

# CSS - Angular



# CSS

Es un lenguaje que trabaja junto con HTML para proveer estilos visuales a los elementos del documento web.

- Ahorra trabajo. Se puede controlar el diseño de varias páginas a la vez.
- Se pueden almacenar en archivos \*.css

*CSS 3 es la última versión estándar.*

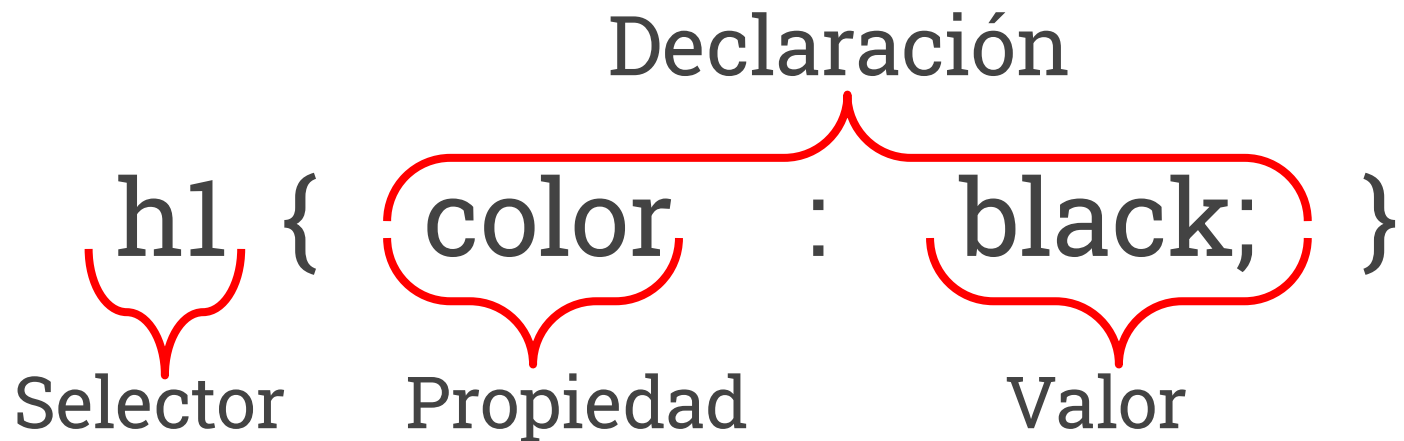


# ¿Para qué utilizar CSS?

Para definir estilos en los documentos web, incluyendo el diseño, la disposición de los elementos y para responder a las variaciones en la pantalla en cuanto a diferentes dispositivos y tamaños de pantalla.

CSS define conjunto de reglas que permiten describir cada una de las partes que componen los estilos CSS.

Regla de css:



**Selector:** indica el elemento o elementos HTML a los que se aplica la regla CSS.

**Declaración:** especifica los estilos que se aplican a los elementos.

**Propiedad:** permite modificar el aspecto de una característica del elemento.

**Valor:** indica el nuevo valor de la característica modificada en el elemento.

# Comentarios

El comienzo de un comentario se indica mediante los caracteres `/*` y el final del comentario se indica mediante `*/`, tal y como se muestra en el siguiente ejemplo:

```
/* Este es un comentario en CSS */
```

# Formas de insertar CSS

- **Hojas de estilo en línea** (Incluir CSS en los elementos HTML)
- **Hojas de estilo interna** (Insertar CSS en el mismo documento HTML)
- **Hojas de estilo externa** (Definir CSS en un archivo externo)
- **Hojas de estilo múltiple**

1. ¿Porque se llama hojas de estilo en cascada?
2. Según tu opinión ¿cuál es la forma de insertar estilos más recomendada y por qué?

Para trabajar en grupos!

# DEMO

---



# Selectores

## “a quién hay que aplicar el estilo”

Una misma regla puede aplicarse a varios selectores y, a un mismo selector se le pueden aplicar varias reglas.

Clic [aquí](#) para saber TODO sobre selectores.

# Selectores básicos

**Selector universal:** Se utiliza para seleccionar todos los elementos de la página.

```
* {  
    margin: 0;  
    padding: 0;  
}
```

**Selector de tipo o etiqueta:** Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector.

```
h1 {  
    color: red;  
}  
h2 {  
    color: blue;  
}  
p {  
    color: black;  
}
```

Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector.

**Actividad: escriba el HTML para estos selectores**

CSS permite agrupar todas las reglas individuales en una sola regla con un selector múltiple.

```
h1, h2, h3 {  
    color: #8A8E27;  
    font-weight: normal;  
    font-family: Arial, Helvetica, sans-serif;  
}
```

Es habitual agrupar las propiedades comunes de varios elementos en una única regla CSS y posteriormente definir las propiedades específicas

```
h1, h2, h3 {  
    color: #8A8E27;  
    font-weight: normal;  
    font-family: Arial, Helvetica, sans-serif;  
}  
h1 { font-size: 2em; }  
h2 { font-size: 1.5em; }  
h3 { font-size: 1.2em; }
```

## Selector descendente

Selecciona los elementos que se encuentran dentro de otros elementos. Se encuentra entre las etiquetas de apertura y de cierre del otro elemento.

```
p span { color: red; }
```

```
h1 span { color: blue; }
```

Actividad: escriba el HTML para estos selectores

La sintaxis formal del selector descendente se muestra a continuación:

**elemento1 elemento2 elemento3 ... elementoN**

Los selectores descendentes siempre están formados por dos o más partes separadas entre sí por espacios en blanco. La última parte indica el elemento sobre el que se aplican los estilos y todas las partes anteriores indican el lugar en el que se tiene que encontrar ese elemento para que los estilos se apliquen realmente.

# Selector de clase

Consideremos el siguiente HTML

```
<body>  
<p>Lorem ipsum dolor sit amet...</p>  
<p>Nunc sed lacus et est adipiscing accumsan...</p>  
<p>Class aptent taciti sociosqu ad litora...</p>  
</body>
```

¿Cómo se pueden aplicar estilos CSS sólo al primer párrafo?

Para debatir en clase!

# Selector class

Identifica una clase de elementos para aplicarle estilo a todos los de la misma clase.

Corrección del HTML

```
<body>  
<p class="destacado">Lorem ipsum dolor sit amet...</p>  
<p>Nunc sed lacus et est adipiscing accumsan...</p>  
<p>Class aptent taciti sociosqu ad litora...</p>  
</body>
```

En el CSS:

```
.destacado { color: red; }
```

Los **selectores de clase** son imprescindibles para diseñar páginas web complejas, ya que permiten disponer de una precisión total al seleccionar los elementos. Además, estos selectores permiten reutilizar los mismos estilos para varios elementos diferentes.

# ejemplo de class

Identifica una clase de elementos para aplicarle estilo a todos los de la misma clase.

<!-- En el html -->

```
<p class="clase_ej">...</p>
```

Se aplica el estilo usando la clase como selector.

Sintaxis: **.classname { style properties }**

/\* En el Css \*/

```
.clase_ej {color: red;} /* Aplica al 'p' anterior*/
```

# Desafío.....

```
<body>
```

```
<p class="destacado">Lorem ipsum dolor sit amet...</p>
```

```
<p>Nunc sed lacus et <a href="#" class="destacado">est adipiscing</a>  
accumsan...</p>
```

```
<p>Class aptent taciti <em class="destacado">socios que...</em>  
litora...</p>
```

```
</body>
```

¿Como hago para que el párrafo esté en color azul y el enlace a en color rojo utilizando el mismo decorador de class sabiendo que: **.destacado {color:red}**?

Para debatir en clase!



Es posible aplicar los estilos de varias clases CSS sobre un mismo elemento.

En HTML:

```
<p class="especial destacado error">Párrafo de texto...</p>
```

En CSS:

```
.error { color: red; }
```

```
.error.destacado { color: blue; }
```

```
.destacado { font-size: 15px; }
```

```
.especial { font-weight: bold; }
```

¿Qué color tendrá el párrafo?

Para debatir en clase!

# Selectores de ID

El selector de ID permite seleccionar un elemento de la página a través del valor de su atributo id.

La sintaxis de los selectores de ID es muy parecida a la de los selectores de clase, salvo que se utiliza el símbolo de la almohadilla (#)

En CSS

```
#destacado { color: red; }
```

En HTML

```
<p>Primer párrafo</p>
```

```
<p id="destacado">Segundo párrafo</p>
```

```
<p>Tercer párrafo</p>
```

Valor del atributo id no se puede repetir en dos elementos diferentes de una misma página.

# Selectores avanzados

## Selector de hijos

Se trata de un selector similar al selector descendente, pero muy diferente en su funcionamiento. Se utiliza para seleccionar un elemento que es hijo directo de otro elemento y se indica mediante el "signo de mayor que" (>):

```
p > span { color: blue; }
```

```
<p><span>Texto Span</span></p>
```

```
<p> Texto <span>span</span></p>
```

En el ejemplo anterior, el selector `p > span` se interpreta como "cualquier elemento `<span>` que sea hijo directo de un elemento `<p>`", por lo que el primer elemento `<span>` cumple la condición del selector. Sin embargo, el segundo elemento `<span>` no la cumple porque es descendiente pero no es hijo directo de un elemento `<p>`.

# Herencia

Cuando se establece el valor de alguna propiedad en un elemento, todos sus descendientes heredan inicialmente ese mismo valor.

```
<html>
<head>
  <title>Ejemplo de herencia de estilos</title>
  <style type="text/css">
    body { font-family: Arial; color: black; }
    h1 { font-family: Verdana; }
    p { color: red; }
  </style>
</head>
<body>
  <h1>Titular de la página</h1>
  <p>Un párrafo de texto.</p>
</body>
</html>
```

# Unidades de medida

## Relativos

Son más flexibles que las unidades absolutas porque se adaptan más fácilmente a los diferentes medios.

- **em**: relativa respecto del tamaño de letra empleado.
- **ex**: relativa respecto de la altura de la letra x ("equis minúscula") del tipo de letra que se esté utilizando
- **px**: (píxel) relativa respecto de la pantalla del usuario

```
body {  
    font-size: 12px;  
    text-indent: 3em;  
}  
h1 { font-size: 15px }
```

# Unidades absolutas

**in:** del inglés "inches", pulgadas (1 pulgada son 2.54 centímetros)

**cm:** centímetros

**mm:** milímetros

**pt:** puntos (1 punto equivale a 1 pulgada/72, es decir, unos 0.35 milímetros)

**pc:** picas (1 pica equivale a 12 puntos, es decir, unos 4.23 milímetros)

```
body { margin: 0.5in; }
```

```
h1 { line-height: 2cm; }
```

```
p { word-spacing: 4mm; }
```

```
a { font-size: 12pt }
```

```
span { font-size: 1pc }
```

# Porcentajes

Un porcentaje está formado por un valor numérico seguido del símbolo % y siempre está referenciado a otra medida.

En el CSS:

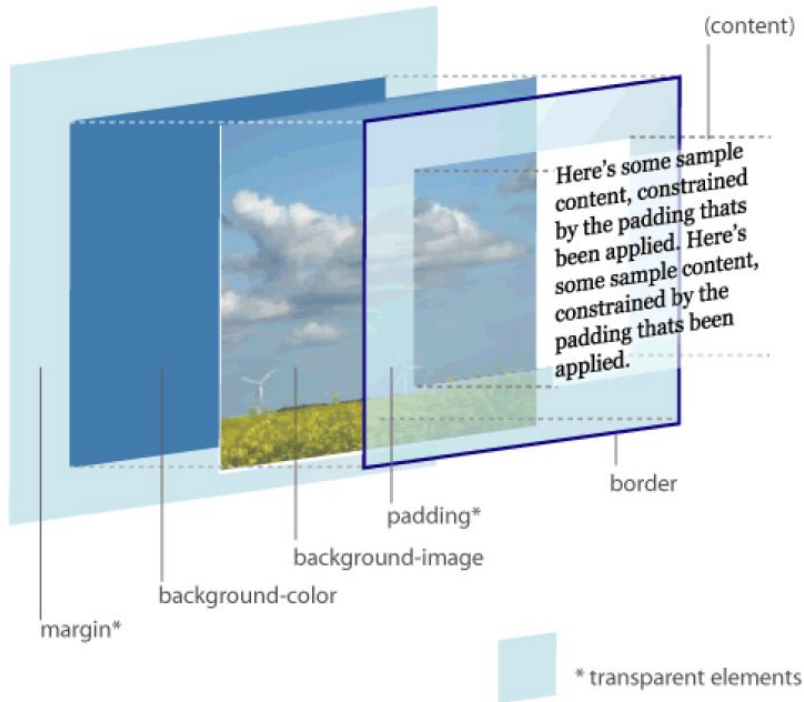
```
div#contenido { width: 600px; }  
div.principal { width: 80%; }
```

En HTML:

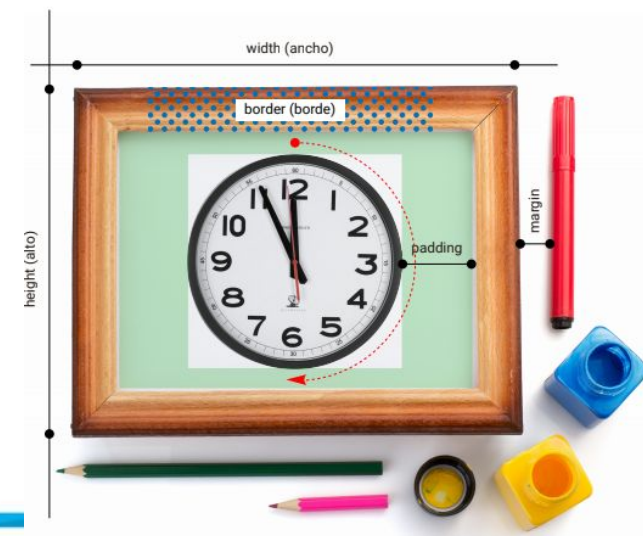
```
<div id="contenido">  
<div class="principal">  
...  
</div>  
</div>
```

Se **recomienda el uso de unidades relativas** siempre que sea posible, ya que mejora la accesibilidad de la página y permite que los documentos se adapten fácilmente a cualquier medio y dispositivo.

# Modelo de cajas



Las cajas de una página se crean automáticamente. Cada vez que se inserta una etiqueta o elemento en la página, se crea una nueva caja rectangular que encierra los contenidos del elemento.





# Anchura y altura

Anchura = width

En CSS:

```
#lateral { width: 200px; }
```

En HTML:

```
<div id="lateral">
```

```
...
```

```
</div>
```

Admite como medidas:

**inherit:** indica que la anchura del elemento se hereda de su elemento padre

**auto:** que es el que se utiliza si no se establece de forma explícita un valor a esta propiedad

# Altura

Altura = height

En CSS:

```
#cabecera { height: 60px; }
```

En HTML:

```
<div id="cabecera">
```

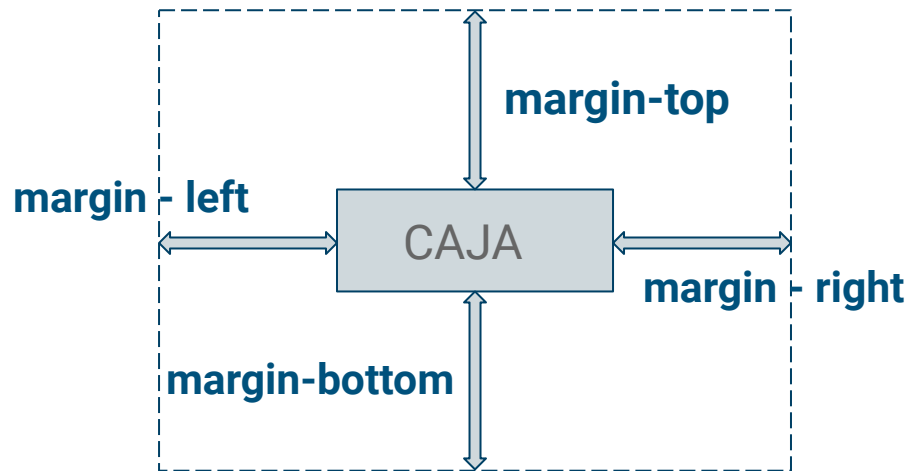
```
...
```

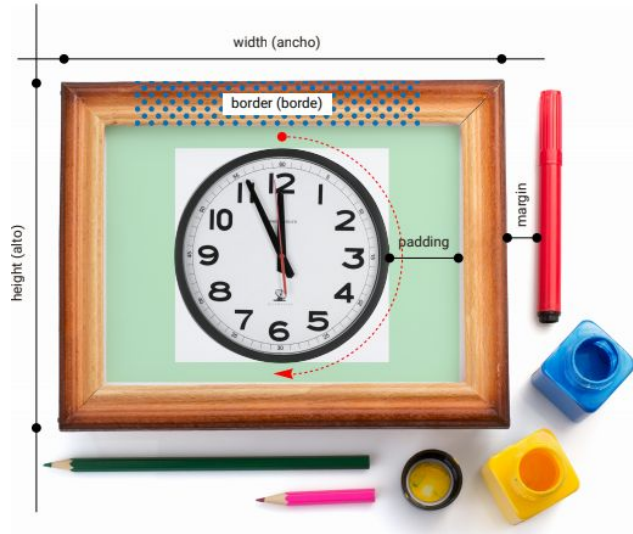
```
</div>
```

Admite como medida  
**porcentaje**  
**auto**  
**inherit**

# Margen

margin-top      => Margen superior  
margin-right    => Margen derecho  
margin-bottom   => Margen inferior  
margin-left     => Margen izquierdo





- ❖ El **padding** es el espacio que hay entre el texto y el borde.

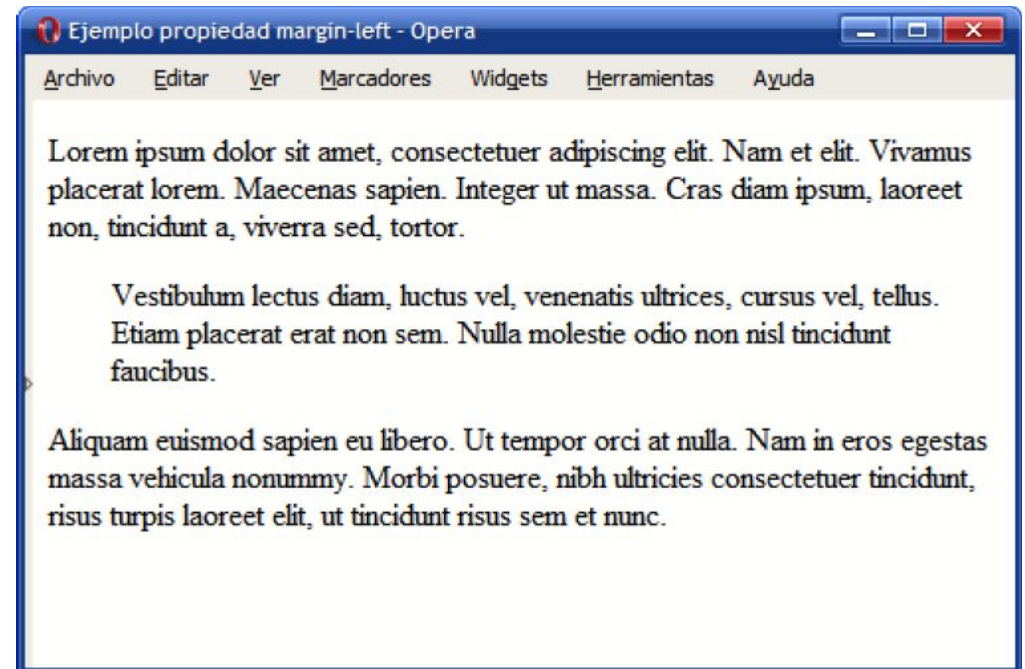
```
1 p {  
2   border: 1px solid powderblue;  
3   padding: 30px;  
4 }
```



- ❖ El **margin** es el espacio que hay entre la caja y las cajas adyacentes.

```
1 p {  
2   border: 1px solid powderblue;  
3   margin: 50px;  
4 }
```

En CSS:  
.destacado  
{  
margin-left: 2em;  
}



Escribe el HTML para obtener como resultado la siguiente página con esos estilos planteados

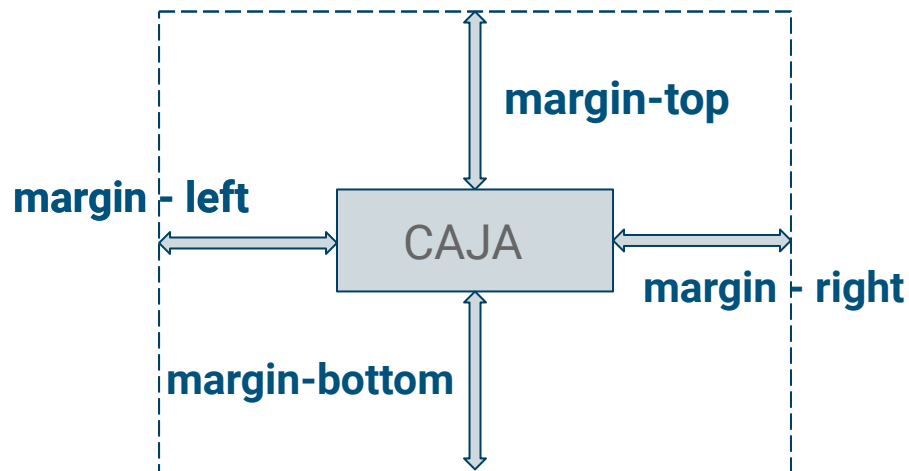
# Relleno

padding-top => Relleno superior

padding-right => Relleno derecho

padding-bottom => Relleno inferior

padding-left => Relleno izquierdo



En CSS:

```
.margen {  
    margin-top: 2em;  
    margin-right: 2em;  
    margin-bottom: 2em;  
    margin-left: 2em;  
}  
  
.relleno {  
    padding-top: 2em;  
    padding-right: 2em;  
    padding-bottom: 2em;  
    padding-left: 2em;  
}
```

¿Te parece que el párrafo  
está alineado a la derecha?

En HTML:

```
<p class="margen">Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Cras vitae dolor eu enim dignissim lacinia. Maecenas blandit. Morbi mi.</p>  
<p class="relleno">Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Cras vitae dolor eu enim dignissim lacinia. Maecenas blandit. Morbi mi.</p>
```

Material extra para ver y practicar de  
manera asíncrona.



# Posicionamiento y visualización

# Propiedad posición

La propiedad **position** especifica el modo de posicionamiento del elemento.

El posicionamiento puede ser:

- **static** normal o estático: se trata del posicionamiento que utilizan los navegadores si no se indica lo contrario.
- **relative** relativo: variante del posicionamiento normal que consiste en posicionar una caja según el posicionamiento normal y después desplazarla respecto de su posición original.
- **fixed** fijo: variante del posicionamiento absoluto que convierte una caja en un elemento inamovible, de forma que su posición en la pantalla siempre es la misma independientemente del resto de elementos e independientemente de si el usuario sube o baja la página en la ventana del navegador.
- **absolute** absoluto: la posición de una caja se establece de forma absoluta respecto de su elemento contenedor y el resto de elementos de la página ignoran la nueva posición del elemento.
- **float** flotante: se trata del modelo más especial de posicionamiento, ya que desplaza las cajas todo lo posible hacia la izquierda o hacia la derecha de la línea en la que se encuentran.

# Static

Los elementos son posicionados "static" por defecto.

Los elementos con este tipo de posicionamiento son posicionados siguiendo el flujo del archivo HTML

```
1  div.static {  
2    position: static;  
3    border: 3px solid #73AD21;  
4  }
```

# Relative

Los elementos con esta posición son puestos en la página relativamente de acuerdo a su posición por defecto.

```
1  div.relative {  
2    position: relative;  
3    left: 30px;  
4    border: 3px solid #73AD21;  
5  }
```

# Fixed

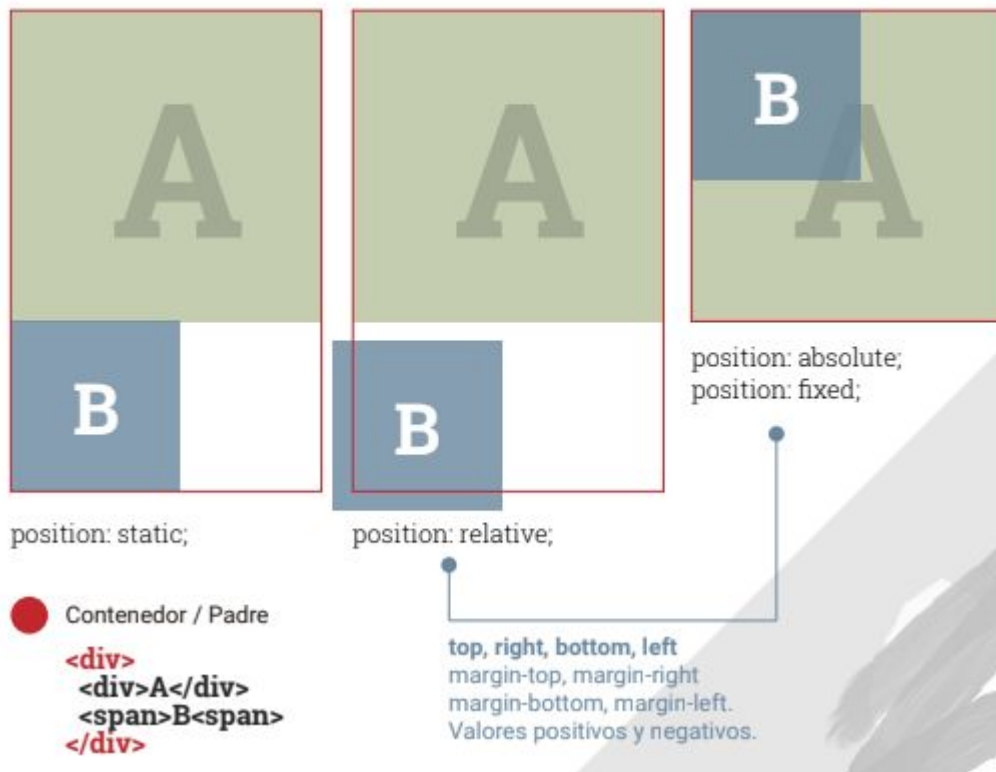
Los elementos con esta posición son posicionados de modo fijo en la vista de la página, es decir, a pesar de tener scroll no se mueven.

```
1  div.fixed {  
2    position: fixed;  
3    bottom: 0;  
4    right: 0;  
5    width: 300px;  
6    border: 3px solid #73AD21;  
7  }
```

# Absolute

Los elementos con esta posición son posicionados de forma relativa respecto a la posición de su ancestro inmediato

## Manipular posición y eje Z



# Superposición de elementos

Los elementos pueden superponerse unos con otros.

La propiedad **z-index** los dispone en capas

```
1  img {  
2    position: absolute;  
3    left: 0px;  
4    top: 0px;  
5    z-index: -1;  
6  }
```



This is a heading  
Because the image has a z-index of -1

# Texto - Tipografía

## font-family

Tipos:

font-family: Arial, Helvetica, sans-serif;

font-family: "Times New Roman", Times, serif;

font-family: "Courier New", Courier, monospace;

font-family: Georgia, "Times New Roman", Times, serif;

font-family: Verdana, Arial, Helvetica, sans-serif;

Además de todas las unidades de medida relativas y absolutas y el uso de porcentajes, CSS permite utilizar una serie de palabras clave para indicar el tamaño de letra del texto:

**tamaño\_absoluto:** indica el tamaño de letra de forma absoluta mediante alguna de las siguientes palabras clave: xx-small, x-small, small, medium, large, x-large, xx-large.

**tamaño\_relativo:** indica de forma relativa el tamaño de letra del texto mediante dos palabras clave (larger, smaller) que toman como referencia el tamaño de letra.

# Enlaces

## Pseudo-clases

**:hover**, aplica estilos al enlace sobre el que el usuario ha posicionado el puntero del ratón.

**:active**, aplica estilos al enlace que está pinchando el usuario. Los estilos sólo se aplican desde que el usuario pincha el botón del ratón hasta que lo suelta, por lo que suelen ser unas pocas décimas de segundo.

**:link**, aplica estilos a los enlaces que apuntan a páginas o recursos que aún no han sido visitados por el usuario.

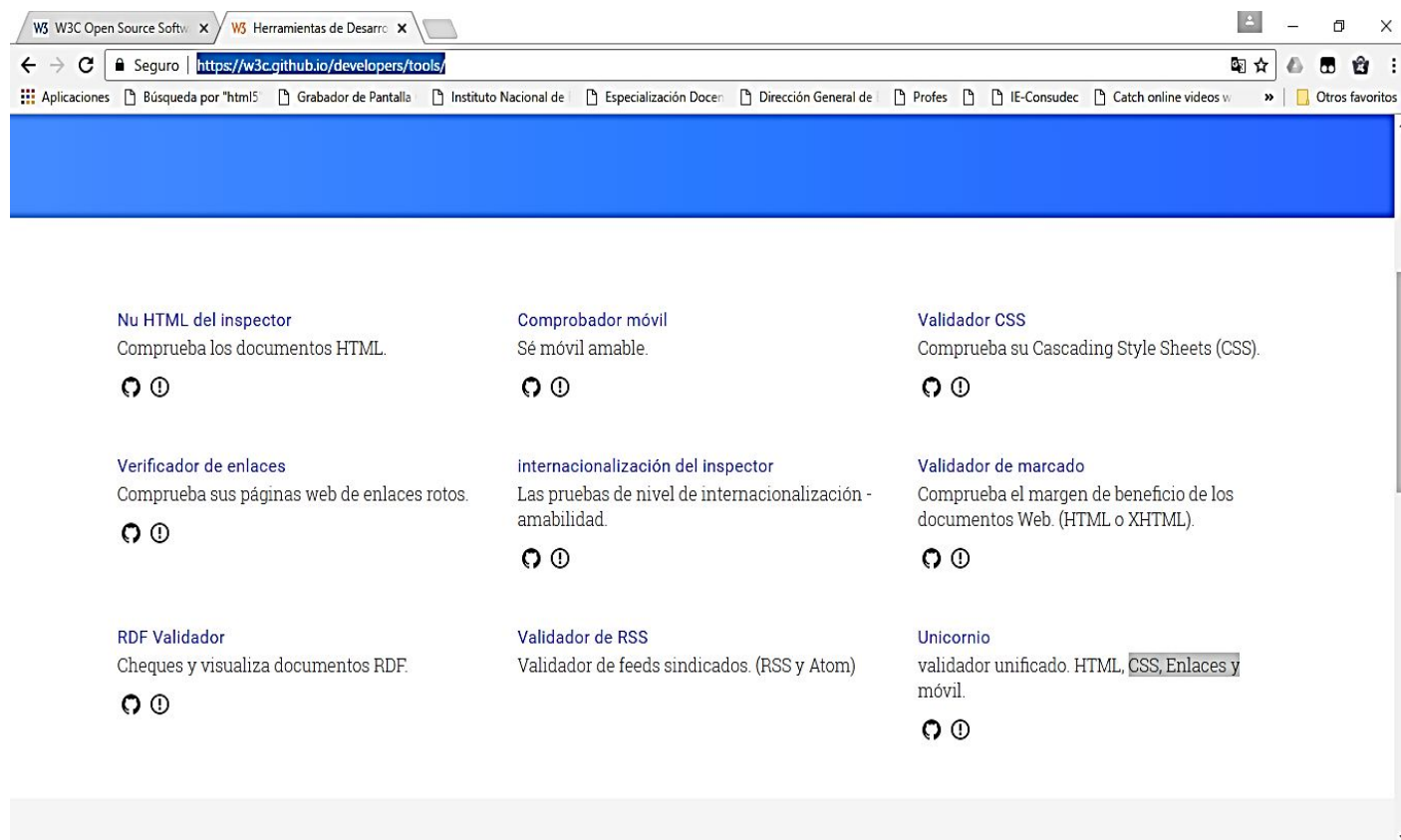
**:visited**, aplica estilos a los enlaces que apuntan a recursos que han sido visitados anteriormente por el usuario. El historial de enlaces visitados se borra automáticamente cada cierto tiempo y el usuario también puede borrarlo manualmente.

Ejemplo: `a:hover { text-decoration: none; }`



# Validación de Código

<https://w3c.github.io/developers/tools/>



## REFERENCIAS:

Javier Eguíluz Pérez. Introducción a css autor:

[https://www.w3.org/wiki/CSS/\\_/\\_Selectores\\_CSS#Lista\\_de\\_selectores\\_css](https://www.w3.org/wiki/CSS/_/_Selectores_CSS#Lista_de_selectores_css)