



CSS | Advanced Selectors

Objetivos de la clase

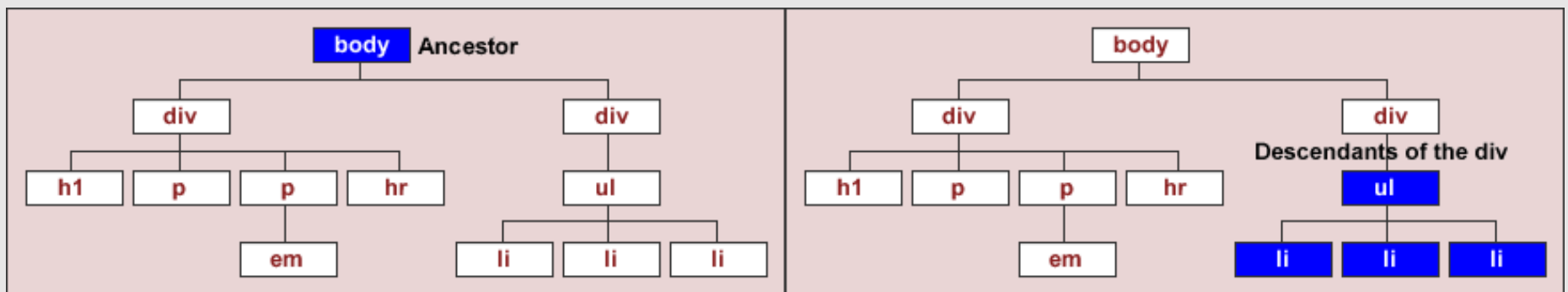
- ✓ Seleccionar elementos usando selectores básicos.
- ✓ Seleccionar elementos combinando diferentes selectores.
- ✓ Seleccionar elementos en función de su relación con otros elementos.
- ✓ Seleccionar elementos en función de sus atributos.

Document Tree

Todos los documentos HTML se pueden representar como árboles.

Por ejemplo, `<body>` es el **antecesor** de todos los demás elementos que se muestran.

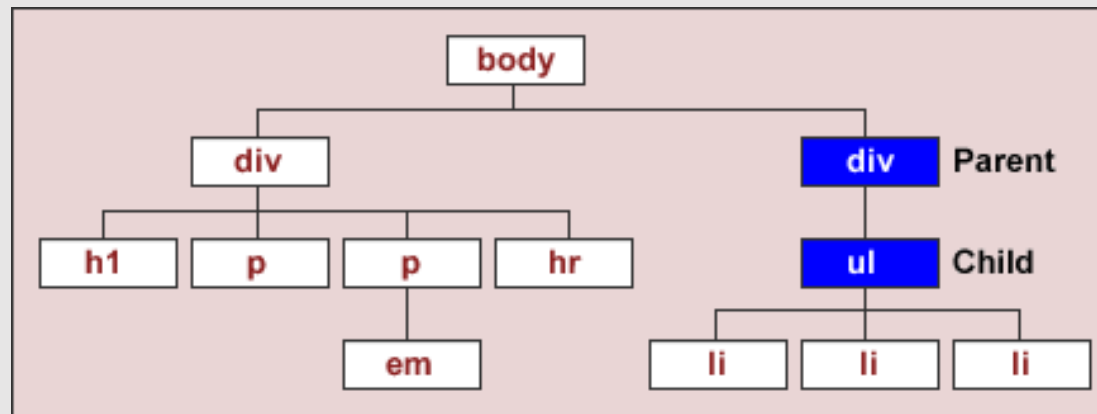
Por el contrario, un **descendiente** se refiere a cualquier elemento que esté conectado pero más abajo en el árbol de documentos, sin importar cuántos niveles sean más bajos.



Document Tree

Un **padre** es un elemento que está directamente arriba y conectado a un elemento en el árbol del documento.

Un **hijo** está directamente debajo y conectado a un elemento en el árbol de documentos.



```
<ul>
  <li class="menu-item"></li>
  <li class="menu-item"></li>
</ul>
```

* es el padre
de cada .
Y los elementos
 son sus hijos.*



¿Recuerdan cuales son los selectores simples?

¿Recuerdan cuales son los selectores simples?

- ✓ Selector de tipo

Selecciona todos los elementos que coinciden con el nombre de nodo dado.

Ej: `<input>`

- ✓ Selector de clase

Selecciona todos los elementos que tienen el atributo de clase dado. Ej: `.test`

- ✓ Selector de ID

Selecciona un elemento en función del valor de su atributo id. Ej: `#test`

- ✓ Selector universal

Selecciona todos los elementos del documento. Ej: `*`

- ✓ * Selector de atributos

Selecciona elementos en función del valor del atributo dado. Ejemplo: `[autoplay]`

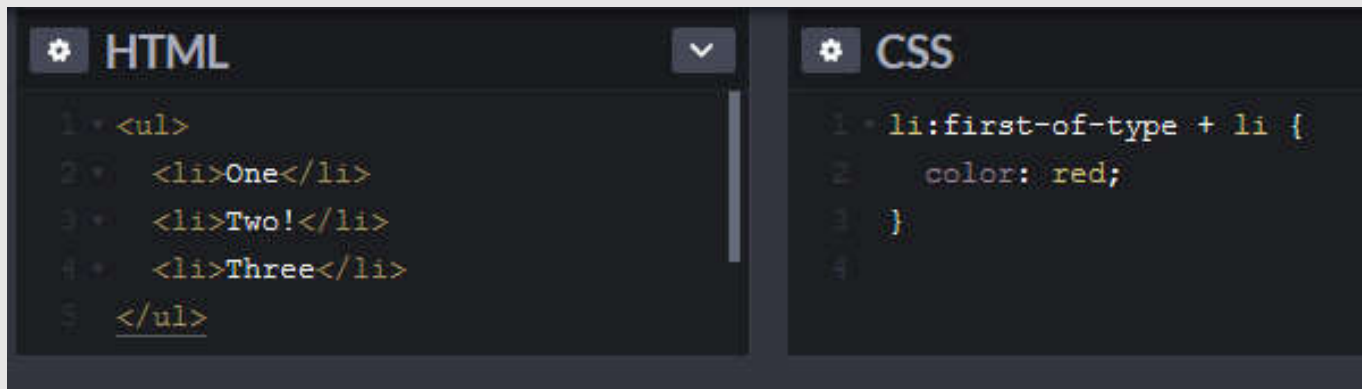
Selectores combinados

✓ Adjacent sibling combinator (+)

Separa dos selectores y hace coincidir el segundo elemento solo si sigue inmediatamente al primer elemento, y ambos son hijos del mismo elemento padre.

Ejemplo:

```
formerElement + targetElement {  
    property: value;  
}
```



```
HTML  
1 <ul>  
2   <li>One</li>  
3   <li>Two!</li>  
4   <li>Three</li>  
5 </ul>  
  
CSS  
1 li:first-of-type + li {  
2   color: red;  
3 }  
4
```

- One
- Two!
- Three

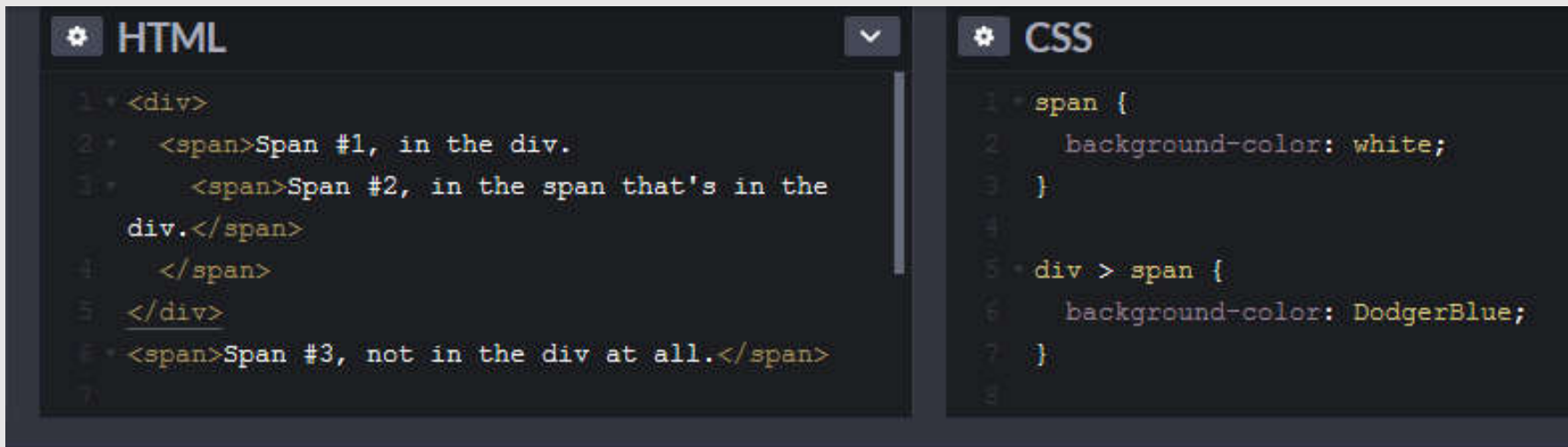
Selectores combinados

✓ Child combinator (>)

Selecciona nodos que son hijos directos del primer elemento.

Sintaxis: A > B

Ejemplo:



```
HTML
1 <div>
2   <span>Span #1, in the div.
3   <span>Span #2, in the span that's in the
    div.</span>
4 </span>
5 </div>
6 <span>Span #3, not in the div at all.</span>
7

CSS
1 span {
2   background-color: white;
3 }
4
5 div > span {
6   background-color: DodgerBlue;
7 }
8
```

Span #1, in the div. Span #2, in the span that's in the div.
Span #3, not in the div at all.

Selectores combinados

✓ Descendant combinator

Selecciona los nodos que son descendientes del primer elemento. Sintaxis: A B

Ejemplo:

```
selector1 selector2 {  
    property: value;  
}
```

The screenshot shows a code editor with two panels. The left panel is titled 'HTML' and contains the following code:

```
1 <div>  
2   <span>Span 1.  
3   <span>Span 2.</span>  
4   </span>  
5 </div>  
6 <span>Span 3.</span>
```

The right panel is titled 'CSS' and contains the following code:

```
1 span { background-color: white; }  
2 div span { background-color: blue; }  
3
```

Span 1. Span 2
Span 3.

Selectores combinados

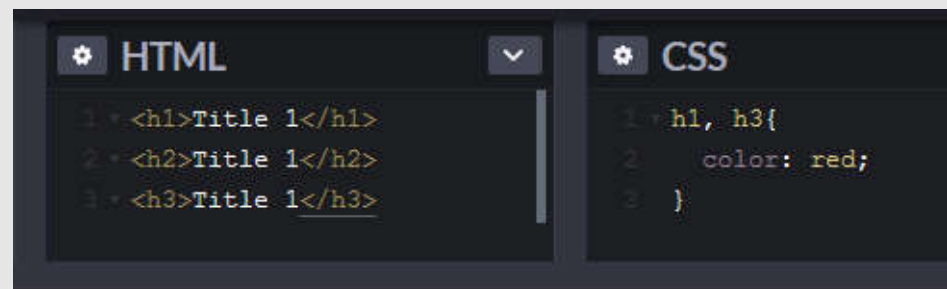
✓ Multiple selection

Utilizado para aplicarle los mismos estilos a varios elementos a la vez. Los separamos con una coma.

Sintaxis: selector1, selector2

Ejemplo:

```
selector1, selector2 {  
    property: value;  
}
```



Title 1

Title 1

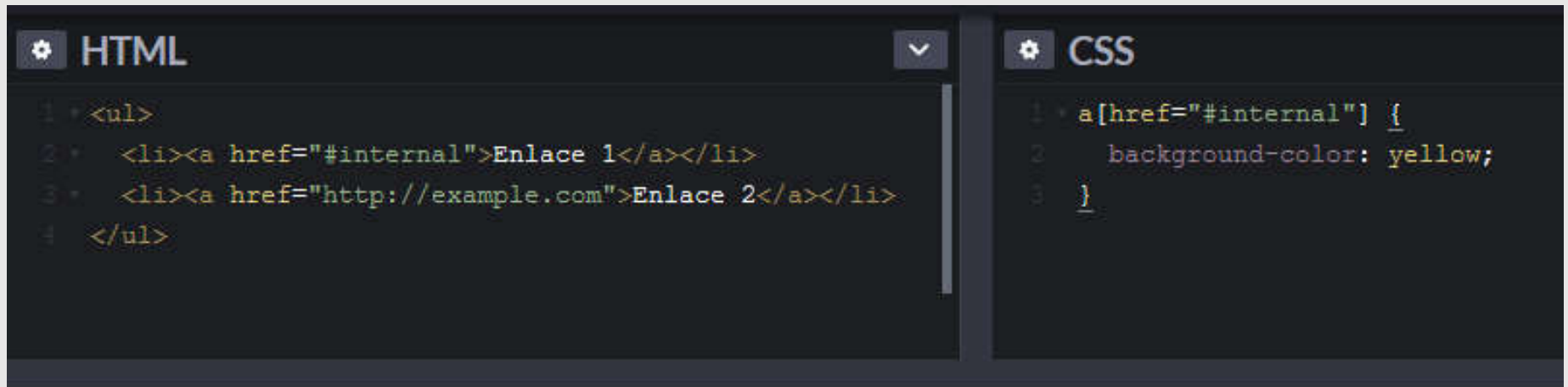
Title 1

Selectores combinados

✓ Attribute Selector

Seleccionar elementos por nombre de sus atributos.

```
element[attr-name="value"] {  
    property: value;  
}
```



The screenshot shows a code editor with two panels. The left panel, titled 'HTML', contains the following code:

```
1 <ul>  
2 <li><a href="#internal">Enlace 1</a></li>  
3 <li><a href="http://example.com">Enlace 2</a></li>  
4 </ul>
```

The right panel, titled 'CSS', contains the following code:

```
1 a[href="#internal"] {  
2     background-color: yellow;  
3 }
```

- [Enlace 1](#)
- [Enlace 2](#)


¿Qué párrafo/s tendrá/n fondo
amarillo?

```
<!DOCTYPE html>
<html>
<head>
  <style>
    div > p {
      background-color: yellow;
    }
  </style>
</head>
<body>

<div>
  <p>Paragraph 1</p>
  <p>Paragraph 2</p>
  <section>
    <p>Paragraph 3</p>
  </section>
  <p>Paragraph 4</p>
</div>

<p>Paragraph 5</p>
<p>Paragraph 6</p>

</body>
</html>
```



Importante: mantener el HTML lo más ordenado y compacto posible. Estos selectores te ayudarán a seleccionar elementos sin tener que agregar clases en todas partes

Resumiendo: Aprendimos...

- ✓ Cómo seleccionar descendientes e hijos
- ✓ Cómo seleccionar hermanos (selector1 + selector2) de un nodo
- ✓ A aplicar estilos a varios elementos a la vez
- ✓ Cómo seleccionar elementos en función de sus valores de atributo elemento [attr = "value"]