



**UTN.BA**  
UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL BUENOS AIRES

**Centro de  
e-Learning**

# **Diplomatura en programación web full stack con React JS**

**Centro de e-Learning SCEU UTN - BA.**

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

**[www.sceu.frba.utn.edu.ar/e-learning](http://www.sceu.frba.utn.edu.ar/e-learning)**



**UTN.BA**  
UNIVERSIDAD TECNOLÓGICA NACIONAL  
FACULTAD REGIONAL BUENOS AIRES

**Centro de  
e-Learning**

p. 2

## **Módulo 1: HTML y CSS**

### **Unidad 3: CSS Inicial**

**Centro de e-Learning SCEU UTN - BA.**

Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

**[www.sceu.frba.utn.edu.ar/e-learning](http://www.sceu.frba.utn.edu.ar/e-learning)**



## Presentación:

Como mencionamos en la Unidad 1, mientras que en HTML estructuramos la información a mostrar en la página web, en la hoja de estilos CSS configuramos la apariencia de dicha información.



## Objetivos:

### Que los participantes\*:

- Incorporen los conceptos de caja y posicionamiento



## Bloques temáticos:

1. Formas de introducir estilos a un documento HTML
2. Concepto de caja y sus propiedades
3. Posicionamiento de los elementos
4. Ejemplo de implementación de una hoja de estilos CSS
5. Trabajo Práctico



## Consignas para el aprendizaje colaborativo

En esta Unidad los participantes se encontrarán con diferentes tipos de actividades que, en el marco de los fundamentos del MEC\*, los referenciarán a tres comunidades de aprendizaje, que pondremos en funcionamiento en esta instancia de formación, a los efectos de aprovecharlas pedagógicamente:

- Los foros proactivos asociados a cada una de las unidades.
- La Web 2.0.
- Los contextos de desempeño de los participantes.

Es importante que todos los participantes realicen algunas de las actividades sugeridas y compartan en los foros los resultados obtenidos.

Además, también se propondrán reflexiones, notas especiales y vinculaciones a bibliografía y sitios web.

El carácter constructivista y colaborativo del MEC nos exige que todas las actividades realizadas por los participantes sean compartidas en los foros.

*\* El MEC es el modelo de E-learning colaborativo de nuestro Centro.*



## Tomen nota:

Las actividades son opcionales y pueden realizarse en forma individual, pero siempre es deseable que se las realice en equipo, con la finalidad de estimular y favorecer el trabajo colaborativo y el aprendizaje entre pares. Tenga en cuenta que, si bien las actividades son opcionales, su realización es de vital importancia para el logro de los objetivos de aprendizaje de esta instancia de formación. Si su tiempo no le permite realizar todas las actividades, por lo menos realice alguna, es fundamental que lo haga. Si cada uno de los participantes realiza alguna, el foro, que es una instancia clave en este tipo de cursos, tendrá una actividad muy enriquecedora.

Asimismo, también tengan en cuenta cuando trabajen en la Web, que en ella hay de todo, cosas excelentes, muy buenas, buenas, regulares, malas y muy malas. Por eso, es necesario aplicar filtros críticos para que las investigaciones y búsquedas se encaminen a la excelencia. Si tienen dudas con alguno de los datos recolectados, no dejen de consultar al profesor-tutor. También aprovechen en el foro proactivo las opiniones de sus compañeros de curso y colegas.



## 1. Formas de introducir estilos a un documento HTML

Tenemos 3 formas de incluir estilos a un documento HTML:

- En las etiquetas
- En el documento HTML general
- Mediante la inclusión de una hoja de estilos

### En las etiquetas

Antes de comenzar, nos gustaría aclarar que en muy contados casos esta opción es admisible. No es una buena práctica la inclusión de código CSS en la etiqueta ya que dificulta el seguimiento del código en general, es más propenso a errores y como consecuencia el mantenimiento del código se torna dificultoso.

#### Ejemplo:

```
<h1 style="color: yellow;">Mi titulo</h1>
```

Mediante el atributo style se incluye el código CSS. Como se puede apreciar, en caso de que tengamos unas cuantas reglas CSS para aplicar, el código se volvería sumamente engorroso.

### En el documento HTML





```
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <title>Document</title>
8      <style>
9          h1 {
10              color: rgba(255,0,255);
11          }
12      </style>
13 </head>
```

En el documento HTML se incluye en la sección head, todo el código CSS quedará comprendido entre las etiquetas style de apertura y cierre.

Si bien es mucho más prolijo, aún existe una mejor forma de incluir CSS y es la opción 3 que veremos a continuación.

### Inclusión de una hoja de estilos

Esta es la alternativa óptima ya que divide cada lenguaje en un archivo separado (HTML por un lado y CSS por el otro) permitiendo así la reutilización y mejorando el mantenimiento de los respectivos códigos.



```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <title>Document</title>
  <link rel="stylesheet" href="main.css">
</head>
```

Se incluye en el HTML la referencia al archivo que contiene el CSS. También se pueden incluir varias hojas de estilo.

Se utiliza la etiqueta link con 2 atributos:

- rel para indicar que es una hoja de estilos ("stylesheet")
- href para indicar la ruta y nombre del archivo css

```
index.html  # main.css  X
varios > # main.css > h1
1  h1 {
2    color: magenta;
3  }
```

En un archivo con extensión css se escriben todos los estilos.

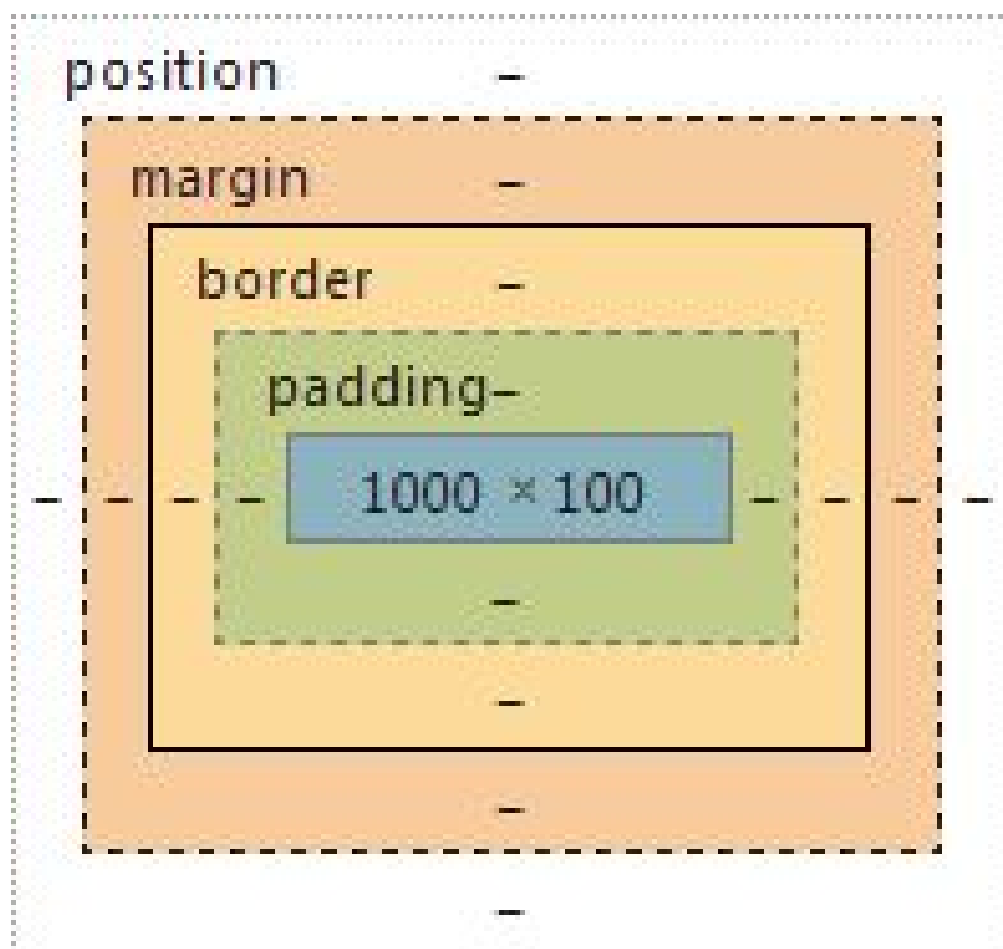


## 2. Concepto de caja y sus propiedades

Cada elemento (digamos cada etiqueta HTML) figurativamente puede representarse con una caja. Esta caja, como una caja de zapatos, posee ciertos espacios y un contenido. El contenido puede ser por ejemplo el texto Mi título contenido en la etiqueta h1

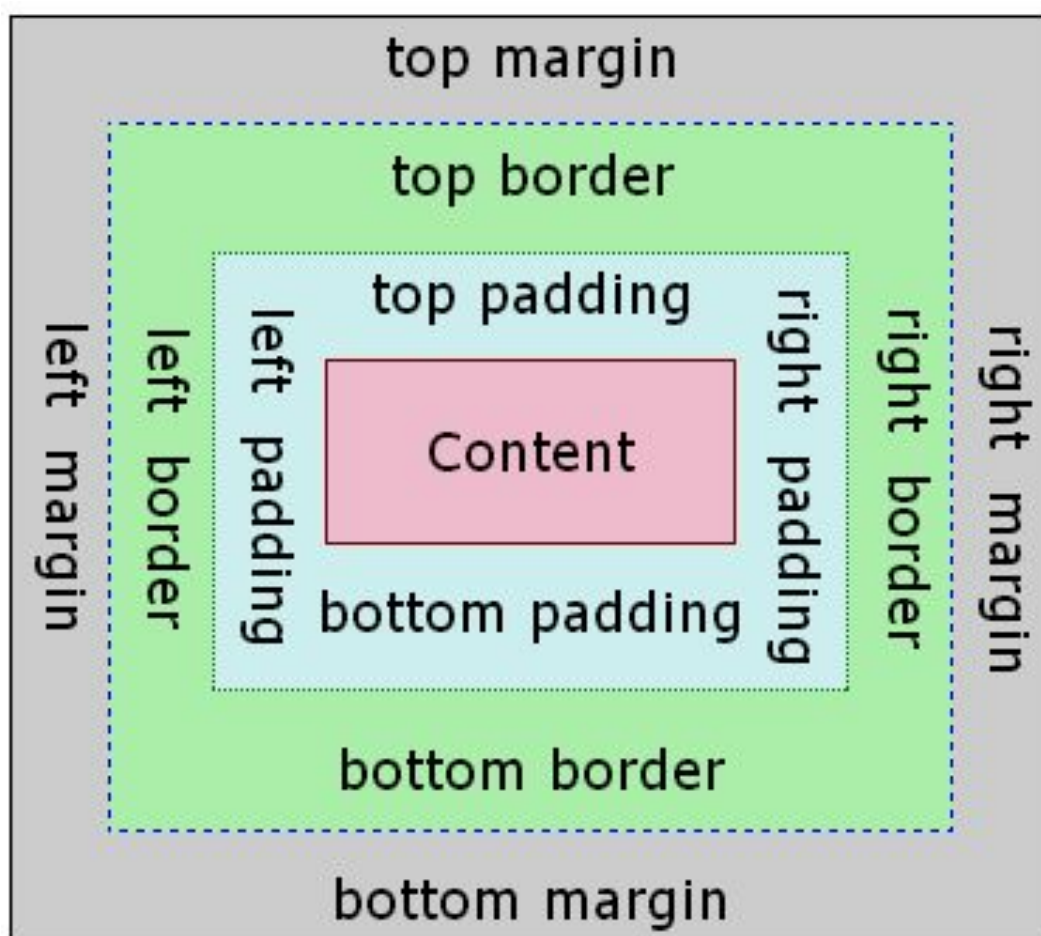
```
<h1>Mi titulo</h1>
```

En cuanto a los espacios, el que queda entre la caja en cuestión y las que se encuentran a su alrededor, se denomina margin. El espacio entre los zapatos y la caja se denomina padding y la caja en sí, el grosor de su cartón es el border.





A su vez, podemos hacer referencia a cada una de las propiedades mencionadas, indicando su ubicación particular: izquierda (left), derecha (right), superior (top), inferior (bottom).



Trabajando desde el archivo css, la sintaxis de cada atributo de la caja tiene unos cuantos variantes. De cada atributo de caja hay una versión expandida y otra resumida. Vamos a verlo con un ejemplo.



## Ejemplo

```
<> index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>New Sorbetes</title>
8      <link rel="stylesheet" href="style.css">
9  </head>
10 <body>
11     <div class="caja_zapato">Sandalias</div>
12     <div class="caja_zapato">Zapatillas</div>
13     <div class="caja_zapato">Botas</div>
14 </body>
15 </html>
```

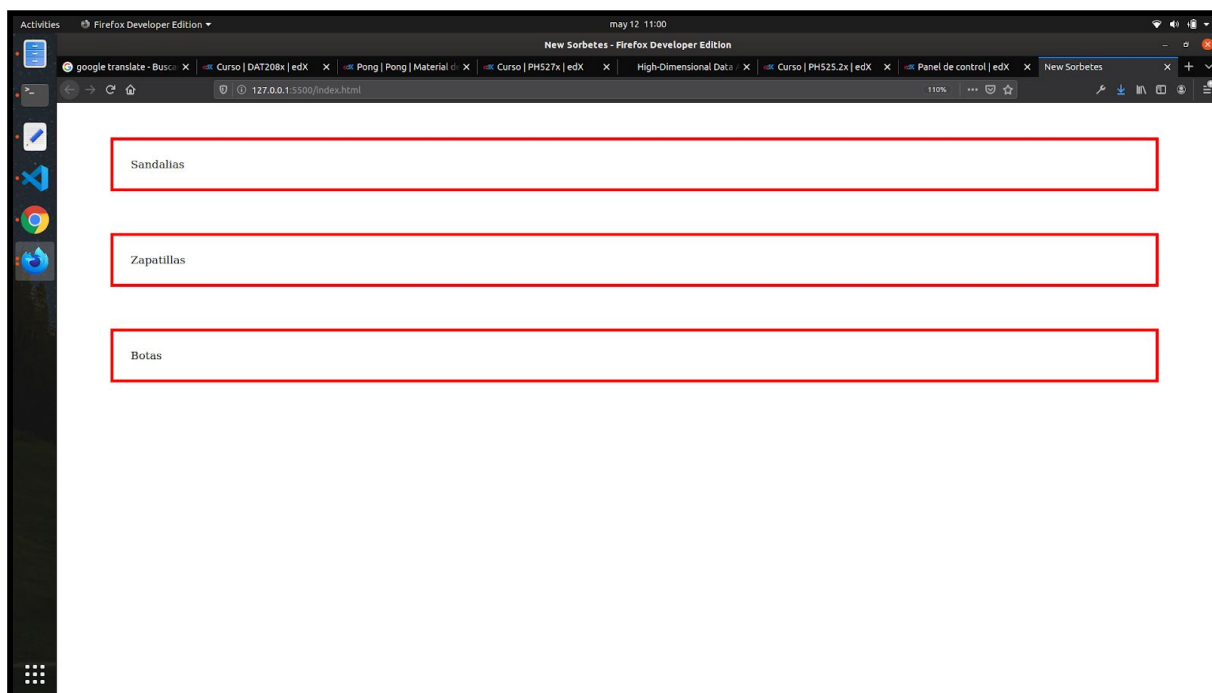
Para las 3 cajas usamos la misma clase ya que todas son cajas de zapatos! En este ejemplo estamos utilizando la inclusión del estilo mediante una hoja de estilos que se llama style.css y que se encuentra en la misma carpeta que el documento html.



```
1  .caja_zapato {  
2      margin-top: 50px;  
3      margin-bottom: 60px;  
4      margin-left: 70px;  
5      margin-right: 80px;  
6  
7      padding-top: 25px;  
8      padding-bottom: 25px;  
9      padding-left: 25px;  
10     padding-right: 25px;  
11  
12     border: 5px solid red;  
13 }
```

Esta hoja de estilo configura la caja de zapatos utilizando la versión detallada de cada atributo, como se puede ver, se especifica cada margin según su posición y del mismo modo el padding. En cuanto al border, utilizamos una versión sintética donde en una única línea indicamos el grosor (5px = 5 pixels), el tipo (solid = sólido, puede ser punteado, etc.) y el color (red).

El resultado final es el siguiente



Tres cajas con margin, border y padding. Un poco más adelante vamos a ver por qué cada caja ocupa tanto espacio (de punta a punta del espacio del navegador).

Si volvemos al css, podemos ver que en el caso del padding en todas las posiciones utilizamos el mismo valor (25px), así que podemos reducir todas esas líneas a tan sólo 1:





```
1  .caja_zapato {  
2      margin-top: 50px;  
3      margin-bottom: 60px;  
4      margin-left: 70px;  
5      margin-right: 80px;  
6  
7      padding: 25px;  
8  
9      border: 5px solid red;  
10 }
```

Como en todas las ubicaciones el padding es exactamente el mismo, se puede usar la versión compacta del atributo. De esta forma se ordena aplicar un padding de 25px arriba, abajo, a la derecha y a la izquierda.

En cuanto al margin, también se puede usar una versión reducida aunque el valor sea diferente:

```
1  .caja_zapato {  
2  
3      margin: 50px 80px 60px 70px;  
4  
5      padding: 25px;  
6  
7      border: 5px solid red;  
8  }
```





De esta forma se pueden indicar todos los márgenes de una sola vez. Al usar esta versión compacta hay que prestar atención a la posición de cada ubicación ya que arranca por el top y se mueve como las agujas del reloj.

```
1 .caja_zapato {  
2  
3     margin: 50px 80px 60px 70px;  
4  
5     padding: 25px;  
6  
7     border: 5px solid red;  
8 }
```

```
1 .caja_zapato {  
2     margin-top: 50px;  
3     margin-bottom: 60px;  
4     margin-left: 70px;  
5     margin-right: 80px;  
6  
7     padding: 25px;  
8  
9     border: 5px solid red;  
10 }
```

El orden es **margin: top right bottom left;**

Para finalizar, se podría haber aplicado la misma compactación para el padding aunque en este caso, donde todos los paddings son del mismo tamaño la compactación elegida en el ejemplo es la óptima.



### 3. Posicionamiento de los elementos

Hay muchas formas de posicionar elementos dentro del diseño de la página pero para comenzar, tenemos que aprender acerca de los tipos de elementos en bloque y en línea.

#### Elementos en bloque y elementos en línea

En nuestro ejemplo anterior, vimos que cada caja de zapatos estaba ocupando el total del espacio horizontal del navegador, en lugar de apegarse al espacio necesario para contener a cada tipo de zapato (el texto del elemento). Esto se debe a que la etiqueta `div` es un elemento del tipo bloque.

Cada elemento html puede ser de tipo bloque (block) lo que indica que ocupa absolutamente todo el espacio disponible, aunque su contenido no lo requiera así o bien puede ser de tipo en línea (inline) lo que implica que sólo va a utilizar el espacio que necesita para su contenido.

Valor	Denominación	Significado	Ejemplo
inline	Elemento en línea	El elemento se coloca en horizontal (ocupa sólo su contenido).	<code>&lt;span&gt;</code>
block	Elemento en bloque	El elemento se coloca en vertical (ocupa todo el ancho disponible).	<code>&lt;div&gt;</code>

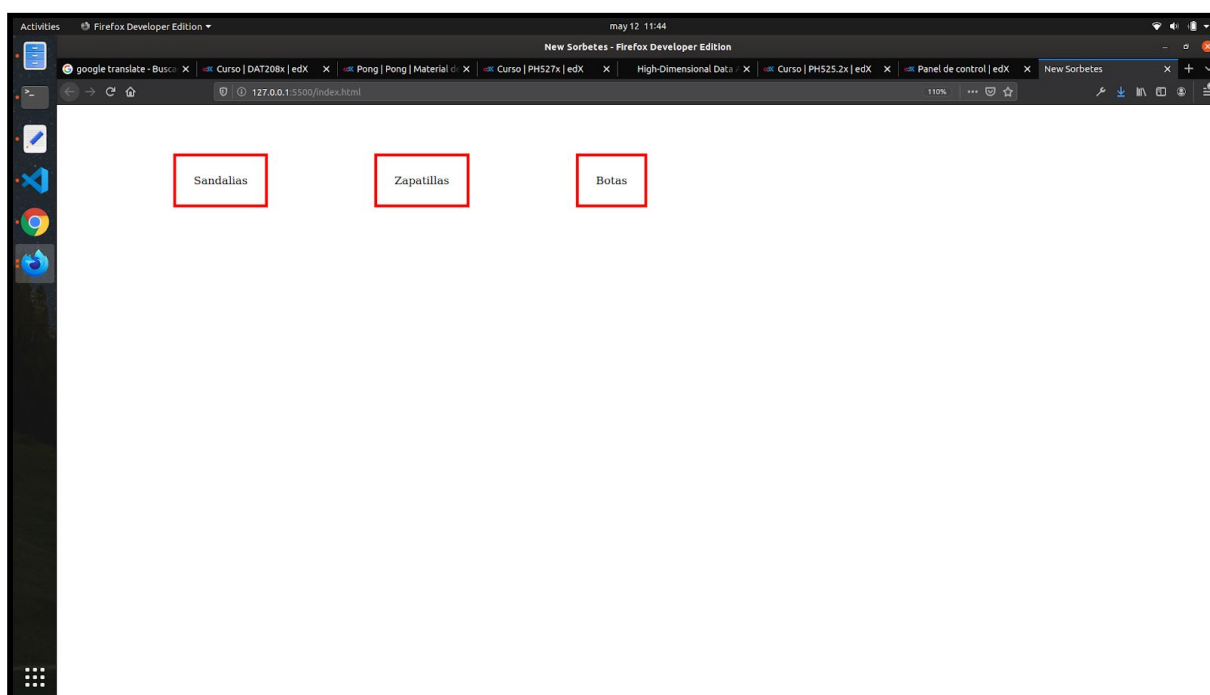
Saber si un elemento es de bloque o en línea es muy importante para poder predecir su comportamiento visual al posicionarlo dentro de la página web.

Vamos a ver qué pasaría si en nuestro ejemplo de las cajas de zapatos, usamos el divisor `<span>` en lugar del divisor `<div>`



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <meta http-equiv="X-UA-Compatible" content="ie=edge">
7   <title>New Sorbetes</title>
8   <link rel="stylesheet" href="style.css">
9 </head>
10 <body>
11   <span class="caja_zapato">Sandalias</span>
12   <span class="caja_zapato">Zapatillas</span>
13   <span class="caja_zapato">Botas</span>
14 </body>
15 </html>
```

Hemos reemplazado los <div> por <span>



En efecto, ahora las cajas se acomodan una al lado de la otra y su tamaño varió para ajustarse al contenido.



Una aclaración: hemos incluido un margin al body para evitar el colapso de los bordes superiores. Pueden buscar información acerca del colapso de bordes si les interesa.

```
1  body {  
2    margin: 100px;  
3  }  
4  
5  .caja_zapato {  
6  
7    margin: 50px 80px 60px 70px;  
8  
9    padding: 25px;  
10  
11    border: 5px solid red;  
12  }
```

Ejemplo de elementos en línea

Los siguientes elementos son "en línea":

- `b`, `big`, `i`, `small`, `tt`
- `abbr`, `acronym`, `cite`, `code`, `dfn`, `em`, `kbd`, `strong`, `samp`, `time`, `var`
- `a`, `bdo`, `br`, `img`, `map`, `object`, `q`, `script`, `span`, `sub`, `sup`
- `button`, `input`, `label`, `select`, `textarea`



En cuanto a elementos del tipo bloque, son la mayoría por lo que podríamos decir que todo el resto de los elementos que no se encuentran en la lista anterior, son en bloque. De todos modos, siempre es bueno buscar en la documentación online de cada elemento sus características.

Concluyendo, la posición de cada elemento html va a estar condicionada por sus valores de caja (bordes, relleno, márgenes) y por su tipo (bloque o en línea).

En cuanto a los valores de caja, son fácilmente modificables mediante CSS con los atributos margin, padding y border en sus versiones compactas o ampliadas.

En lo referido al tipo, existe un atributo que nos permite modificar el tipo de un elemento. Es el atributo display. El atributo display es amplio sin embargo para los efectos de esta sección vamos a aprender cómo modificarlo para que un elemento en bloque se comporte como elemento inline o viceversa.



## 4. Ejemplo de implementación de una hoja de estilos CSS

### Ejemplo 1

Vamos a cambiarle el tipo a los elemento span que como vimos, son nativamente elementos inline.

Utilizamos el mismo html que en los ejemplos anteriores.

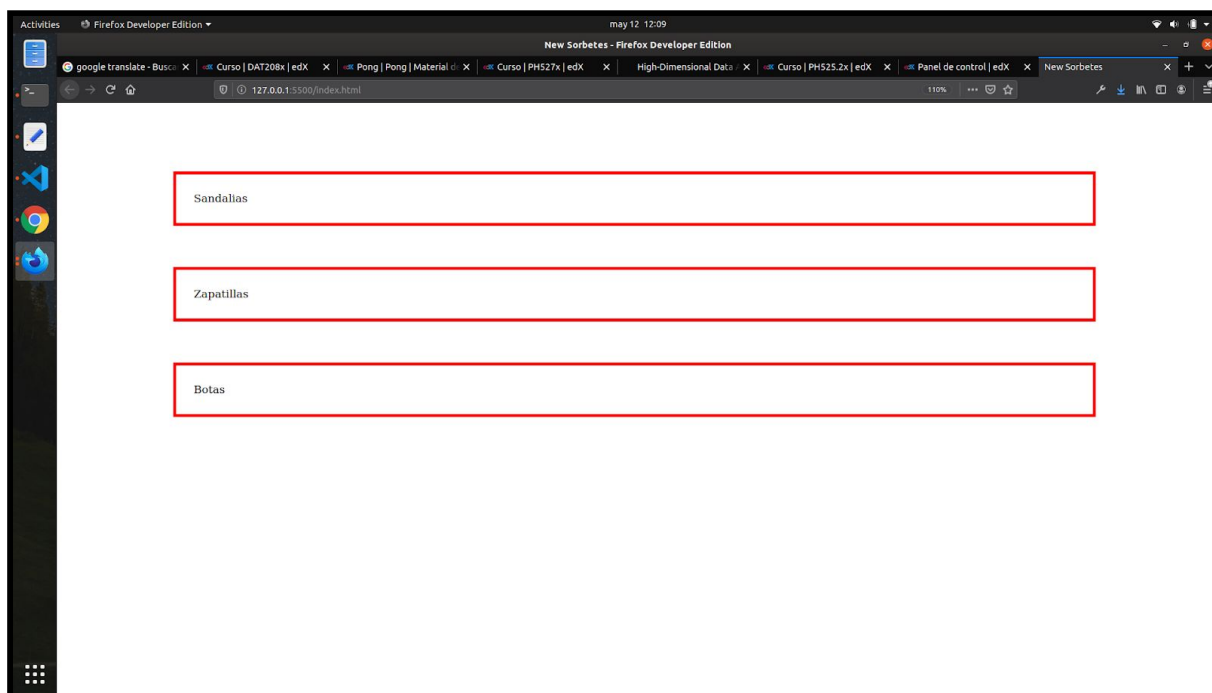
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>New Sorbetes</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <span class="caja_zapato">Sandalias</span>
  <span class="caja_zapato">Zapatillas</span>
  <span class="caja_zapato">Botas</span>
</body>
</html>
```

En el CSS vamos a aplicarle a la clase caja\_zapato un display: block;



```
1  body {  
2    margin: 100px;  
3  }  
4  
5  .caja_zapato {  
6  
7    margin: 50px 80px 60px 70px;  
8  
9    padding: 25px;  
10  
11    border: 5px solid red;  
12  
13    display: block;  
14  }
```

Así cambiamos el comportamiento de la clase `caja_zapatos` aplicada a los `span`, por consiguiente, cambia el comportamiento de dichos `span` que empiezan a dibujarse en la página como elementos de bloque.



## Ejemplo 2

Ahora vamos a modificar el comportamiento de la etiqueta div, que es nativamente un elemento de bloque para que sea inline.

Trabajamos con el siguiente html





```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <meta http-equiv="X-UA-Compatible" content="ie=edge">
7   <title>New Sorbetes</title>
8   <link rel="stylesheet" href="style.css">
9 </head>
10 <body>
11   <div class="caja_zapato">Sandalias</div>
12   <div class="caja_zapato">Zapatillas</div>
13   <div class="caja_zapato">Botas</div>
14 </body>
15 </html>
```

Y el CSS ahora tiene display: inline; para la clase caja\_zapato.

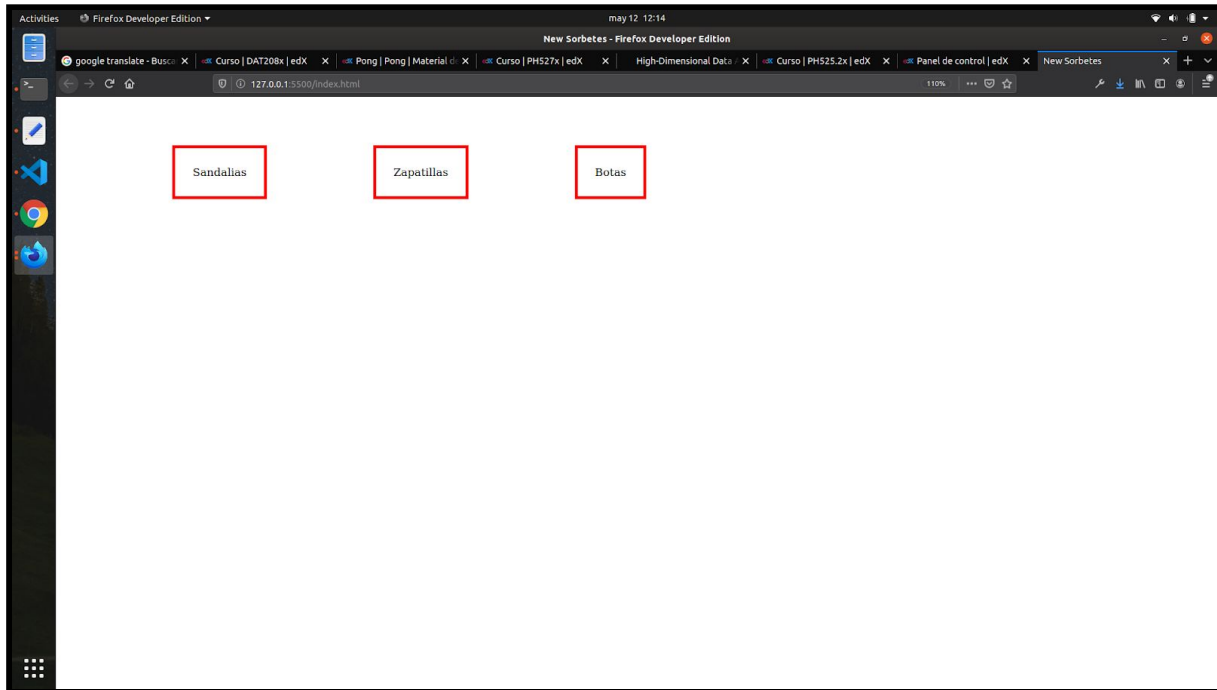
```
1 body {
2   margin: 100px;
3 }
4
5 .caja_zapato {
6
7   margin: 50px 80px 60px 70px;
8
9   padding: 25px;
10
11   border: 5px solid red;
12
13   display: inline;
14 }
```

El resultado final es

Centro de e-Learning SCEU UTN - BA.

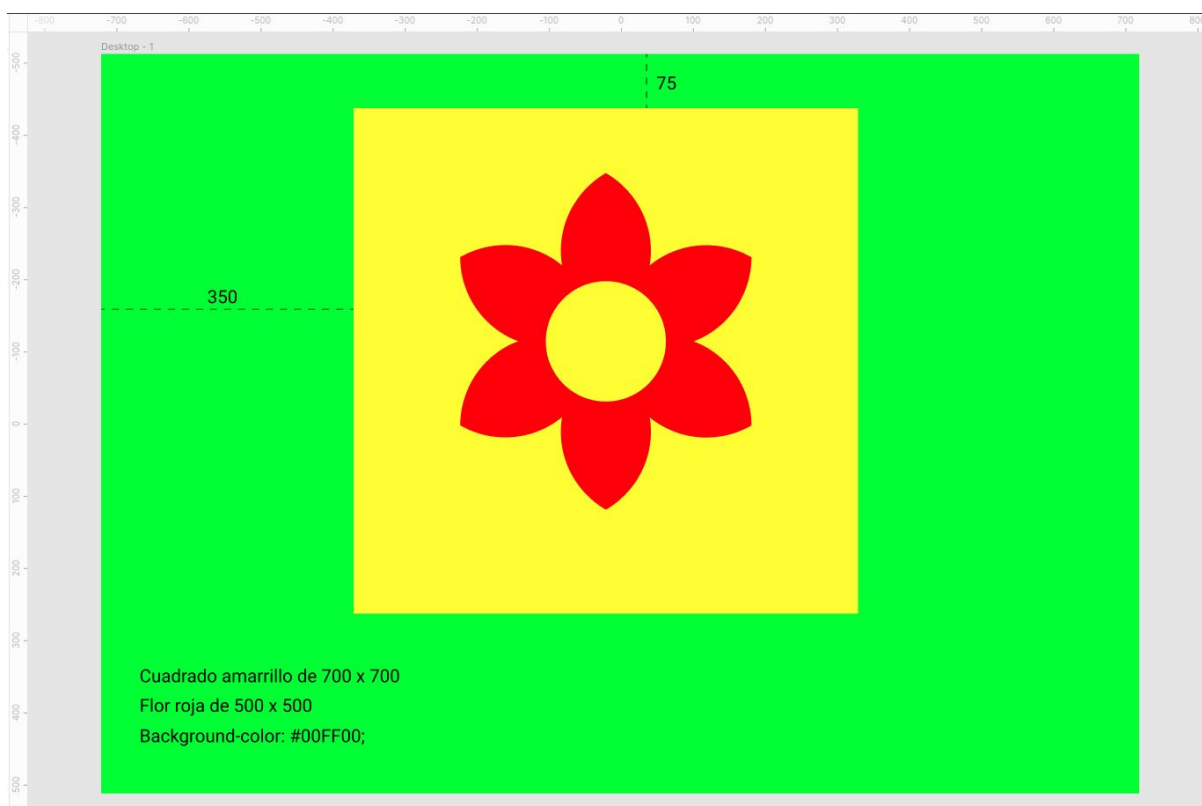
Medrano 951 2do piso (1179) // Tel. +54 11 4867 7589 / Fax +54 11 4032 0148

[www.sceu.frba.utn.edu.ar/e-learning](http://www.sceu.frba.utn.edu.ar/e-learning)





## 5. Trabajo Práctico



Crear el archivo index.html y el archivo style.css para conseguir una página como la que se muestra en la imagen.

La flor pueden descargarla de <https://material.io/resources/icons/?icon=extension&style=baseline>

Debido a que estas páginas son dinámicas, si no encuentran la misma flor, utilicen cualquier otra imagen de ese enlace.



## Bibliografía utilizada y sugerida

World Wide Web Consortium. (n.d.) Recuperado de <https://www.w3.org/>

w3schools.com. (n.d.) Recuperado de <https://www.w3schools.com/>

MDN web docs CSS. (n.d.) Recuperado de:

<https://developer.mozilla.org/es/docs/Web/CSS>

MDN web docs HTML5. (n.d.) Recuperado de:

<https://developer.mozilla.org/es/docs/HTML/HTML5>



## Lo que vimos:

- Conceptos de caja y posicionamiento en CSS.



## Lo que viene:

- Conceptos intermedios de CSS.

