

INSTITUTO SUPERIOR DE FORMACIÓN TÉCNICA N° 177



PRÁCTICAS PROFESIONALIZANTES

Unidades de Medida, Displays, Modelo de caja

Prof. Lic. Walter Carnero

UNIDADES DE MEDIDA

Debemos saber que existen dos tipos de unidades de medidas

1.Absolutas

2.Relativas

La diferencia entre estos dos tipos radica en que si el valor de la propiedad tomará siempre el mismo tamaño o si estará relacionado o dependerá del tamaño de otro Elemento.

MEDIDAS ABSOLUTAS

Son fijas, no dependen de ningún otro factor

En su momento fueron ampliamente utilizadas, en la actualidad son poco adecuadas para la web, debido a que no se adaptan a distintas resoluciones o pantallas, que es lo que se busca al desarrollar webs responsivas.

Las unidades absolutas utilizadas en desarrollo web se denominan **píxeles** son muy fáciles de utilizar y aplicar en distintas pantallas.

Un ejemplo aplicado a un contenedor sería:

```
div{
  width: 350px;
  font-size: 23px;
}
```

MEDIDAS RELATIVAS

Al contrario de las absolutas, dependen de algún otro factor como el tamaño de letra, la resolución, etc.)

En CSS tienen mayor potencia y flexibilidad, son ideales para trabajar en dispositivos con diferentes tamaños y resoluciones ya que brindan una amplia versatilidad.

em, 1em = tamaño de fuente relativo a la **herencia** o al valor por defecto del navegador.

rem, 1rem = tamaño de fuente **relativo al valor por defecto** del navegador.

vw, 100vw = al **total del ancho visible** del navegador.

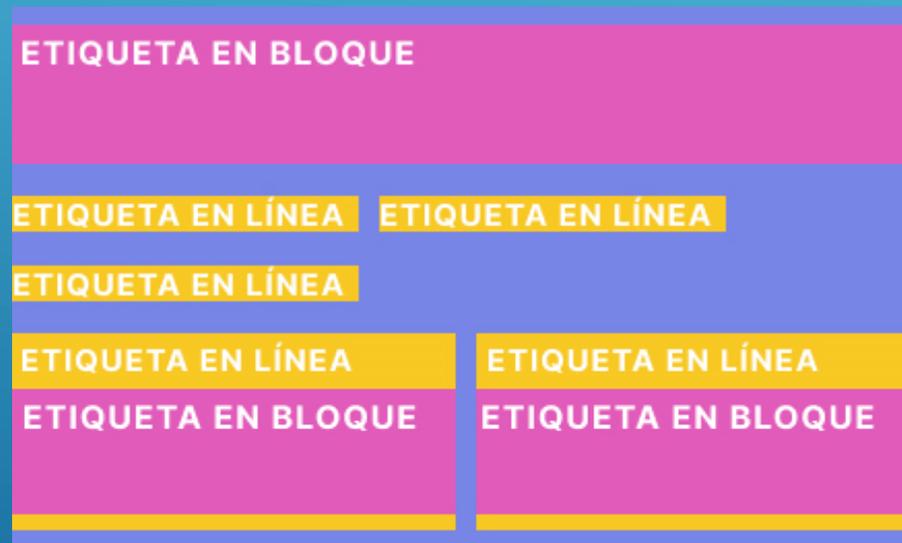
vh, 100vh = al **total del alto visible** del navegador.

%, porcentaje **relativo** al tamaño del **elemento padre**.

DISPLAYS

Como vimos en clases anteriores, cada elemento HTML tiene su tipo de salida concreta

Esos valores están dados por **display** **block** o **inline** y dichos valores estaban relacionados de forma nativa a cada elemento HTML.



Pero... Estos comportamientos nativos pueden ser **modificados**.

VALORES PARA LA PROPIEDAD **DISPLAY**

La propiedad **display** cambia el tipo de representación del elemento indicado, si bien puede tomar muchos valores diferentes, ahora solo nos concentraremos en **4 de ellos**:

```
display: block | inline | inline-block | none;
```

VALORES PARA LA PROPIEDAD **DISPLAY**

block

Ocupa el 100% del ancho de su contenedor y comienzan en una nueva línea.

inline

Ocupan el ancho de su contenido y **no aceptan** propiedades de **width, height** o **margins** y **padding**s superiores.

inline-block

Es la combinación de los anteriores, ocupa el ancho de su contenido pero **si acepta que se modifique su tamaño o sus propiedades del modelo de caja.**

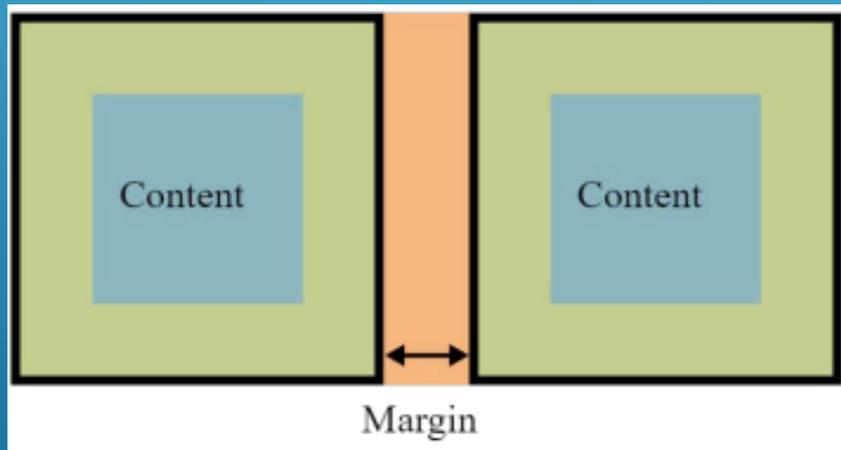
none

Este valor hace que el **elemento referenciado no sea mostrado** el elemento no ocupa espacio donde debería estar, desaparece del flujo del navegador.

PROFUNDIZANDO LAS PROPIEDADES **MARGIN Y PADDING**

La propiedad **margin**

Recordemos que se utiliza para crear espacios alrededor de los elementos, este espacio se encuentra **por fuera** de los bordes definidos.

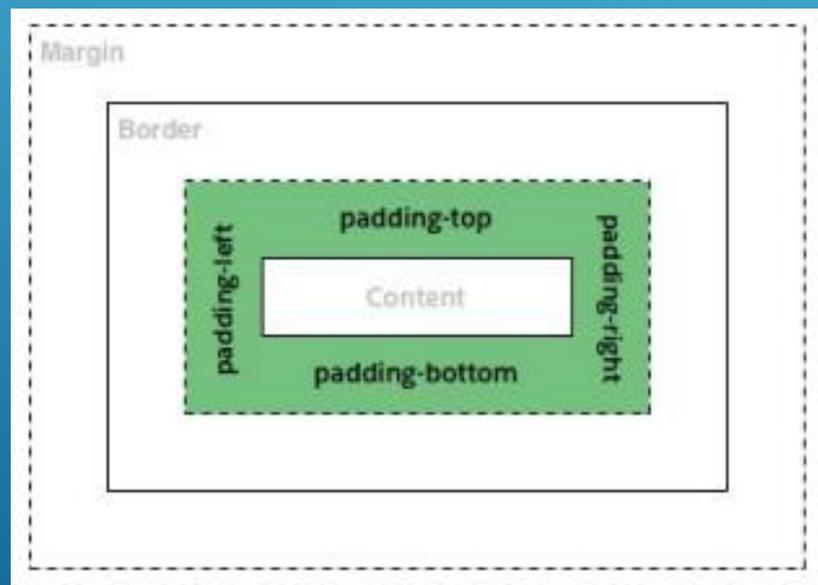


```
/* top right bottom left */  
margin: 10px 20px 10px 20px;  
/* top right/left bottom */  
margin: 10px 20px 10px;  
/* top/bottom right/left */  
margin: 10px 20px;  
/* top/right/bottom/left */  
margin: 10px;
```

PROFUNDIZANDO LAS PROPIEDADES **MARGIN Y PADDING**

La propiedad **padding**

Se utiliza para crear espacios alrededor de los elementos, este espacio se encuentra **por dentro** de los bordes definidos.



```
/* top right bottom left */  
padding: 10px 20px 10px 20px;  
/* top right/left bottom */  
padding: 10px 20px 10px;  
/* top/bottom right/left */  
padding: 10px 20px;  
/* top/right/bottom/left */  
padding: 10px;
```

**FIN DE LA PRESENTACIÓN, AHORA A ACLARAR
TODAS LAS DUDAS...**

GRACIAS POR LA ATENCIÓN !!!