

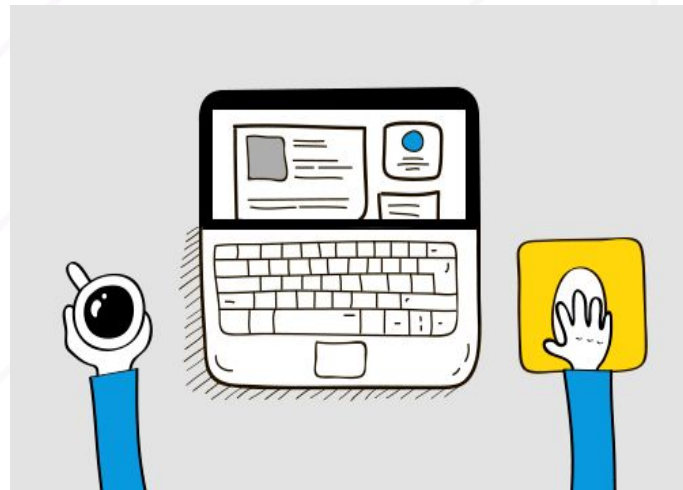
BOX MODEL, FLOAT Y POSITION

DEV.F
DESARROLLAMOS(PERSONAS);

dev

Objetivos

- ¿Qué es el modelo de caja?
- Tipos básicos del modelo de caja.
- ¿Qué es float en CSS?
- ¿Qué es position en CSS?



The logo consists of the text 'DEV.F.' in a bold, white, sans-serif font. The 'F' is stylized with three small squares at its top right corner. The logo is centered within a dark blue diamond shape.

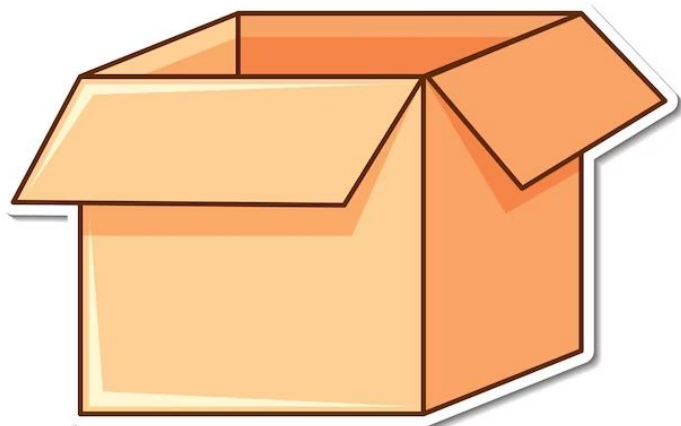
DEV.F.

**¿Que es el
modelo de caja?**

¿Que es el modelo de caja?

El modelo de caja (Box Model en inglés) es un concepto importante en diseño web que describe cómo se estructura y muestra el contenido en una página web.

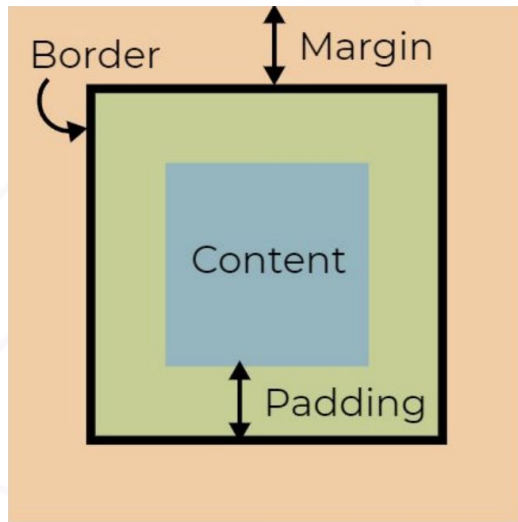
Lo primero que debemos de entender, es que cada elemento que definimos en un documento HTML se mostrará en el navegador como una caja rectangular, esta es la forma en que se representan todos los elementos, no existen elementos triangulares, redondos, poligonales etc.



¿Que es el modelo de caja?

El navegador renderiza cada elemento de la página web como un elemento rectangular y con CSS nosotros definimos las propiedades como el tamaño, el color, el margen, la separación entre otras cajas, etc.

Entonces cada elemento HTML está formado por 4 áreas: content, padding, border y margin. Estas áreas se apilan para formar el modelo de caja (box model).



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

¿Que es el modelo de caja?

HolaBoxModel

HolaBoxModel

HolaBoxModel

HolaBoxModel

■ Content

■ Border

■ Padding

■ Margin



<https://codepen.io/carolineartz/pen/ogVXZj>



The logo consists of the text 'DEV.F.' in a bold, white, sans-serif font. The 'F' is stylized with three small squares at its base. The logo is centered within a dark purple diamond shape.

DEV.F.

Tipos básicos del modelo de caja

Tipos básicos del modelo de caja

Como ya lo mencionamos anteriormente cada elemento HTML es una caja, pero hay dos tipos básicos de cajas:

- Block boxes.
- Inline boxes.

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

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

A dichos tipos se le llama “display”.

block vs inline

The diagram illustrates the difference between block and inline-block display types. It is divided into two sections. The top section, titled 'block vs inline', shows a code snippet on the left:

```
div { display: block }
```

 and a visual representation on the right of three yellow rectangular boxes stacked vertically, each taking up the full width of the container. The bottom section shows a code snippet on the left:

```
div { display: inline-block }
```

 and a visual representation on the right of six green rectangular boxes arranged in a 3x2 grid, where each box only takes up the space of its content.

Tipos básicos del modelo de caja

Por default cada elemento html tiene asignado un tipo de display.

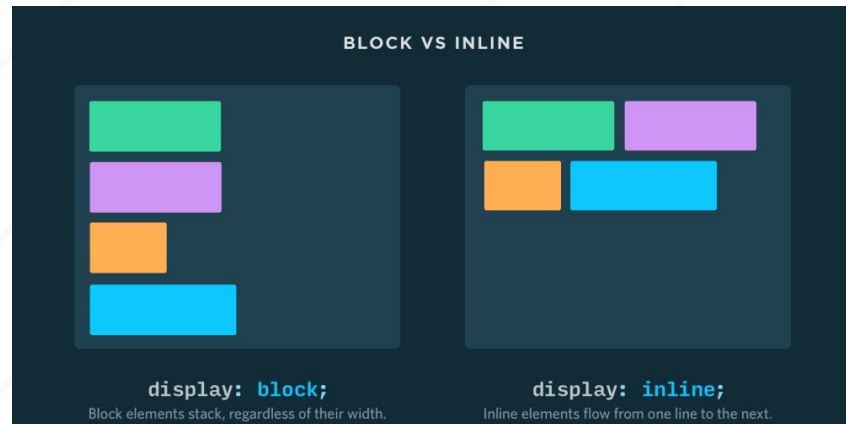
Sin embargo, posteriormente lo podemos cambiar desde una hoja de estilos.

<input checked="" type="checkbox"/> experimental	<input checked="" type="checkbox"/> meta	<input checked="" type="checkbox"/> self-closing	<input checked="" type="checkbox"/> inline	<input checked="" type="checkbox"/> block
a	meta	self-closing	inline	block
abbr	meta	self-closing	inline	block
address	meta	self-closing	inline	block
area	meta	self-closing	inline	block
article	meta	self-closing	inline	block
aside	meta	self-closing	inline	block
audio	meta	self-closing	inline	block
b	meta	self-closing	inline	block
base	meta	self-closing	inline	block
bdi	meta	self-closing	inline	block
bdo	meta	self-closing	inline	block
blockquote	meta	self-closing	inline	block
body	meta	self-closing	inline	block

Tipos básicos del modelo de caja

Nota: los elementos en línea no admiten dimensiones (width, height) pero sí admiten margin, padding y border, aunque quieras declarar un ancho y alto a un elemento de línea no podrá tener efecto ya que estos elementos no aceptan dimensiones

<https://htmlreference.io>



Tipos básicos del modelo de caja

inline

inline

block

inline-block

inline

inline-block

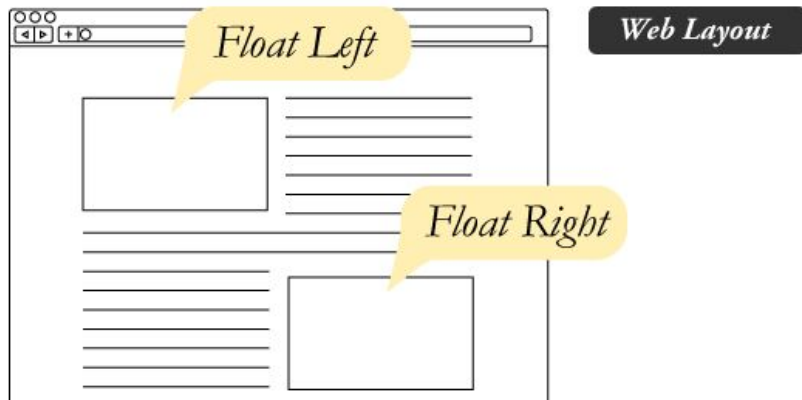
block

DEV.F

¿Qué es float en CSS?



¿Qué es float en CSS?



En CSS, float es una propiedad que se utiliza para controlar cómo se alinean los elementos dentro de un contenedor.

Cuando aplicas la propiedad float a un elemento, ese elemento se "flota" hacia la izquierda o hacia la derecha dentro de su contenedor, y otros elementos se ajustan a su alrededor.

DEV.F

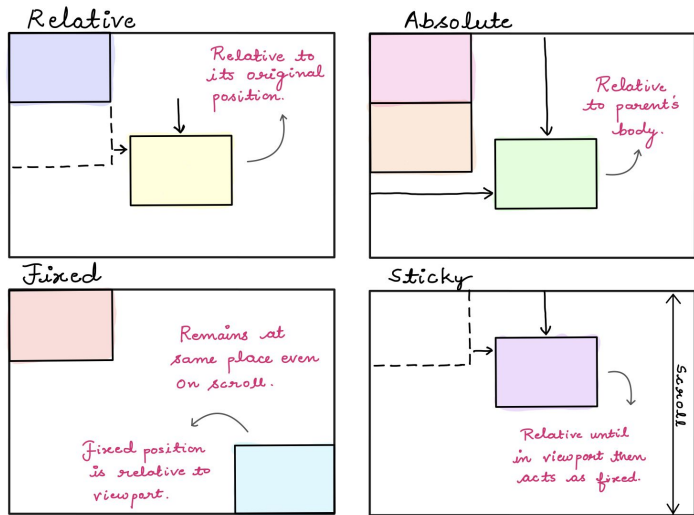
¿Qué es position en CSS?



¿Qué es position en CSS?

CSS POSITIONING

@freethkum



La propiedad "position" de CSS se utiliza para controlar cómo se posiciona un elemento HTML en relación con su contenedor o con otros elementos en la página web.

Hay varios valores que puedes asignar a la propiedad "position", veremos dos de ellos.

Los tipos más básicos de posición son estática y relativa.

position: absolute;

Por defecto el valor de **position** es **static**.

Estos elementos se salen del flujo de la página.

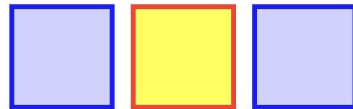
Los elementos con **position: absolute;** se posicionan con respecto al contenedor padre más cercano que tenga un posicionamiento diferente a **static**

Si no encuentra uno, se irá al elemento HTML

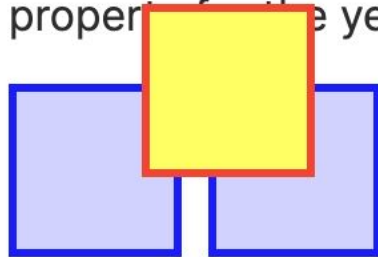
Al establecer el valor de **absolute** desbloqueamos las propiedades:

```
top: ;  
right: ;  
bottom: ;  
left: ;
```

In this demo you can control the position property for the yellow box.



In this demo you can control the position property for the yellow box.



position: relative;

Estos elementos se quedan en el flujo de la página.

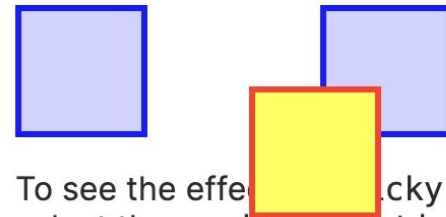
Al establecer el valor de **relative** desbloqueamos las propiedades:

```
top: ;  
right: ;  
bottom: ;  
left: ;
```

In this demo you can control the position property for the yellow box.



In this demo you can control the position property for the yellow box.



To see the effect of sticky position, select the position: sticky

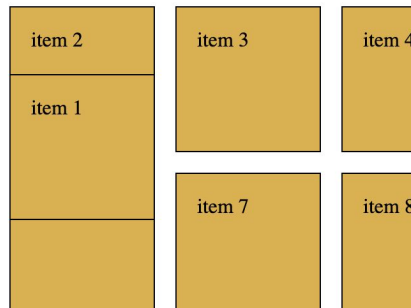
position: fixed;

Se comporta muy similar a **position absolute**, ya que el elemento se remueve del flujo de la página.

De manera similar a **position: absolute**, busca al contenedor padre con un posicionamiento diferente a static, si no se posiciona con respecto al elemento HTML.

Al establecer el valor de **fixed** desbloqueamos las propiedades:

```
top: ;  
right: ;  
bottom: ;  
left: ;
```



position: sticky;

Se comporta muy similar a **position: relative**, ya que el elemento queda en flujo de la página.

Pero una vez que definimos la propiedad **top**, se quedará pegada a tantos pixeles de la parte superior de nuestra página.

