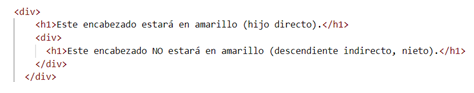
**SELECTORES**

* **Hijo ( > )**

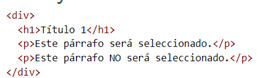


El elemento hijo directo

* **Adyacente ( + )**



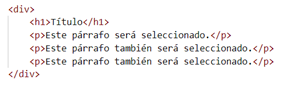
Hermanos y directamente adyacentes



* **Hermano ( ~ )**



Hermanos directamente adyancentes o no

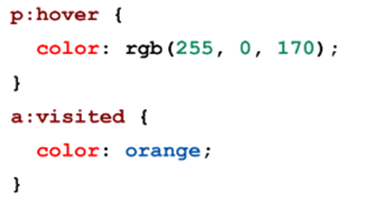


* **Selector con clase**

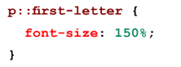


Selecciona los párrafos con clase “destacado”

* **Pseudoclases**

****

* + Hover: cuando se pasa el raton por encima
  + Visited: para enlaces ya visitados
  + Active: cuando el usuario activa el elemento
  + Focus: cuando el elemento recibe el foco
* **Pseudoelementos**

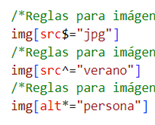
****

* + **first-line:** la primera línea de un texto
  + **before:** inserta contenido antes del elemento
  + **after:** inserta contenido despues del elemento
  + **first-letter**: la primera letra de un texto
  + **placeholder:** estilos al place holder de un input
  + **selection:** parte del contenido seleccionado por el usuario
* **Selectores de atributo**

****

Afecta a los elementos a que tengan un atributo target=”\_blank”

Mas Ejemplos:



**PROPIEDADES ABREVIADAS**

* **margin**

1. margin-top
2. margin-right
3. margin-bottom
4. margin-left

o

1. margin-top y margin-bottom
2. margin-right y margin-left



* **padding**

1. padding-top
2. padding-right
3. padding-bottom
4. padding-left



o

1. padding-top y padding-bottom
2. padding-right y padding-left

* **font**

1. font-style
2. font-variant
3. font-weight
4. font-size
5. line-height
6. font-family



* **background**

1. background-color
2. background-image
3. background-position
4. background-size
5. background-repeat
6. background-attachment
7. background-clip
8. background-origin



* **border**

1. border-width
2. border-style
3. border-color



* **border-radius**

1. top-left
2. top-right
3. bottom-right
4. bottom-left



* **outline:** borde que no afecta al tamaño total del elemento

1. outline-width
2. outline-style
3. outline-color



* **box-shadow**

1. desplazamiento-horizontal
2. desplazamiento-vertical
3. desenfoque (opcional)
4. color



* **flex**

1. flex-grow
2. flex-shrink
3. flex-basis



* **grid**

1. grid-template-rows
2. grid-template-columns
3. grid-template-areas
4. grid-auto-rows
5. grid-auto-columns
6. grid-auto-flow



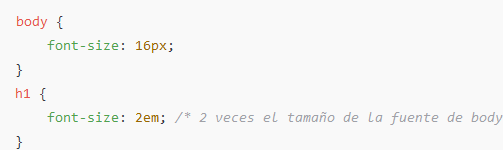
* **grid-area**

1. grid-row-start
2. grid-column-start
3. grid-row-end
4. grid-column-end

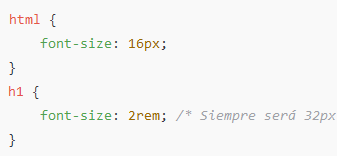


**UNIDADES RELATIVAS**

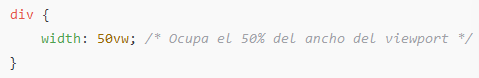
* **em:** relativo al tamaño de la fuente del elemento padre



* **rem:** relativo al tamaño de la fuente del elemento raíz (html)



* **vw:** relativo al 1% del ancho de la ventana visible del navegador



**MODELO DE CAJAS**

**BOX-SIZING:** Como se calculan las dimensiones de un elemento, si el padding y el borde se incluyen o no del tamaño total del elemento (width y height)

* **content-box -> width y height solo incluyen el contenido del elemento**
* **border-box -> width y height incluyen el contenido, el padding, y el border**

**DISPLAY**

* **none ->** elimina el elemento, queda invisible y se comporta como si no existiese en el flujo del diseño
* **block ->** se comporta como un bloque, Inicia en una nueva línea, ocupa el ancho completo del contenedor padre
* **inline ->** se comporta como contenido en línea, no inicia en una nueva línea, solo ocupa el ancho necesario para su contenido
* **inline-block ->** se comportan como bloques dentro de esa línea, no inicia en una nueva línea,
* **flex ->** se convierte en un contenedor flexible, tiene control de alineación y distribución con distintas propiedades
* **inline-flex ->** el contenedor flexible se comporta como un elemento en línea
* **grid ->** se comporta como un contenedor de cuadrícula, tiene control de distribución con distintas propiedades
* **inline-grid ->** el contenedor de cuadrícula se comporta como un elemento en línea

**POSICIONAMIENTO**

**POSITION**

* **static:** se colocan de acuerdo al orden normal del documento (código)
* **relative:** se colocan de acuerdo al orden normal, pero se pueden ,over respecto a su posición original (top, right, bottom…)
* **absolute:** se posiciona en relación con el contenedor mas cercano que tenga una posición distinta de static, en su defecto se posiciona en relación con <html>
* **fixed:** se posiciona en relación con la ventana del navegador, permanecerá en la misma posición incluso cuando el usuario haga scroll
* **sticky:** se comporta como relative hasta que llega a un cierto punto en la pagina
* **z-index:** controla que elementos se muestran por encima de otros