

UD 03 ELABORACIÓN DE HOJAS DE ESTILO

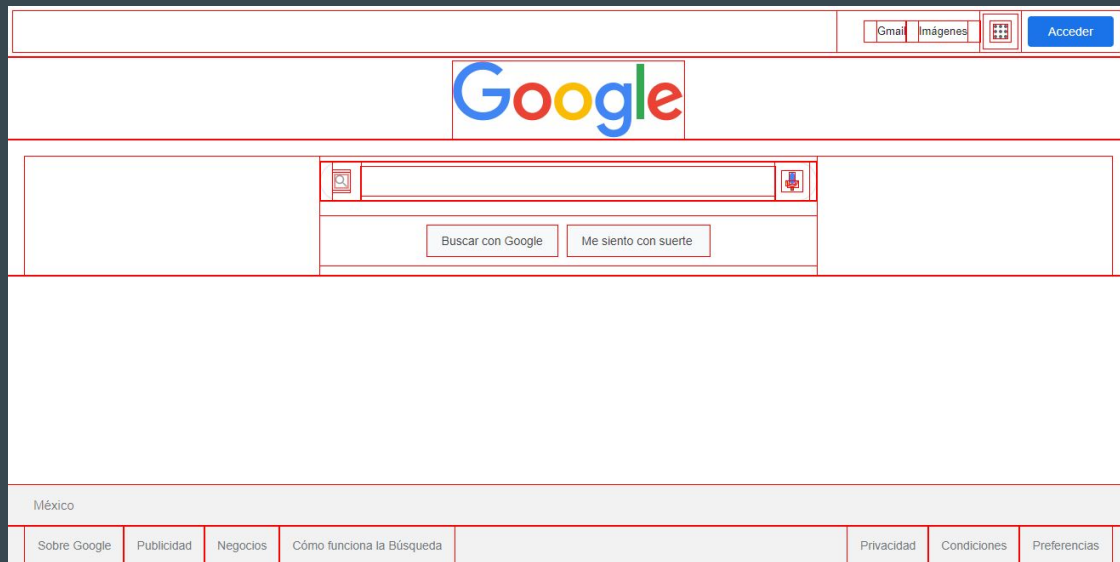
...

3.4 Modelo de cajas y Display

Modelo de cajas

Como ya hemos visto, cada elemento HTML en una página web es una caja rectangular, esta es la forma en que se representan todos los elementos.

Todos los elementos en HTML por defecto son rectangulares ya que internamente el navegador dibuja un rectángulo.



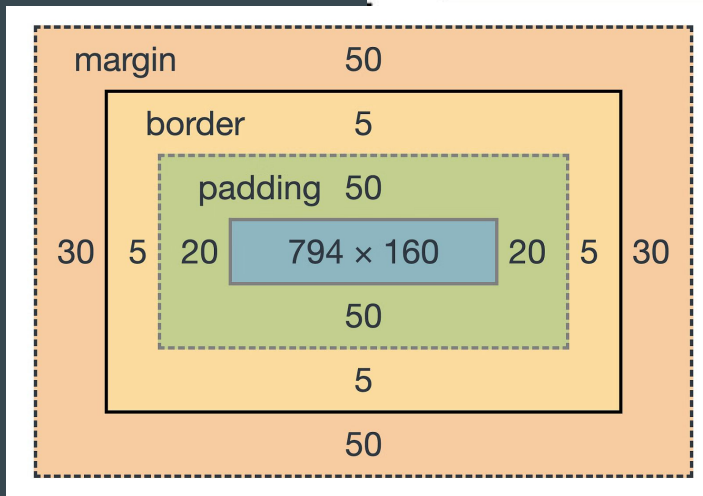
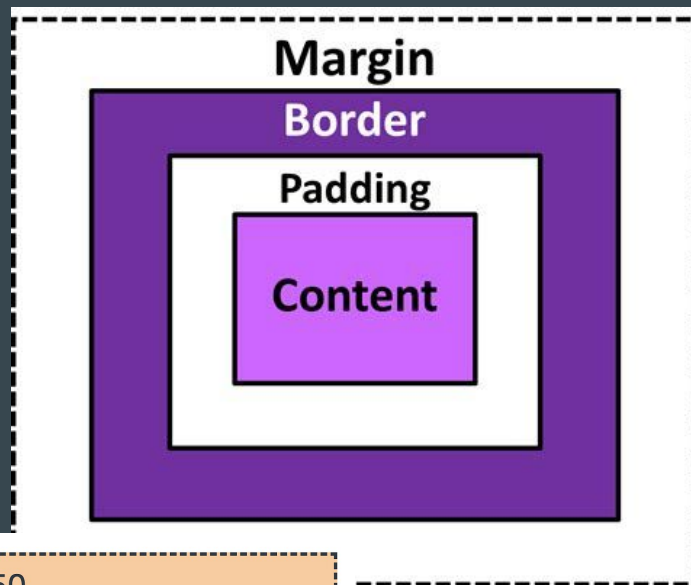
```
* {  
    outline: 1px solid red;  
}
```

Modelo de cajas

A su vez, cada elemento HTML esta formado por 4 áreas:

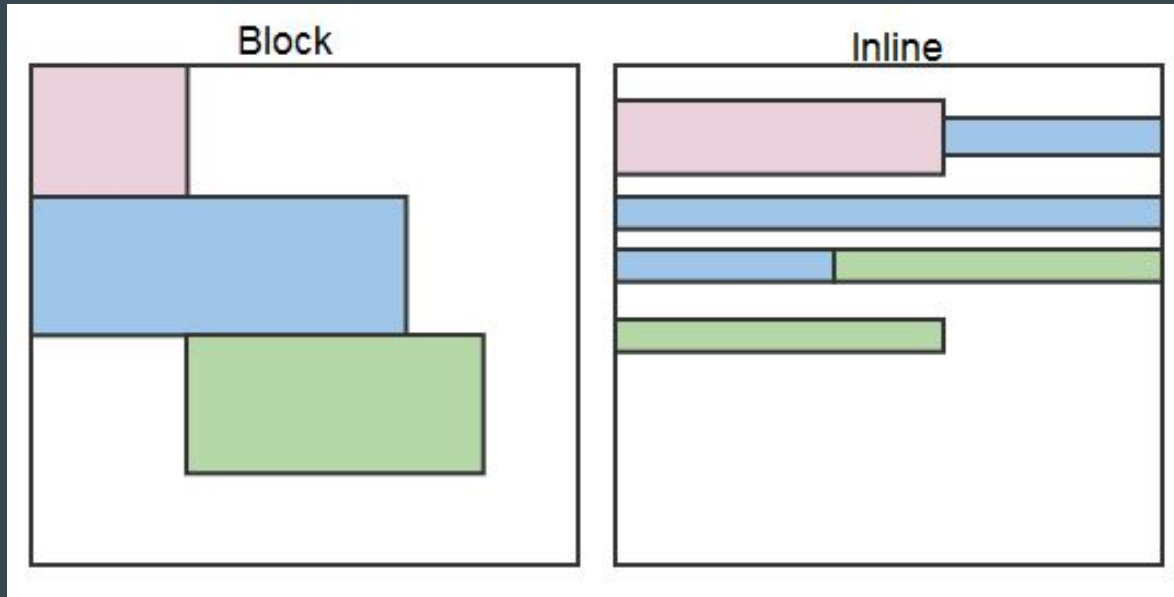
- Content
- Padding
- Border
- Margin.

Estas áreas se apilan para formar el modelo de caja (box model).



Modelo de cajas

Como ya mencionamos cada elemento HTML es una caja, pero hay dos tipos básicos de cajas: **Block boxes** e **Inline boxes**.



Modelo de cajas: Comportamiento inline o en línea

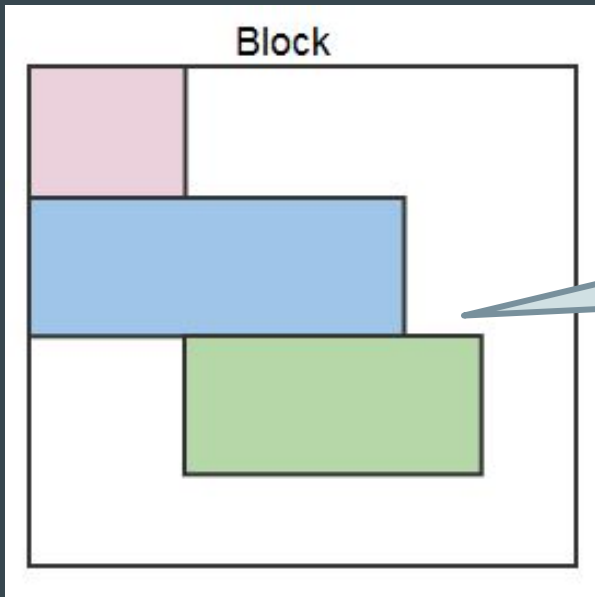
Las cajas en línea, por defecto, toman el espacio según su contenido

```
<h1>SPAN</h1>  
  <span>1- Lorem ipsum dolor sit amet</span>  
  <span>2- Consectetur adipisicing elit. </span>  
  <span>3- Aut optio autem exercitationem consequ  
  <span>4- Dolores tenetur laborum eligendi conse
```



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.

Modelo de cajas: Comportamiento **block** o de bloque



Las cajas de bloque, por defecto, ocupan todo el espacio a lo ancho del contenedor.

```
<h1>PÁRRAFOS</h1>
<p>1- Lorem ipsum dolor sit amet</p>
<p>2- Consectetur adipisicing elit. </p>
<p>3- Aut optio autem exercitationem cons
<p>4- Dolores tenetur laborum eligendi co
```

`<address>`, `<article>`, `<aside>`, `<audio>`, `<blockquote>`, `<canvas>`, `<dd>`, `<div>`, `<dl>`,
`<fieldset>`, `<figcaption>`, `<figure>`, `<footer>`, `<form>`, `<header>`, `<hgroup>`, `<hr>`,
``, `<main>`, `<nav>`, `<noscript>`, ``, `<output>`, `<p>`, `<pre>`, `<section>`, `<table>`,
`<tfoot>`, ``, `<video>`, `<h1>`... `<h6>`.

La propiedad Display

La propiedad ***display*** se usa para especificar como se muestra un elemento en una página web.

Cada elemento HTML tiene su propio valor por defecto (***inline*** o ***block*** en la mayoría de los casos).

Usaremos display para modificar ese comportamiento por defecto.

Display: inline

- Los elementos en línea no admiten dimensiones (width, height) pero sí admiten margin, padding y border.
- Solo ocupan el tamaño que marque su contenido, es decir, aunque quieras declarar un ancho y alto a un elemento de línea no tendrá efecto.
- Si hay varios elementos en línea estarán colocados de izquierda a derecha (uno al lado de otro).

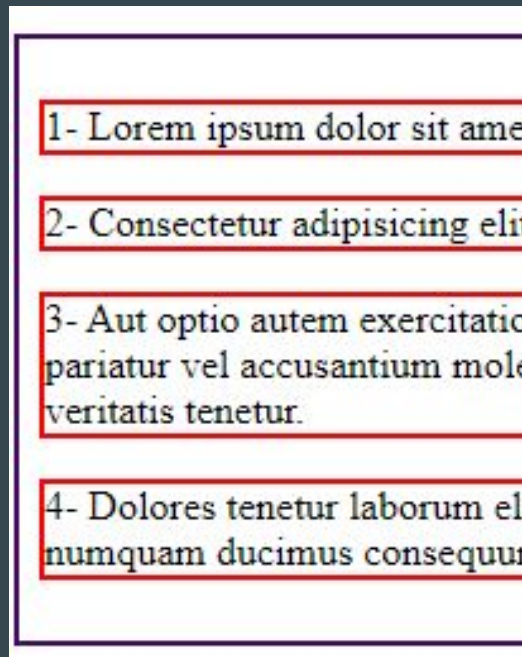
1- Lorem ipsum dolor sit amet 2- Consectetur adipisicing elit. 3- Aut optio autem exercitatio
tenetur laborum eligendi consequatur numquam ducimus consequuntur rem dicta!

¡Practiquemos!

Display: block

- Un elemento de bloque es un elemento que ocupa el ancho máximo disponible. Tal y como dice la documentación de la [MDN](#):

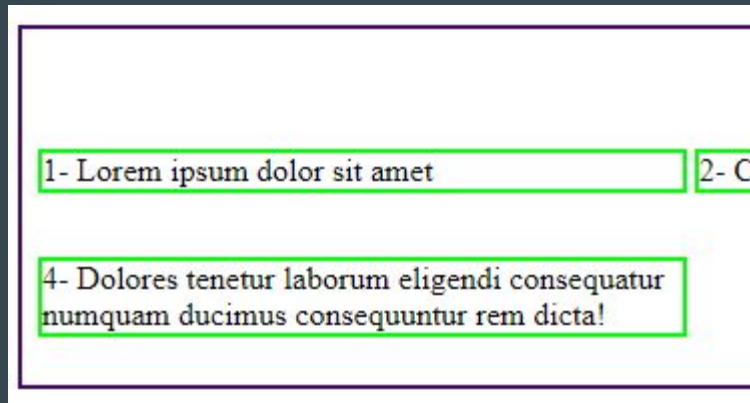
“De manera predeterminada, el contenido de un elemento de nivel de bloque es el 100% del ancho de su elemento padre y su altura viene determinada por su contenido.”



¡Practiquemos!

Display: inline-block

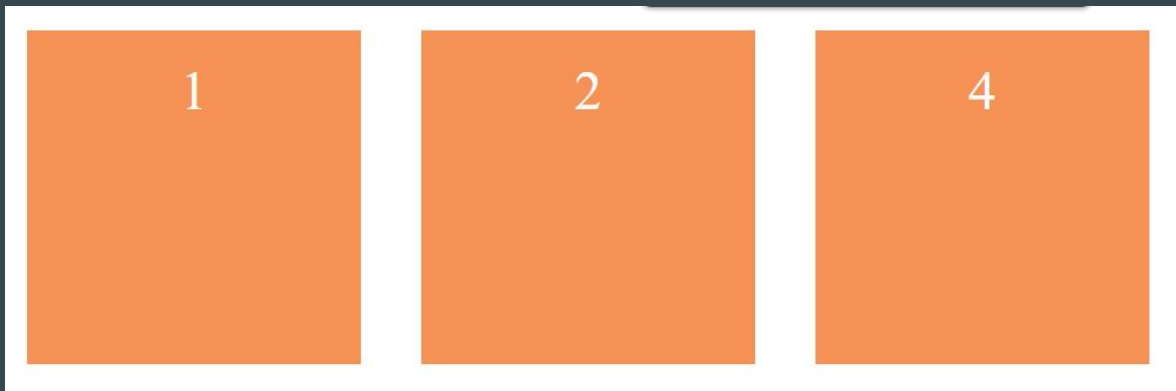
- Es una combinación entre block e inline.
- Los elementos con el valor inline-block admiten dimensiones pero todavía son elementos de línea, es decir se colocarán uno al lado de otro.



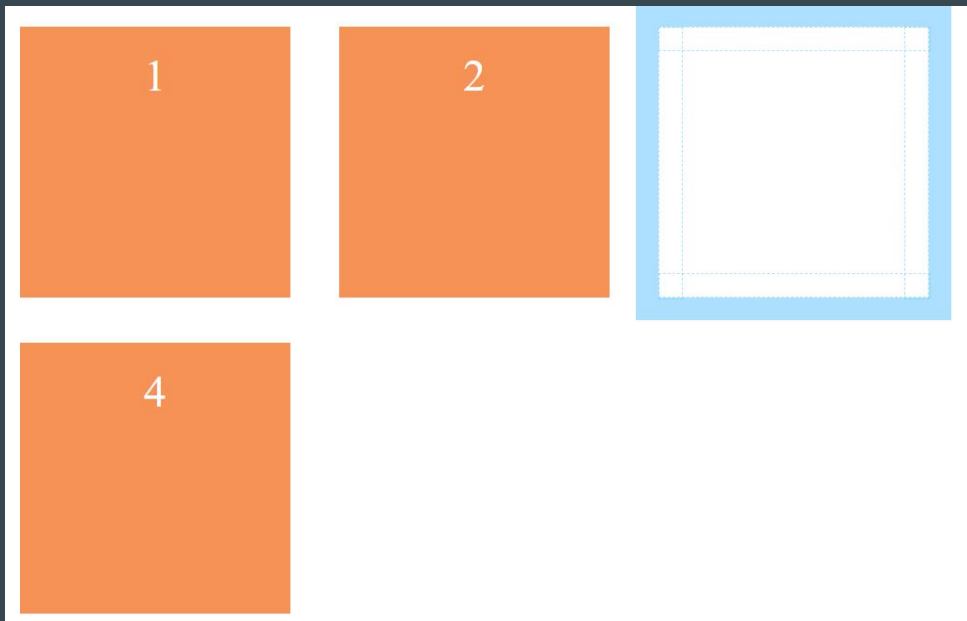
¡Practiquemos!

Display: none

- El valor **none** “elimina” un elemento, por lo que no ocupará ningún espacio.
- El elemento estará oculto y la página será desplegada como si el elemento no estuviera allí. (Desaparece de la estructura **DOM**)



Display: none vs visibility:hidden



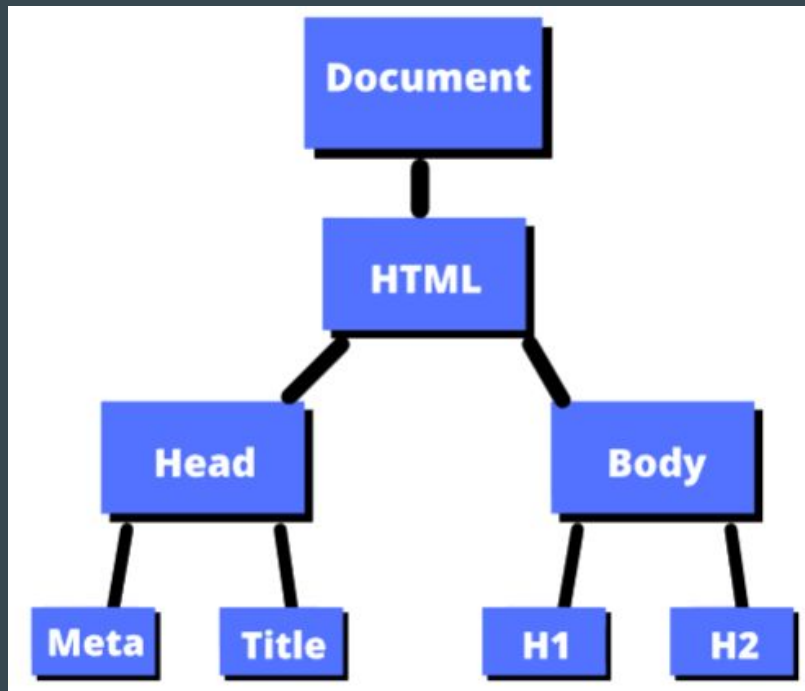
¡Practiquemos!

- El valor ***none*** como hemos dicho hace desaparecer el elemento de la estructura de la web.
- Sin embargo la propiedad `visibility`, con su valor en `hidden`, oculta el elemento, pero este sigue “existiendo”, sigue ocupando su espacio, (no desaparece del DOM).

¿Pero qué es el DOM? (Document Object Model)

El **DOM** no es más que una representación en árbol en la que cada rama representa un elemento de HTML.

La idea de representarlo como un árbol viene de buscar representar la jerarquía de nuestros documentos, es decir, quiénes son hijos de quién, cuáles son hermanos, etc.



¿Pero qué es el DOM?

```
<html>
<head>
  <title>Title</title>
</head>
<body>
  <div></div>
  <div>
    <p>Párrafo 1</p>
    <p>Párrafo 2</p>
  </div>
</body>
</html>
```

HTML

DOM

