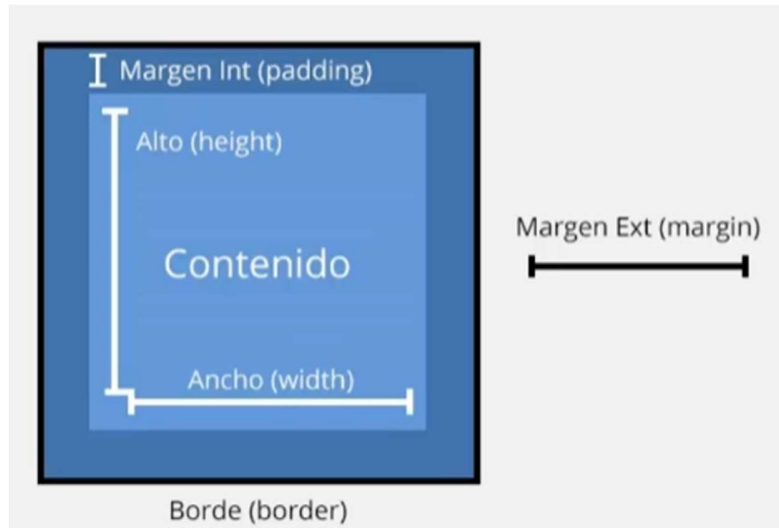


00 Modelo de caja

Para entender el modelo de caja, es necesario familiarizarse con algunas propiedades, vamos a ver la siguiente imagen:



Una “caja” en la que tengamos nuestro contenido, puede tener un ancho (width) y un alto (height), adicionalmente tendrá un margen interno (padding) y un margen externo (margin). El margen interno es el espacio que va entre el contenido de la caja y el borde (border) de la misma. El margen externo es el espacio que hay entre el borde y algún otro elemento externo.

Vamos a construir lo siguiente:

```
<> index.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      <link rel="stylesheet" href="styles/styles.css">
7      <title>Modelo de Caja</title>
8  </head>
9  <body>
10     <header>
11         <h2>Modelo de caja</h2>
12     </header>
13     <main>
14         <div class="box"></div>
15         <div class="box"></div>
16     </main>
17 </body>
18 </html>
19
```

La previsualización de nuestra página nos mostrará algo como:

Modelo de caja

Podemos observar que, aparte del título, no se logra visualizar nada más, esto es porque las cajas no tienen contenido, pero tampoco tienen una altura o un ancho. Para empezar a observar algo, vamos a agregar propiedades a la clase box, vamos a colocar un color de fondo, alto y ancho:

```
1 .box {  
2   background: #dd1c1a;  
3   width: 200px;  
4   height: 200px;  
5 }
```

Modelo de caja



El resultado muestra como si tuviésemos una sola caja, pero realmente tenemos dos, lo que pasa es que están juntas y no se ve espacio entre ellas.

Para corregir esto, podemos agregar el atributo margin. Este atributo lo podemos usar de varias maneras:

```
margin: 20px; /* Aplica margen en todos los lados */  
margin: 10px 20px; /* Aplica 10px de margen arriba y abajo y 20px a los lados */  
margin: 5px 5px 10px 10px; /* Especifica uno a uno las margenes */  
margin-top: 10px; /* Margen en la parte superior */  
margin-left: 10px; /* Margen en la parte izquierda */  
margin-right: 10px; /* Margen en la parte derecha */  
margin-bottom: 10px; /* Margen en la parte inferior */
```

Ahora con un margen de 20 px a todos los lados tendremos lo siguiente:

Modelo de caja



Si ahora colocamos algo de texto en cada caja:

```
<main>
  <div class="box">
    Lorem ipsum dolor sit amet consectetur adipisicing elit.
    Eius atque sit aut eligendi perspiciatis quaerat temporibus,
    esse rerum tempora tempore fugit dicta non, accusamus magnam nulla sequi
    ullam? Quibusdam, odio.
  </div>
  <div class="box">
    Lorem ipsum dolor sit amet consectetur adipisicing elit.
    Eius atque sit aut eligendi perspiciatis quaerat temporibus,
    esse rerum tempora tempore fugit dicta non, accusamus magnam nulla sequi
    ullam? Quibusdam, odio.
  </div>
</main>
```

Lorem ipsum dolor sit amet
consectetur adipisicing elit.
Eius atque sit aut eligendi
perspiciatis quaerat
temporibus, esse rerum
tempora tempore fugit dicta
non, accusamus magnam
nulla sequi ullam?
Quibusdam, odio.

Adicionalmente colocamos algunos atributos al body para que veamos mejor el texto:

```
body {
  color: #fff;
  font-family: Arial;
  line-height: 22px;
}
```

Lorem ipsum dolor sit amet
consectetur adipisicing elit.
Eius atque sit aut eligendi
perspiciatis quaerat
temporibus, esse rerum
tempora tempore fugit dicta
non, accusamus magnam
nulla sequi ullam?
Quibusdam, odio.

El line-height da un espacio entre líneas. Ahora colocamos un padding a nuestra clase box:

```
padding: 20px;
```

Ahora veamos el efecto que causó agregar el padding:

Lorem ipsum dolor sit amet
consectetur adipisicing elit.
Eius atque sit aut eligendi
perspiciatis quaerat
temporibus, esse rerum
tempora tempore fugit dicta
non, accusamus magnam
nulla sequi ullam?
Quibusdam, odio.

Con eso ya podemos ver entonces que el margin es el espacio de la caja a los elementos externos, en este caso, la otra caja, mientras que el padding es la margen interna, es decir, del borde al texto. Ahora, ¿qué pasa cuando el texto contenido en la caja, supera la altura que definimos?

Lorem ipsum dolor sit amet
consectetur adipisicing elit.
Eius atque sit aut eligendi
perspiciatis quaerat
temporibus, esse rerum
tempora tempore fugit dicta
non, accusamus magnam
nulla sequi ullam?
Quibusdam, odio. atque sit
aut eligendi perspiciatis
quaerat temporibus, esse
rerum tempora tempore

Lorem ipsum dolor sit amet
consectetur adipisicing elit.
Eius atque sit aut eligendi
perspiciatis quaerat
temporibus, esse rerum
tempora tempore fugit dicta
non, accusamus magnam
nulla sequi ullam?
Quibusdam, odio.

Entonces quitamos el atributo height, para que la altura sea automática.

Elementos en bloque y elementos en línea

Vamos a hacer algunas modificaciones para entender la diferencia entre elementos en línea y en bloque.

```
<section>
  <h1>Soy un titulo</h1>
  <p>Soy un parrafo <a href="#">Soy un link dentro del parrafo</a></p>
  <div>Soy un contenedor div</div>
  <a href="#">Soy un enlace</a>
  <a href="#">Soy otro enlace</a>
</section>
```

Agregamos al principio una sección con un título (h1) un párrafo (p) que interiormente contiene un enlace (a), después detenemos un contenedor (div) y otro par de enlaces (a).

```
body {
  font-family:Arial;
  line-height: 22px;
}

h1, p, div {
  background: #ccc;
}

a {
  background: #000;
  color: #fff;
}
```

Al body le quitamos el color blanco para poder ver mejor el texto y entender lo que pasa. Agregamos un color de fondo gris claro a los elementos h1, p y div. Al elemento a, le agregamos un color de fondo negro y un texto blanco:

Soy un titulo

Soy un parrafo Soy un link dentro del parrafo

Soy un contenedor div

Soy un enlace Soy otro enlace

Los elementos **h1**, **p** y **div**, son elementos en bloque, mientras que el elemento **a**, es un elemento en línea. ¿Esto qué significa?, al colocar el color de fondo sobre los elementos, nos damos cuenta de que los elementos de bloque ocupan el 100% del ancho de la página, mientras que los elementos en línea solo ocupan exactamente el espacio de su contenido.

Vamos a agregar a los elementos en bloque, una margen interna (padding) de 10px y un ancho (width) de 500px.

```
h1, p, div {
  background: #ccc;
  padding: 10px;
  width: 500px;
}
```

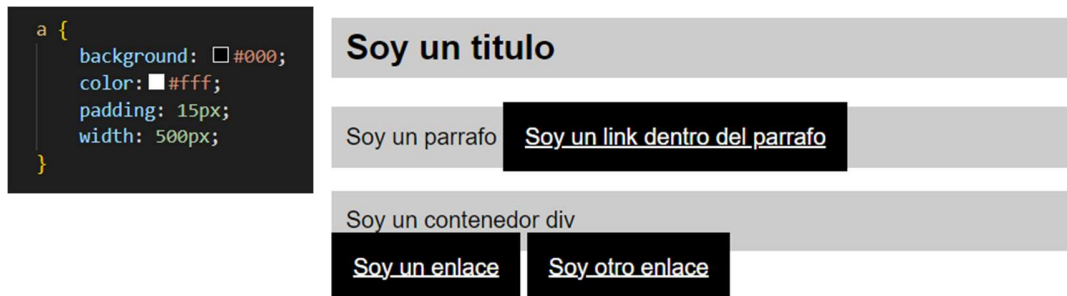
Soy un titulo

Soy un parrafo Soy un link dentro del parrafo

Soy un contenedor div

Soy un enlace Soy otro enlace

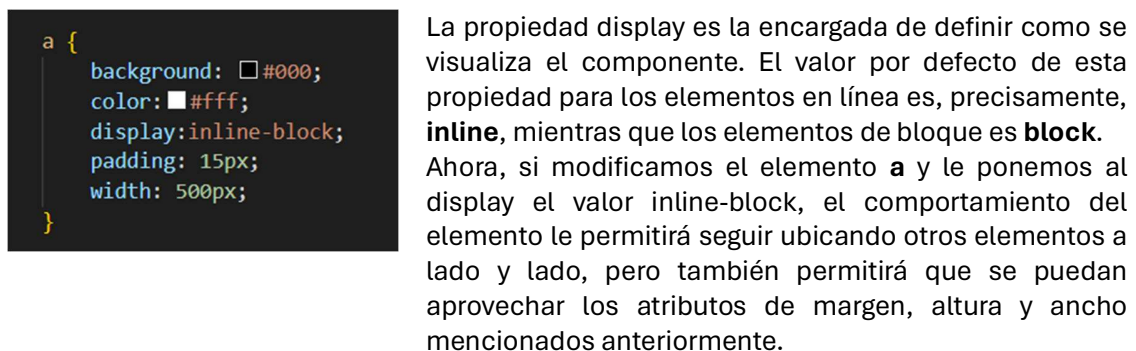
Claramente podemos ver la margen interna y también vemos como se ha modificado el ancho de los elementos, sin embargo, pese a tener espacio suficiente al lado, estos elementos no se acomodan uno junto al otro. Esta es otra característica importante de los elementos en bloque, aunque eso se puede modificar. Tratemos de modificar también la margen interna y el ancho del elemento **a**:



Notemos que el ancho no fue modificado y, aunque la margen interna sí se modifica, esta se superpone con sus elementos adyacentes. Esto es porque a los elementos en línea no se les puede algunos de sus atributos como altura, ancho, margen interna. Para que estos elementos puedan tener este comportamiento, debemos hacer un proceso adicional para convertirlos.

¿Cómo saber si un elemento es de tipo bloque o de tipo en línea? Así como lo realizamos en este ejemplo, una forma simple es ponerle al elemento un color de fondo que resalte, si ocupa el 100% del espacio disponible, es de tipo bloque, si ocupa exactamente el espacio de su contenido, es de tipo línea.

Vamos ahora a modificar el comportamiento del elemento **a** cambiando uno de sus atributos:



Ahora si vemos el efecto del width y del padding sin sobreponerse.

Si aplicamos la misma propiedad a los elementos en bloque, observaremos que ya los elementos se pueden posicionar uno a continuación del otro:

