elemento de bloque y un elemento *inline*. Lo entenderás en tanto estudies CSS a profundidad. Puedes encontrar más en cuanto a los diferentes valores disponibles para *display* en la página de referencia de display.