

Curso:

Programador Web Inicial - Front End Developer

Módulo 1:

Introducción a las tecnologías web

Unidad 3:

CSS



Presentación

En esta unidad nos introducimos en el lenguaje de estilos (CSS) para poder definir cómo va a ser la apariencia de nuestro sitio.



Objetivos

Que los participantes logren...

- Aprender el lenguaje CSS.
- Conocer cuál es la estructura de una regla en CSS
- Entender y utilizar los distintos selectores
- Conceptualizar las distintas propiedades básicas en CSS



Bloques temáticos

1. ¿Qué es CSS?
2. Estructura de la regla.
3. Selectores.
4. Propiedades básicas.
5. Posicionamiento
6. Flexbox

1. ¿Qué es CSS?

El CSS es un lenguaje de estilos empleado para definir la presentación, el formato y la apariencia de un documento html.

Las hojas de estilos nacen de la necesidad de diseñar la información de tal manera que podemos separar el contenido de la presentación y, así, por una misma fuente de información, generalmente definida mediante un lenguaje de demarcación, ofrecer diferentes presentaciones en función de dispositivos, servicios, contextos o aplicativos.

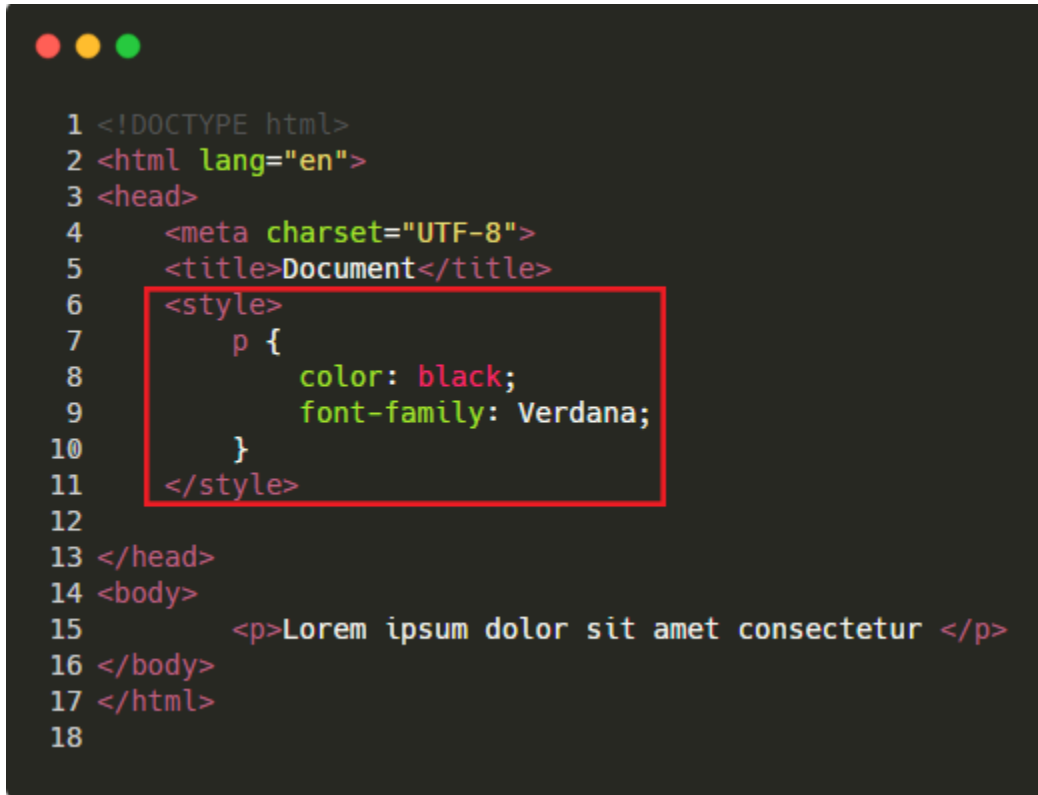
La especificación del CSS la mantiene el World Wide Web Consortium (W3C <http://www.w3c.org>).

¿Cómo incluir CSS en un documento HTML?

Una de las principales características de CSS es su flexibilidad y las diferentes opciones que ofrece para realizar una misma tarea. De hecho, existen tres opciones para incluir CSS en un documento HTML.

Incluir CSS en el mismo documento HTML

Los estilos se definen en una zona específica del propio documento HTML. Se emplea la etiqueta <style> de HTML y solamente se pueden incluir en la cabecera del documento (<head>).



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6   <style>
7     p {
8       color: black;
9       font-family: Verdana;
10    }
11  </style>
12
13 </head>
14 <body>
15   <p>Lorem ipsum dolor sit amet consectetur </p>
16 </body>
17 </html>
18
```

Este método se emplea cuando se define un número pequeño de estilos o cuando se quieren incluir estilos específicos en una determinada página HTML que complementen los estilos que se incluyen por defecto en todas las páginas del sitio web.

El principal inconveniente es que si se quiere hacer una modificación en los estilos definidos, es necesario modificar todas las páginas que incluyen el estilo que se va a modificar.

Definir CSS en un archivo externo

En este caso, todos los estilos CSS se incluyen en un archivo de tipo CSS que las páginas HTML enlazan mediante la etiqueta <link>. Un archivo de tipo CSS no es más que un archivo simple de texto cuya extensión es .css. Se pueden crear todos los archivos CSS que sean necesarios y cada página HTML puede enlazar tantos archivos CSS como necesite.

En el siguiente ejemplo, se crea un archivo de texto, se cambia su nombre a estilos.css y se incluye el siguiente contenido:

```
1 p{
2     color:red;
3     font-family:arial;
4 }
5
```

A continuación, en la página HTML se utiliza la etiqueta <link> para enlazar el archivo CSS externo que tiene los estilos que va a utilizar la página:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6     <link rel="stylesheet" href="estilos.css">
7 </head>
8 <body>
9     <p>Lorem ipsum dolor sit, amet consect</p>
10 </body>
11 </html>
```

Normalmente, la etiqueta incluye atributos cuando se enlaza un archivo CSS:

- **rel:** indica el tipo de relación que tiene el recurso enlazado (en este caso, el archivo CSS) y la página HTML. Para los archivos CSS, siempre se utiliza el valor stylesheet
- **href:** indica la URL del archivo CSS que contiene los estilos. La URL indicada puede ser relativa o absoluta y puede apuntar a un recurso interno o externo al sitio web.

Incluir CSS en los elementos HTML

Este es último método para incluir estilos CSS en documentos HTML es el menos utilizado, ya que el principal inconveniente es que si se quiere hacer una modificación en los estilos definidos, es necesario modificar todos los elementos en todas las páginas.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6 </head>
7 <body>
8   <p style="color: blue; font-family: Verdana;">Un párrafo de texto.</p>
9 </body>
10 </html>
```

2. Estructura de la regla



Imagen: <https://developer.mozilla.org>

Las partes principales son:

- **Selector:** El elemento HTML en el que comienza la regla. Esta selecciona el(los) elemento(s) a dar estilo. Para dar estilo a un elemento diferente, solo necesitamos cambiar el selector.
- **Declaración:** Especifica a cuál de las propiedades del elemento queremos dar estilo.
- **Propiedad:** Seleccionar qué propiedad queremos afectar en tu regla.
- **Valor de la propiedad:** es el valor de la propiedad, para elegir una de las muchas posibles apariencias para una propiedad determinada.

-

3. Selectores

Diferentes tipos de selectores

Existen diferentes tipos de selectores que se pueden utilizar en un sitio web.

En el cuadro de abajo están algunos de los tipos de selectores más comunes:

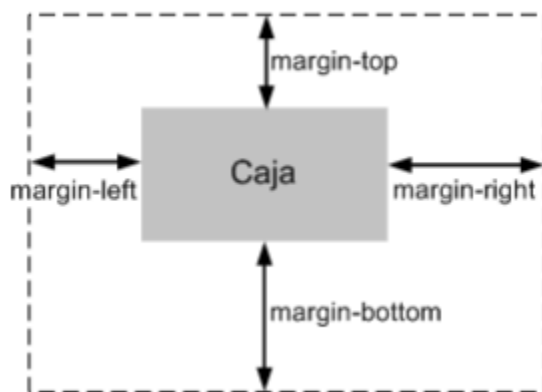
Nombre del selector	Qué selecciona	HTML	CSS
Selector de etiqueta o tipo	Todos los elementos HTML del tipo especificado.	<code><p> este es el contenido de..</p></code>	<code>p{ atributo:valor; }</code>
Selector de identificación (ID)	El elemento en la página con el ID especificado (en una página HTML dada, solo se permite un único elemento por ID).	<code><div id="testimonios"> </div></code>	<code>#testimonios{ atributo:valor; }</code>
Selector de clase	Los elementos en la página con la clase especificada (una clase puede aparecer varias veces en una página).	<code><div class="destacados"> </div></code>	<code>.destacados{ atributo:valor; }</code>

Selector de pseudoclase	Los elementos especificados, pero solo cuando esté en el estado especificado, por ejemplo cuando el mouse sobre él.	<code> Home </code>	<pre>a:hover{ atributo:valor; }</pre> <p>Selecciona <a>, pero solo cuando el mouse esté sobre el enlace.</p>
-------------------------	---	---	--

4. Propiedades básicas

Podemos encontrar las siguientes:

- **font-family:** se refiere a la(s) fuente(s) que deseamos usar en nuestro texto.
- **font-size:** define tamaño de la fuente
- **color:** color de la fuente
- **text-align:** alineación de los textos.
- **background-color:** cambiar el color de fondo del elemento..
- **width:** establece el ancho.
- **height:** establece el alto.
- **border:** no permite definir el aspecto de cada uno de los cuatro bordes horizontales y verticales de los elementos. Para cada borde se puede establecer su anchura, color y/o estilo.
- **margin:** define cuatro propiedades para controlar cada uno de los márgenes horizontales y verticales de un elemento.(margin-top margin-right margin-bottom margin-left).

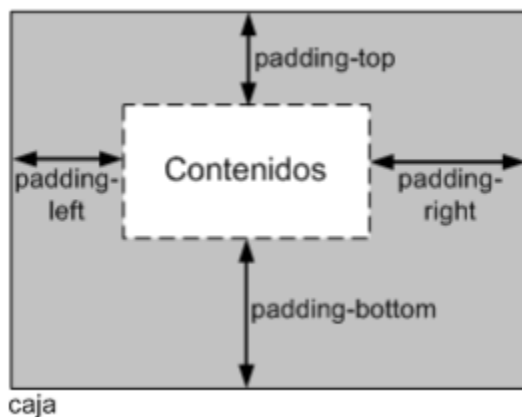


- **padding:** define cuatro propiedades para controlar cada uno de los espacios de relleno horizontales y verticales de un elemento. Cada una de las propiedades establece la separación entre el lateral de los contenidos y el borde lateral de la caja.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso

(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148

www.sceu.frba.utn.edu.ar/e-learning



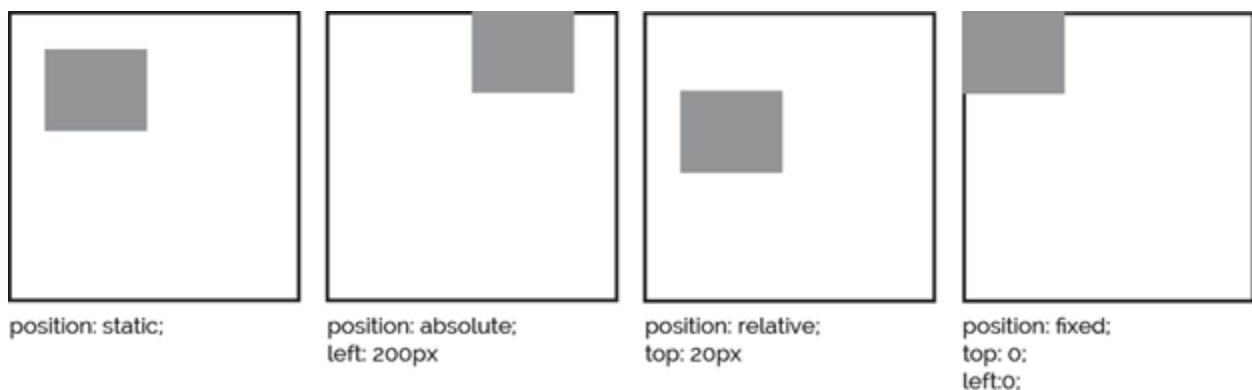
5. Posicionamiento

Los navegadores crean y posicionan de forma automática todas las cajas que forman cada página HTML. No obstante, CSS permite al diseñador modificar la posición en la que se muestra cada caja.

Utilizando las propiedades que proporciona CSS para alterar la posición de las cajas es posible realizar efectos muy avanzados y diseñar estructuras de páginas que de otra forma no serían posibles.

El estándar de CSS define cinco modelos diferentes para posicionar una caja:

- **Posicionamiento normal o estático (static):** se trata del posicionamiento que utilizan los navegadores si no se indica lo contrario.
- **Posicionamiento relativo (relative):** variante del posicionamiento normal que consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original.
- **Posicionamiento absoluto (absolute):** la posición de una caja se establece de forma absoluta respecto de su elemento contenedor y el resto de elementos de la página ignoran la nueva posición del elemento.
- **Posicionamiento fijo (fixed):** variante del posicionamiento absoluto que convierte una caja en un elemento inamovible, de forma que su posición en la pantalla siempre es la misma independientemente del resto de elementos e independientemente de si el usuario sube o baja la página en la ventana del navegador.

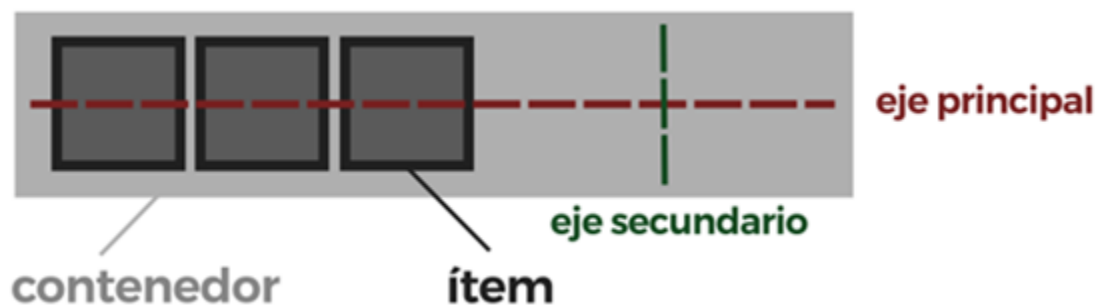


6. Flexbox

Las cajas flexibles, comúnmente llamadas flexbox, fueron diseñadas como un modelo unidimensional de layout, y como un método que pueda ayudar a distribuir el espacio entre los ítems de una interfaz y mejorar las capacidades de alineación.

Conceptos

Para empezar a utilizar flexbox lo primero que debemos hacer es conocer algunos de los elementos básicos de este nuevo esquema, que son los siguientes:



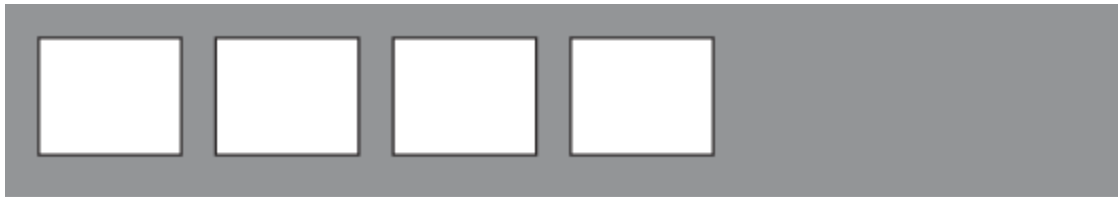
- **contenedor:** Existe un elemento padre que es el contenedor que tendrá en su interior cada uno de los ítems flexibles y adaptables.
- **ítem:** Cada uno de los hijos flexibles tendrá el contenedor en su interior.
- **eje principal:** Los contenedores flexibles tendrán una orientación principal específica. Por defecto, es en horizontal (fila).
- **eje secundario:** De la misma forma, los contenedores flexibles tendrán una orientación secundaria, perpendicular a la principal. Si la principal es en horizontal, la secundaria será en vertical, y viceversa.

Propiedad display

Controla la forma en la que se muestra el contenedor flexbox.

Valor	Descripción
flex	Establece un contenedor de ítems flexible en bloque.
inline-flex	Establece un contenedor de ítems flexible en línea (inline-block).

flex



inline-flex



Propiedad flex-direction

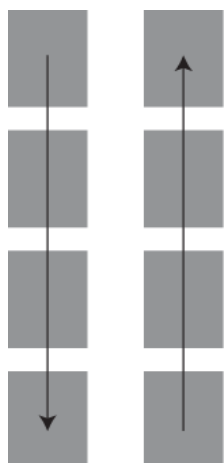
Define el eje principal y su dirección

Valor	Descripción
row	Establece la dirección del eje principal en horizontal.
row-reverse	Establece la dirección del eje principal en horizontal (invertido).
column	Establece la dirección del eje principal en vertical.
column-reverse	Establece la dirección del eje principal en vertical (invertido).

row/row-reverse



column / column-reverse



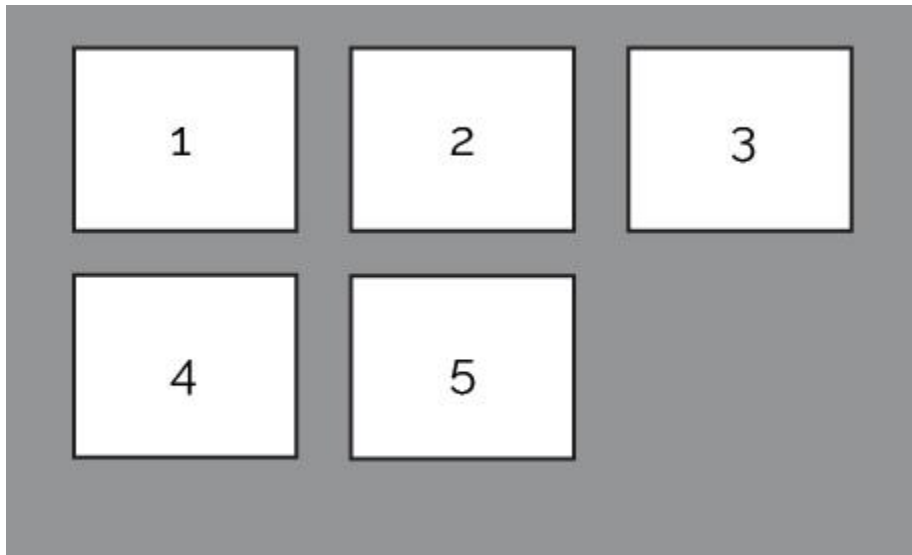
Propiedad flex-wrap

Controla el desbordamiento del contenedor cuando los ítems superan el ancho del mismo.

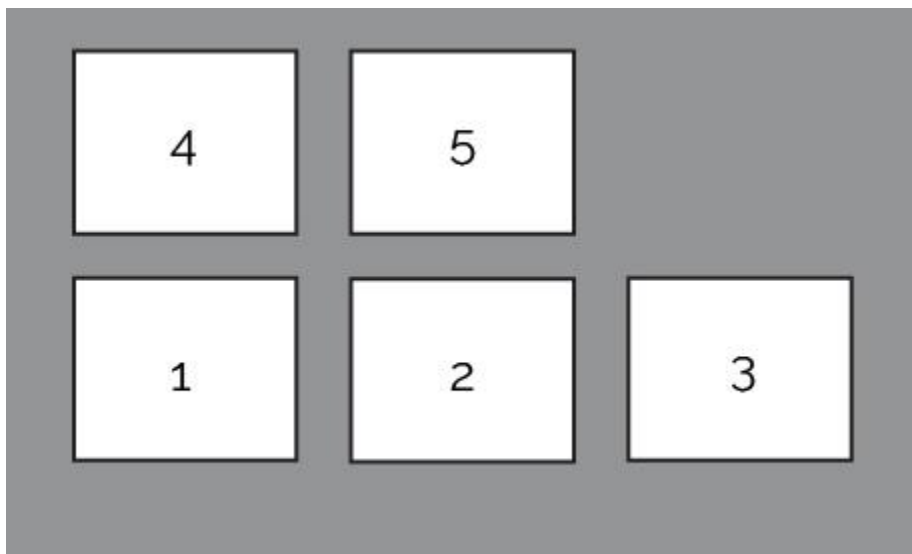
wrap

Valor	Descripción
nowrap	Establece los ítems en una sola línea (no permite desborde del contenedor).
wrap	Establece los ítems en modo multilínea (permite que se desborde el contenedor).
wrap-reverse	Establece los ítems en modo multilínea, pero en dirección inversa.

wrap



wrap-reverse



Propiedades de alineación de ítems

La propiedad **justify-content**, alinea los elementos en el eje principal

La propiedad **align-content**, alinea los elementos en el eje secundario cuando el contenedor es multilínea.

La propiedad **align-items**, alinea los elementos en el eje secundario cuando el contenedor tiene una línea única.

Estas son algunas de la características más usadas:

Valor	Descripción
flex-start	Los elementos se distribuyen desde el inicio de la línea.
flex-end	Los elementos se distribuyen al final de la línea.
center	Los elementos se centran a lo largo de la línea.
space-between	Los elementos se distribuyen uniformemente en la línea; el primer elemento está en la línea de inicio, el último elemento en la línea final.
space-around	Los elementos se distribuyen uniformemente en la línea con igual espacio a su alrededor. Tenga en cuenta que visualmente los espacios no son iguales, ya que todos los elementos tienen el mismo espacio en ambos lados.
space-evenly	Los elementos se distribuyen de manera que el espacio entre dos elementos (y el espacio hasta los bordes) sea igual.

flex-start



flex-end



center



space-between

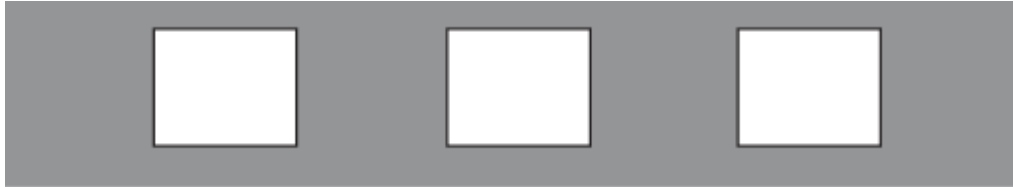


space-around



Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

space-evenly





Bibliografía utilizada y sugerida

Libros y otros manuscritos:

Castillo, Lucía. Iniciación al diseño Web. Editorial Dinámica. 2013.

Gauchat. Juan Diego, El gran libro de HTML5, CSS3 y Javascript. 1era Edición. 2012

Artículos de revista en formato electrónico:

Guia para utilizar Flexbox. Disponible desde la URL: ·

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Mozilla MDN Web Docs. Disponible desde la URL:

<https://developer.mozilla.org/es/>