**Modelo de caja:**

• Cada elemento en HTML es una caja, y esas cajas se componen de márgenes (margin), bordes (border), relleno (padding) y finalmente el contenido (content).

• Mediante CSS podemos manipular todas estas propiedades para cambiar la apariencia de cada elemento. A su vez, estas propiedades pueden aplicarse de manera diferente a los cuatro lados de cada caja (top, right, bottom y left).

• Muchas de las propiedades solo aplican a elementos de bloque o semibloque.



**Propiedades**:

• Width: si un elemento de bloque no tiene declarada la propiedad width, el ancho será igual al 100% de su padre contenedor. Podemos asignarle un valor a esta propiedad usando cualquier unidad de medida, como % o px.

• Height: si un elemento no tiene declarado la propiedad height, el alto será igual a la altura que le proporcione su contenido. Sea un elemento de bloque o de línea. Podemos asignarle un valor a esta propiedad usando cualquier unidad de medida, como px. Para la altura de los elementos no se recomienda usar %.

• Padding: es el espacio de relleno que podemos agregar entre el contenido del elemento y su borde. Podemos asignarle un valor a esta propiedad usando cualquier unidad de medida indicando los valores de la siguiente manera:

* *Único valor*: se aplicará a todos los lados. Ej.: div {padding: 12px;}.
* *Dos valores*: el 1ero va a indicar de arriba y abajo, y el 2do el de la izquierda y la derecha. Ej.: div {padding: 12px 20px;}.
* *Tres valores*: el 1ero va a indicar el de arriba, el 2do el de la izquierda y la derecha y el 3ero será el de abajo. Ej.: div {padding: 12px 20px 18px;}.
* *Cuatro valores*: representarán los 4 costados individualmente. Empezaremos por el valor de arriba y seguiremos (en el sentido de las agujas del reloj) por derecha, abajo e izquierda.

• Border: se ubica entre el contenido y el margen. Podemos asignarle un valor a esta propiedad definiendo el *estilo* de línea (*solid*, *dotted*, *dashed* o *double*), su *espesor* (cualquier unidad de medida de CSS) y su *color* (ser cualquier color válido de CSS). Ej.: div {border: solid 3px yellow}.

• Margin: margen exterior del elemento. Sirve para separar una caja de la otra. Podemos asignarle valor a esta propiedad usando cualquier unidad de medida. De la misma manera que con padding, podemos asignar también 2, 3 y hasta 4 valores para la propiedad.

• Box-sizing: ayuda a calcular el ancho o el alto total de un elemento, sumando todas las propiedades que anteriores.

* *Border-box*: el ancho y alto que indiquemos tomará en cuenta no solo el contenido del elemento, sino también el padding y el borde, dejando solo el margen por fuera.

**POSITION**: permite trasladar o superponer un elemento desde su posición original a una nueva posición.

**Puntos de referencia**: cada uno de los elementos de la página web tiene 4 puntos de referencia y esos son sus costados (*top, right, bottom y left*). Cuando desplazamos un elemento tomando un costado como referencia,

empujaremos el elemento si el nº es positivo o tiraremos de él si el nº es negativo.

**Propiedades**:

• Relative: permite trasladar un elemento desde su posición original a una nueva posición. Al posicionar la caja 1 de manera relativa, el espacio que ocupaba originalmente seguirá ocupado, es decir, que los elementos que estén a su lado (caja 2) no van a ser afectados por esta modificación. Solemos utilizarlo cuando queremos desplazar un elemento sin modificar el flujo original de los demás que están a su lado.



• Absolute: permite trasladar un elemento tomando como puntos de referencia los costados el body. Cuando movemos una caja de manera absoluta, el espacio que ocupaba quedará vacío y otros elementos podrán ocuparlo.

* Position relative + position absolute: si cajas (hijas) están dentro de otra caja (padre), el punto de referencia seguirá siendo el body a menos que hagamos relativa la posición de su padre. Se suele utilizar cuando queremos sacar un elemento del flujo normal y posicionarlo en un punto fijo con respecto a su contenedor o el body.

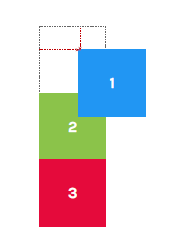
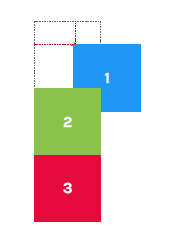


• Fixed: los puntos de referencia serán los costados la ventana del navegador. Sin importar que hagamos scroll en la página, el elemento siempre se mantendrá fijo con respecto a la ventana del navegador. Se utiliza cuando queremos que un elemento siga al usuario a medida que navega nuestro sitio. Ej.: un botón fijo para siempre tener disponible la opción de crear un nuevo producto.

• Sticky: puede considerarse un *híbrido* de los posicionamientos relativo y fijo. Un elemento con posicionamiento sticky es tratado como un elemento posicionado relativamente hasta que cruza un umbral especificado, en cuyo punto se trata como fijo hasta que alcanza el límite de su padre.

**Z-INDEX**: permite cambiar el orden de las “capas” dentro de un documento HTML. Esta propiedad controla cómo se apilan las capas en CSS. Podemos considerar que cada elemento está en una capa diferente y que esas capas van en el orden en que aparecen los elementos en el HTML. Solo podemos modificar el z-index de los elementos que tengan position: relative, absolute o fixed.

1. Cualquier elemento al cual le asignemos position se mostrará por encima del resto y tendrá un valor de z-index de 0. Ej.: .caja-1 {position: relative; left: 100px; top: 50px;}

1)2)

1. Si a la segunda caja le agregamos la propiedad z-index con el valor 10, se moverá delante del resto ya que ahora tiene un valor mayor. .caja-1 { … } / .caja-2 {position: relative; z-index: 10;}.