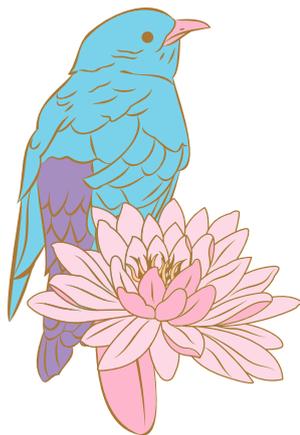


CSS



*¿Qué es?*



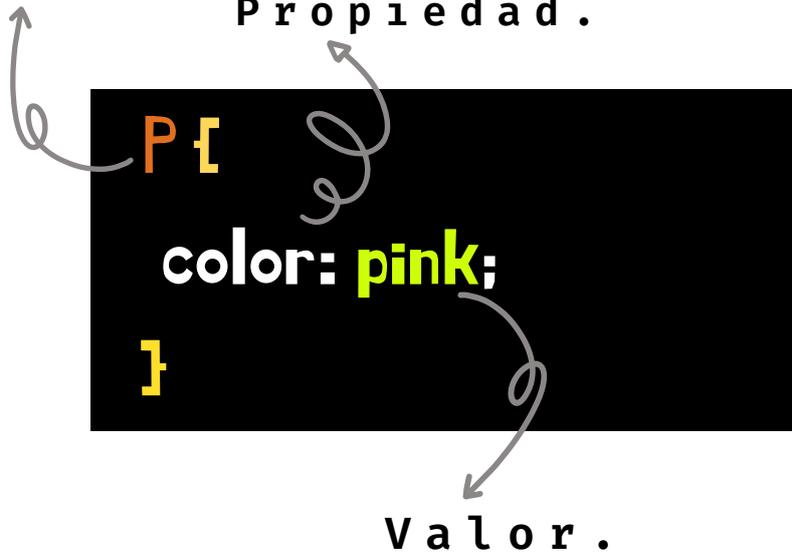


# Anatomía CSS



S e l e c t o r .

P r o p i e d a d .



V a l o r .

## Selector

Son la referencia a los elementos **HTML** que queremos **estilizar**. Los nombres de estas etiquetas van seguidas de una llave de **apertura** y otra de **cierre**.  
Por ejemplo: `h1 {}`

## Propiedades

Son el tipo de **estilo** que queremos darle a nuestros **elementos**. Van seguidas de **dos puntos**. Las propiedades deben estar **dentro de las llaves** del **selector** que definimos anteriormente. Podemos escribir **diferentes propiedades en un mismo selector**.  
Por ejemplo: `h1 { color: black; }`

## Valores

Son el **estilo** que queremos que tomen nuestros **elementos HTML** con **respecto a una propiedad**. Van seguidas de un **punto y coma**.  
Por ejemplo: `h1 { color: red; }`



@irisval06

# Tipos de selectores



```
*{ → Universal
margin: 0;
}

h1{ → Tipo
color: pink;
}

.saludo{ → Clase
font-size: 2em;
}

#id{ → Id
border-radius: 20px;
}
```

## Selector de Clase

Si nuestras etiqueta de **HTML** tienen un **atributo class** podemos usar ese valor o identificador para que los cambios en el CSS **afecten únicamente a ese elemento**.

## Selector Universal

Las propiedades se aplicarán a **todos los elementos** de nuestro **HTML**.



## Selector de ID

Es similar al selector de clase, si la etiqueta **HTML** tiene un **ID** podemos afectar **solo ese elemento**.

## Selector de tipo

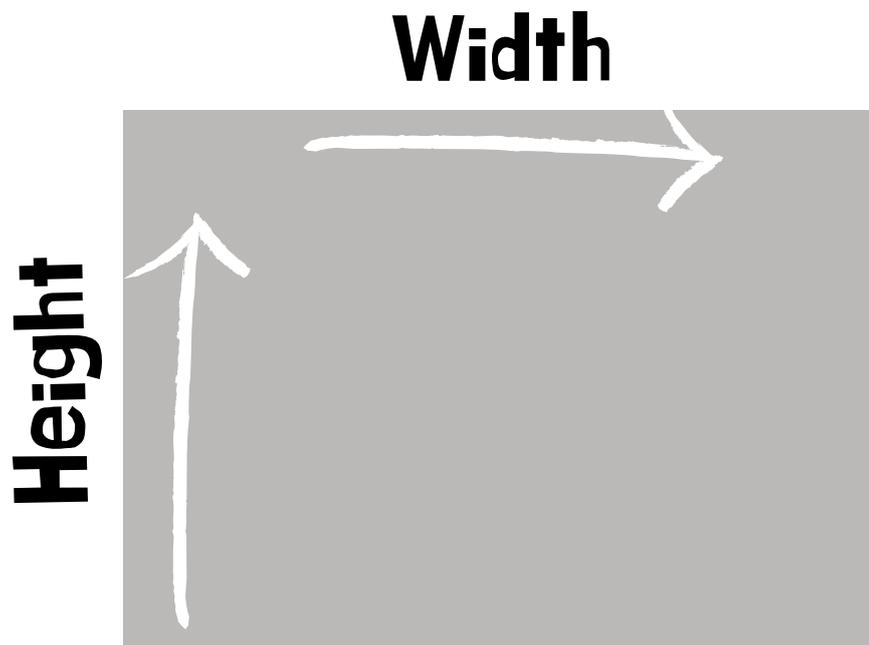
Son selectores que se aplican a **cierto elemento HTML** en **específico**. Las propiedades se aplicarán a la **etiqueta que queremos**.

## Width

Significa la anchura del contenido de un elemento.

## Height

Significa la altura del contenido de un elemento.  
En ambos se pueden aplicar unidades de medidas. Tanto absolutas como relativas



# Propiedades Css más usadas



## background

Puede definir el color de fondo o la url de fondo de un elemento. Por ejemplo:

```
background: url('puppy.png');
```

## background-color

Define el color de fondo de un elemento. Por ejemplo:

```
background-color: red;
```

## font-size

Define el tamaño de la fuente (letra). Por ejemplo:

```
font-size: 20px;
```

## font-family

Define la familia tipográfica de la fuente. Por ejemplo:

```
font-family: 'Roboto', sans-serif;
```

## opacity

Determina la transparencia del elemento. Tiene valores entre **0** y **1**, que pueden verse como un porcentaje. Por ejemplo, si quiero que mi elemento se vea con una transparencia del **50%**, escribiría:

```
opacity: 0.5;
```

## animation

Esta propiedad permite que animemos nuestros elementos. La veremos a profundidad después.

## transition

Las transiciones CSS le permiten cambiar los valores de las propiedades sin problemas durante una duración determinada. Debemos tener presente que una transición **NO es una animación**.

## border

Define el tamaño, estilo y color del borde de un elemento. Por ejemplo:

```
border: 2px solid yellow;
```

## border-radius:

Define qué tan redondeado quiero mi elemento. Por ejemplo:

```
border-radius: 20px;
```

# Pseudo-Clases

Las Pseudo-clases y Pseudo-elementos **nos permiten ser aún más específicos** con **qué elementos** o **partes** de **nuestros elementos** deben recibir los estilos.

Para usarlas debemos **definir** el **selector** base por ejemplo **"p"**. **seguido de dos puntos** y la pseudo-clase que **queremos estilizar**.

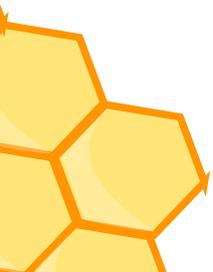
por ejemplo: **p:first-child**.

En el caso de los **pseudo-elementos** debemos usar el dos puntos, 2 veces **p::first-letter**.

```
P: first-child{
  color: white;
}
P: last-child{
  color: pink;
}
```

## **:active**

Representa un elemento (como un **botón**) que el usuario **está activando**. Cuando el usuario **presiona** el botón, **comienza la activación** y **termina cuando lo suelta**. Se usa normalmente en los elementos **<a>** y **<button>**, se puede usar en otros elementos.



## :checked

Representa cualquier **radio**

```
<input type="radio">
```

En **checkbox**

```
<input type="checkbox">
```

E incluso **option** en un elemento **select** que está marcado a un **estado on**.

## :any-link

Representa un elemento que actúa como el **ancla origen de un hipervínculo** independientemente de si ha sido visitado. Coincide con cualquier elemento `<a>`, `<area>` o `<link>` con un atributo **href**. Por lo tanto, coincide con todos los elementos que coincidan con `:link` o `:visited`.

## :-webkit-autofill

Selecciona un elemento `<input>` cuando su **valor** es **rellenado automáticamente** por el navegador.

## :default

Representa cualquier elemento de un formulario que sea el **predeterminado** entre un **grupo de elementos relacionados**. Se usa en los elementos `<button>`, `<input type="checkbox">`, `<input type="radio">` y `<option>`.

## :defined

Representa cualquier elemento que **haya sido definido**. Esto incluye cualquier **elemento estándar** integrado en el navegador y **elementos personalizados** que se **hayan definido correctamente**, con el método `CustomElementRegistry.define()`



@irisval06

## :disabled

Representa cualquier **elemento deshabilitado**. Un elemento se encuentra deshabilitado **si no puede ser activado**; por ejemplo ser **seleccionado**, hacer click en él o aceptar texto de entrada. El elemento tiene además **un estado habilitado** en el cual puede ser **activado** o **recibir foco**.

## :empty

Representa cualquier elemento **que no tenga hijos**. Los hijos podrían ser **nodos de elemento o texto**, **incluido el espacio en blanco**. Los **comentarios** o las **instrucciones de procesamiento no afectan** si un elemento **se considera vacío**.

```
div:empty {  
background: lime;  
}
```

## :enabled

Representa **cualquier elemento habilitado**. (se puede **hacer click** en él, **introducir texto**, etc.) o si **acepta el foco**. El elemento **también tiene un estado deshabilitado** el en cuál, **no puede ser activado ni acepta el foco**.

```
input:enabled {  
color: blue;  
}
```



## :first-child

Representa el primer elemento entre un grupo de elementos hermanos.

```
<p>Este texto está seleccionado!</p>
```

```
<p>Este texto no está seleccionado.</p>
```

```
p:first-child {  
color: lime;  
}
```

## :first-of-type

Representa el primer elemento de su tipo entre un grupo de elementos hermanos.

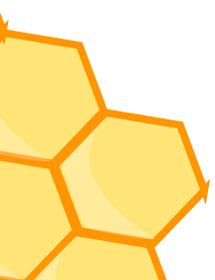
```
p:first-of-type {  
color: red;  
}
```



# Pseudo-elementos

Al igual que las **pseudo-clases**, los **pseudo-elementos** se **añaden a los selectores**, pero en cambio, no **describen un estado especial** sino que, **permiten añadir estilos a una parte concreta** del documento. Por ejemplo, el pseudo-elemento **::first-line** selecciona solo la primera línea del elemento especificado por el selector.

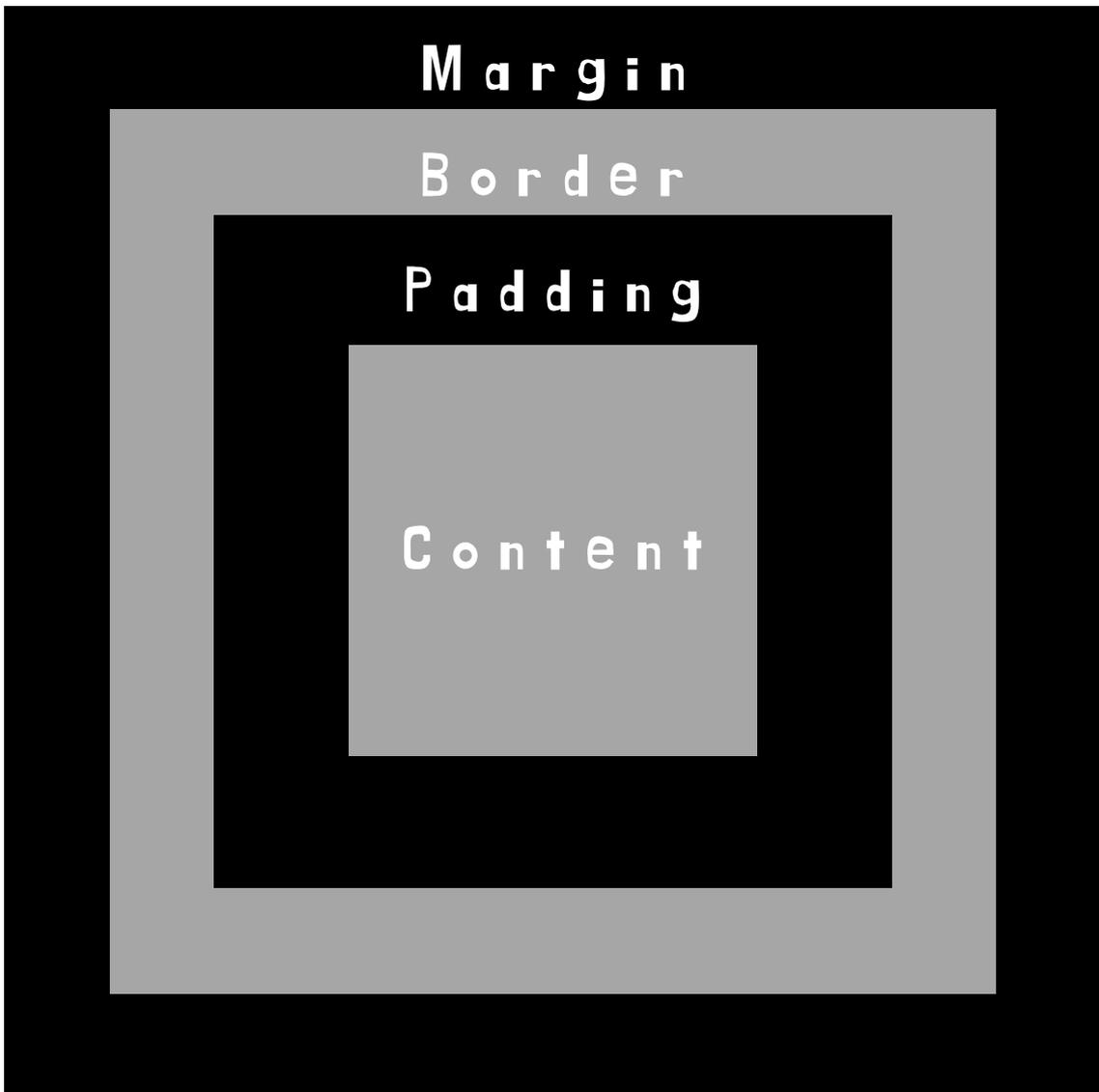
```
.boring-text::after
```



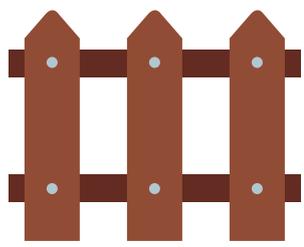


## ¿Qué es?

Nos permite posicionar y alinear diversos tipos de elementos. Es fundamental comprender que todo en CSS es una caja.



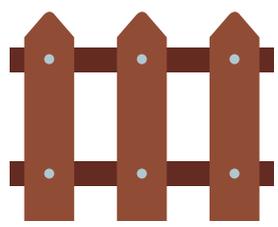
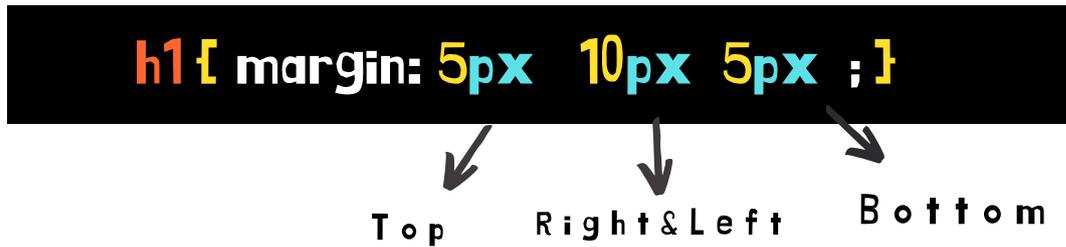
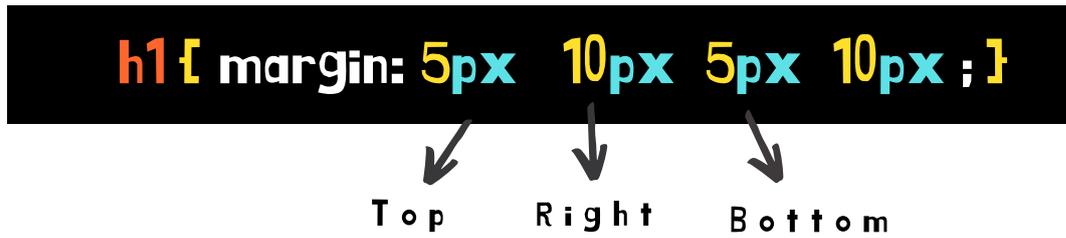
@irisval06



# Margin

Establece la **anchura** de **algunos o todos los márgenes de los elementos**. Establece el margen para los **cuatro lados**. Es una **abreviación** para evitar tener que establecer cada lado **por separado** con las otras propiedades de margen:

**margin-top**, **margin-right**, **margin-bottom**, **margin-left**.  
Se permiten valores **negativos**



## Border

La propiedad de borde **envuelve el contenido** y **cualquier relleno**. Su tamaño y estilo se pueden modificar y usar mediante **border** y las propiedades relacionadas. Un ejemplo puede ser `border-radius`.



```
border-radius: 10px;
```

## Padding

La propiedad `padding` establece el **espacio de relleno** requerido por **todos los lados** de un **elemento**. El área de **padding** es el **espacio** entre el **contenido** del elemento y su **borde**. La propiedad `padding` es un atajo para evitar la asignación de cada lado por separado (`padding-top`, `padding-right`, `padding-bottom`, `padding-left`). No se permiten valores **negativos**.

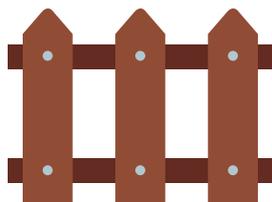
```
padding: 5%; /*Aplica 5% de padding en todos los lados*/
```

```
padding: 10px; /*Aplica 10px de padding en todos los lados */
```

```
padding: 10px 20px; /* Arriba y abajo, 10px de padding, izquierda y derecha, 20px de padding */
```



@irisval06

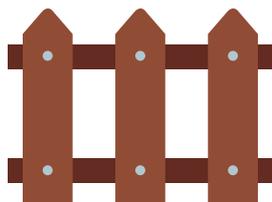


# Content

Es el área donde se **muestra el contenido**, su tamaño **puede cambiarse** utilizando las propiedades ya vistas como **width** y **height**.



@irisval06





# Unidades de medida

