

## Contenido

Técnicas para combinar selectores de CSS y la utilización de herencias. ....	2
¿Qué es un selector? .....	2
Combinación de Selectores .....	3
Prevalencias entre selectores CSS. ....	3
Especificidad .....	3
Efecto cascada .....	3
Tipos de Selectores .....	4
Reglas de especificidad .....	4
Tipo de framework de CSS .....	5
Bibliografía .....	6

# Técnicas para combinar selectores de CSS y la utilización de herencias.

## ¿Qué es un selector?

Los selectores son referencias a los elementos HTML a los cuales se van a aplicar reglas CSS.

### Se clasifican en:

- **Selector de tipo:** Selecciona todos los elementos del tipo que se ha definido en el selector.  

```
div {  
    //El estilo se aplicará a todos los elementos div  
}
```
- **Selector de Clase:** Selecciona todos los elementos que tienen la clase selector.  

```
.ejemplo {  
    // El estilo se aplicará a todos los elementos que tengan la clase  
    "ejemplo".  
}
```
- **Selector de ID:** Selecciona el elemento que tenga la ID, que es única, definido en el selector.  

```
#ejemplo {  
    // El estilo se aplica al elemento con el id "ejemplo".  
}
```
- **Selector universal:** Selecciona todos los elementos.  

```
* {  
    // El estilo se aplica a todos los elementos.  
}
```
- **Selector de atributo:** Seleccionan los elementos que tengan el atributo definido en el selector.  

```
[attr] {  
    // El estilo se aplica a todos los elementos con ese atributo.  
}  
[attr=value] {  
    // El estilo se aplica a todos los elementos con ese atributo y valor.  
}
```

Existen otros más que son los selectores de Pseudo-clases y Pseudo-elementos, que dan estilo a los elementos HTML según el estado del elemento y a una parte del elemento, respectivamente.

## Combinación de Selectores

Con los selectores anteriores se pueden realizar selecciones más complejas usando “combinadores”, que son operadores que se utilizan entre selectores.

### Combinador de hermanos adyacentes

Selecciona hermanos adyacentes, dos elementos que comparten padre y uno se encuentra a lado del otro. El combinador es el símbolo **+**.

```
h1 + p {  
    // El estilo se aplica a todos los p que estén inmediatamente a  
    // continuación de un h1.  
}
```

### Combinador general de hermanos

Selecciona hermanos, sin necesidad de que uno siga a otro. El combinador es el símbolo **~**.

```
h1 ~ p {  
    // El estilo se aplica a los p que son hermanos del h1 que les precede.  
}
```

### Combinador de hijo

Selecciona los hijos directos del primer elemento. El combinador es el símbolo **>**.

```
div > p {  
    // El estilo se aplica a los p que son hijos de un div  
}
```

### Combinador de descendientes

Selecciona todos los elementos descendientes del primer elemento. El combinador es un espacio.

```
div li {  
    // El estilo se aplica a los li que estén dentro de un div, da igual si son hijos directos o  
    // no.  
}
```

## Prevalencias entre selectores CSS.

**Especificidad:** Es la manera mediante la cual los navegadores deciden qué valores de una propiedad CSS son más relevantes para un elemento y serán aplicados.

**Efecto cascada:** Si hay múltiples reglas para la misma clase, ID o etiqueta, la última gana.

```
#miTexto {  
  color: blue;  
}
```

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

El texto será rojo, porque tienen la misma clase pero la última regla gana.

## Tipos de Selectores

**Diferentes tipos de selectores** tienen diferentes especificidades. Un selector de ID tiene una especificidad más alta que un selector de clase, por ejemplo. De hecho, un solo ID tiene una especificidad más alta que un selector con cualquier número de clases. De manera similar, un selector de clase tiene una especificidad más alta que un selector de etiqueta (también llamado selector de tipo).

## Reglas de especificidad

Las reglas exactas de especificidad son:

1. Los estilos en línea tienen mayor especificidad.
2. Si un selector tiene más IDs, gana.
3. Si eso resulta en un empate, el selector con más clases gana.
4. Si eso resulta en otro empate, el selector con más nombres de etiquetas gana.

Para CSS considera los selectores que se muestran en la siguiente lista. Estos están escritos en orden de especificidad creciente.

```
html body header h1 {  ◀ ① Four tags  
  color: blue;  
}  
  
body header.page-header h1 {  ◀ ② Three tags and  
  color: orange;                one class  
}  
  
.page-header .title {  ◀ ③ Two classes  
  color: green;  
}  
  
#page-title {  ◀ ④ One ID  
  color: red;  
}
```

El selector más específico aquí es **4**, con un ID, por lo que su declaración de color rojo se aplica al título. El siguiente en especificidad es **3**, con dos nombres de clase. Este se aplicaría si el selector de ID estuviera ausente. El selector **3** tiene una especificidad más alta que el selector **2**, a pesar de su longitud: dos clases son más específicas que una

clase. Finalmente, **1** es el menos específico, con cuatro tipos de elementos (es decir, nombres de etiquetas) pero sin IDs ni clases.

## Tipo de framework de CSS

Un framework de CSS es un conjunto de herramientas, estilos y reglas predefinidas que facilitan el desarrollo de sitios web sin escribir todo el CSS desde cero. Estos frameworks incluyen sistemas de grillas, estilos básicos, componentes preconstruidos y utilidades para hacer el desarrollo más rápido y eficiente.

Su aplicación en un proyecto permite ahorrar tiempo, crear una consistencia visual, mejorar el diseño responsivo y se aprovechan clases predefinidas ahorrando crearlas manualmente.

La elección de un framework depende de las necesidades específicas del proyecto, considerando sus ventajas y desventajas antes de utilizarlo.

Existen varios tipos de framework de CSS, de las cuales citamos:

- **Bootstrap:** es un framework de diseño responsivo, fue desarrollado por twitter y lanzado al público en 2011. Dentro de su núcleo incluye HTML, SASS y Javascript para dotarlo de funcionalidades y componentes.
  - **Ventajas:** Es soportado por todos los navegadores, tiene una comunidad que lo respalda, cuenta con componentes listos para diferentes funcionalidades, es fácil de usar y ofrece servicio gratuito.
  - **Desventajas:** No permite reescribir las clases ya existentes de forma sencilla, es muy usado y crea un ambiente de monotonía entre páginas, el proyecto final crea archivos muy pesados.
- **Foundation:** Foundation es un framework diseñado para la accesibilidad y la adaptabilidad en cualquier dispositivo. Es altamente personalizable y modular, ideal para proyectos que requieren flexibilidad.
  - **Ventajas:**
    - ✓ Fácil de usar con CLI incluido,
    - ✓ Modular y extensible.
    - ✓ Ofrece componentes y plugins de JavaScript.
  - **Desventajas:**
    - Comunidad más pequeña en comparación con otros frameworks.
    - Complejidad y curva de aprendizaje elevada.
- **Tailwind CSS:** Tailwind CSS es un framework utilitario que genera estilos basados en clases definidas en los archivos HTML y JavaScript. Es conocido por su flexibilidad y eficiencia.
  - **Ventajas:**

- ✓ Alta productividad y código reutilizable.
- ✓ Ligero y altamente configurable.
- ✓ No requiere tiempo de ejecución.
- **Desventajas:**
  - Curva de aprendizaje pronunciada.
  - Requiere configuración previa.
- **Bulma:** Bulma es un framework gratuito y de código abierto, ideal para principiantes. Ofrece una librería de componentes para interfaces responsivas.
  - **Ventajas:**
    - ✓ Clases fáciles de identificar.
    - ✓ Curva de aprendizaje plana.
    - ✓ Popular y sencillo de usar.
  - **Desventajas:**
    - No soporta navegadores antiguos.
    - Puede carecer de funcionalidades avanzadas.
- **Semantic UI:** Semantic UI se enfoca en la creación de un vocabulario compartido para interfaces de usuario, basado en principios de lenguaje natural. Es altamente flexible y responsivo.
  - **Ventajas:**
    - ✓ Accesible y bien documentado.
    - ✓ Integración con bibliotecas como Angular y React.
    - ✓ Convenciones de codificación estrictas.
  - **Desventajas:**
    - Curva de aprendizaje empinada.
    - Reglas estrictas que pueden ser difíciles de seguir.

## Bibliografía

1. Selectores: [https://ahorasomos.izertis.com/autentia/wp-content/uploads/libros/Front\\_HTMLyCSS-Autentia.pdf](https://ahorasomos.izertis.com/autentia/wp-content/uploads/libros/Front_HTMLyCSS-Autentia.pdf)
- ✓ Prevalencia entre selectores:
  - [https://developer.mozilla.org/es/docs/Web/CSS/CSS\\_cascade/Specificity](https://developer.mozilla.org/es/docs/Web/CSS/CSS_cascade/Specificity)
  - <https://dl.ebooksworld.ir/motoman/Manning.CSS.in.Depth.www.EBooksWorld.ir.pdf>, pág 11
2. Frameworks CSS: <https://platzi.com/blog/los-5-mejores-frameworks-de-css-para-tus-proyectos/>