

# CIP FP ceheste

Diseño de Interfaces Web

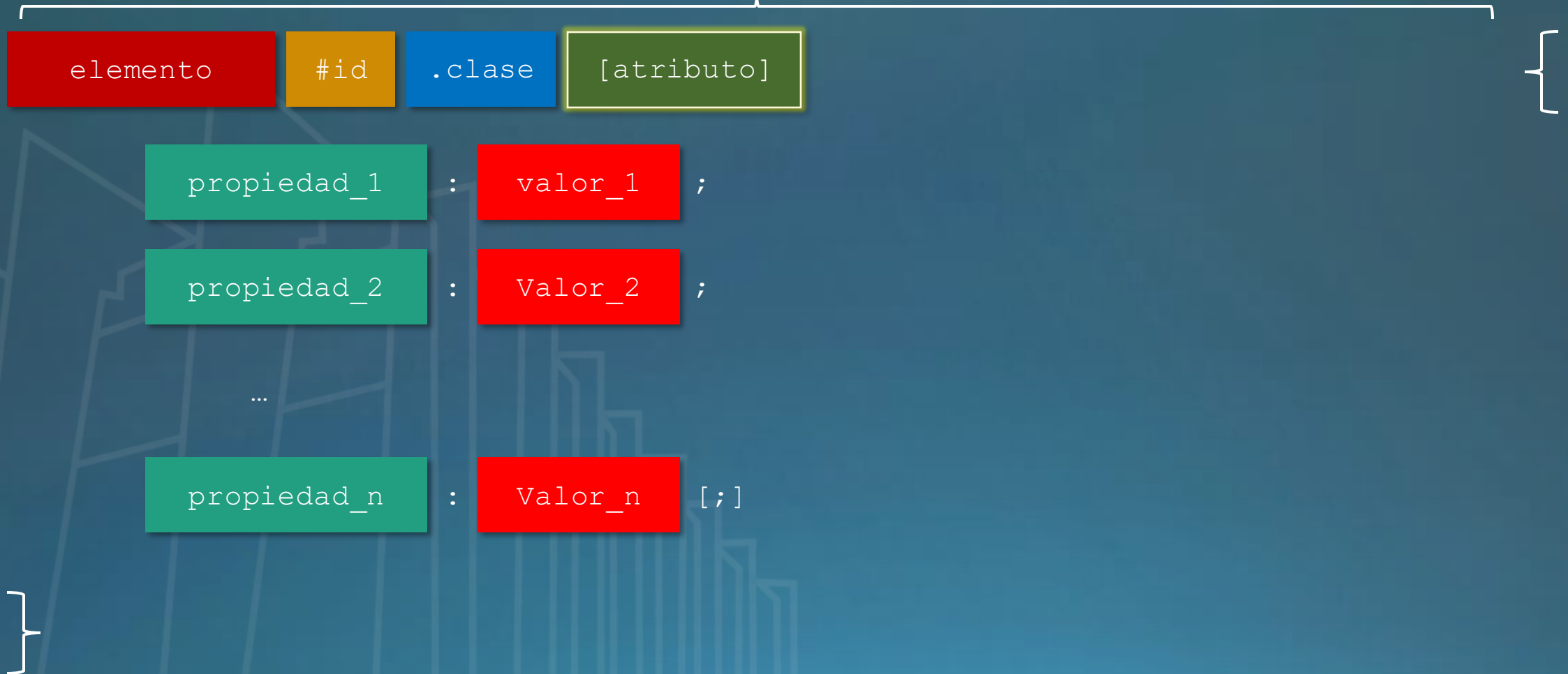
## CSS [2]

Selectores avanzados y más

# CSS

## Selectores avanzados

Selector



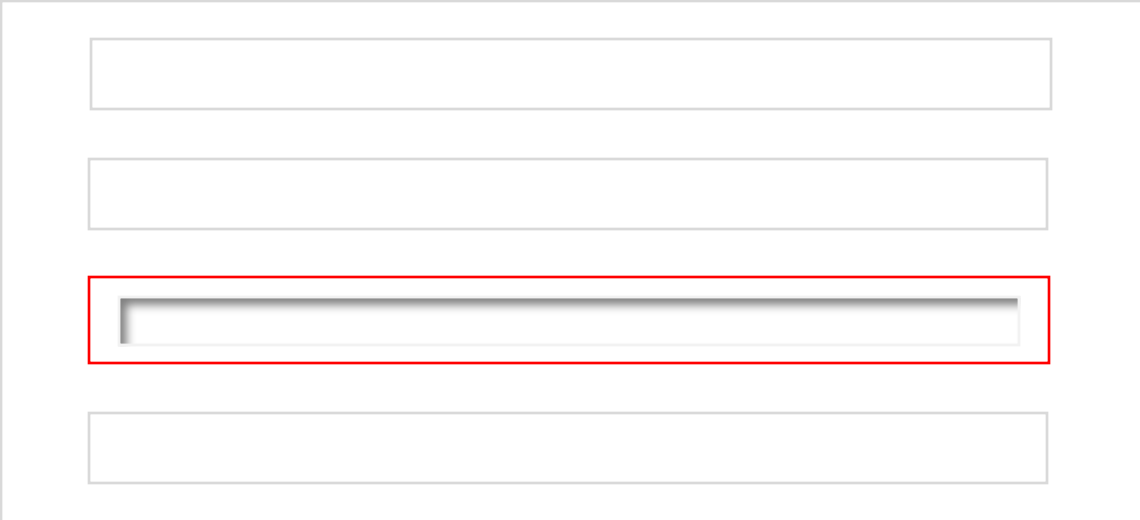
# Selectores

## Selectores avanzados

el **[attr]**: Elementos que tienen un determinado atributo.

```
article[style] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article></article>  
    <div id="div2" class="movidas">  
        </div>  
    → <article style="color:white">  
        <input type="number">  
    </article>  
    <div class="movidas"></div>  
</div>
```



# Selectores

## Selectores avanzados

**e[attr=valor]:** Elementos que tienen un determinado atributo con un valor.

```
input[type="number"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article></article>  
    <div id="div2" class="movidas">  
    </div>  
    <article style="color:white">  
        → <input type="number">  
    </article>  
    <div class="movidas"></div>  
</div>
```



# Selectores

## Selectores avanzados

**e[atr\*=val]:** Elementos cuyo valor del atributo `atr` contiene la cadena `val`.

```
*[class*="vi"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article></article>  
→ <div id="div2" class="movidas">  
    </div>  
    <article style="color:white">  
        <input type="number">  
    </article>  
→ <p class="movidas">parrafo</p>  
</div>
```

Visual representation of the HTML structure. The container `div1` contains an `article` and a `div2`. `div2` contains another `article` and a `p` with class `movidas`. The `article` inside `div2` contains a number input. The `p` contains the text "parrafo". The red border highlights the elements selected by the CSS selector `[class*="movidas"]`.

# Selectores

## Selectores avanzados

**e[attr~=val]**: Elementos cuyo val está en una lista en attr.

```
*[class~="dos"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article></article>  
    <div id="div2" class="movidas">  
    </div>  
    <article style="color:white">  
        <input type="number">  
    </article>  
→ <p class="uno dos">parrafo</p>  
</div>
```

Visual representation of the HTML structure. A container box contains three stacked input fields, a progress bar, and a paragraph labeled "parrafo". The paragraph is highlighted with a red border, corresponding to the 'e[attr~=val]' selector where val is 'dos'.

# Selectores

## Selectores avanzados

**el[attr|=val]:** Tiene un atributo `attr` con un valor `val`, o es uno de una lista separada por "-".

```
*[class|="uno"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article></article>  
    <div id="div2" class="movidas">  
    </div>  
    <article style="color:white">  
        <input type="number">  
    </article>  
    <p class="uno dos">parrafo</p>  
</div>
```

The visual representation shows a large container box. Inside, there are four elements stacked vertically: 1. An empty rectangular box representing the first article. 2. A second empty rectangular box representing the 'div2' with class 'movidas'. 3. A rectangular box representing the second article with white text; it contains a horizontal number input field. 4. A rectangular box representing the paragraph 'parrafo' with the text 'parrafo' inside.

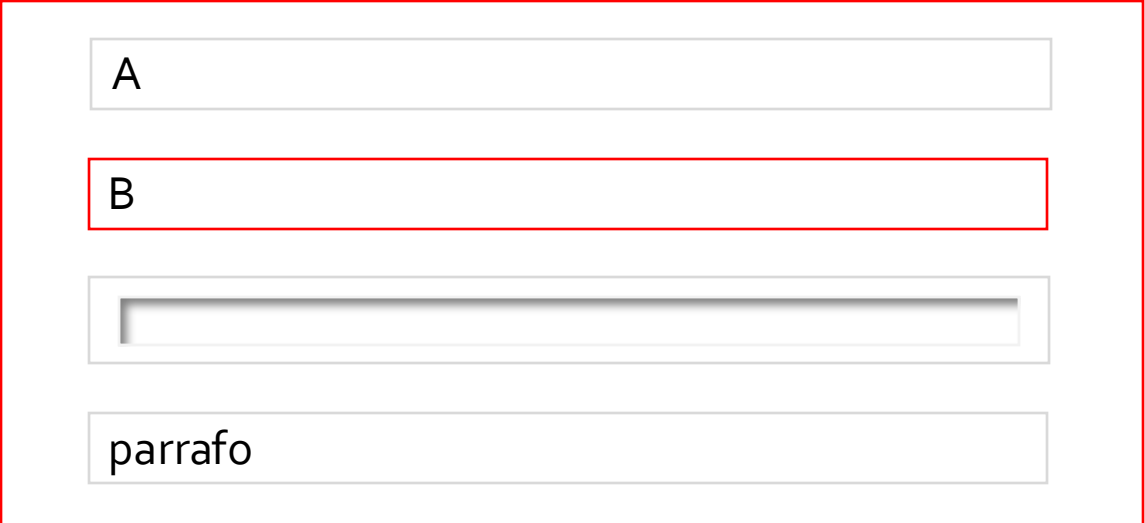
# Selectores

## Selectores avanzados

**el[attr^=val]:** Si tiene un atributo `attr` con un valor que empieza por `val`.

```
*[id^="div"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
→ <div id="div1">  
    <article>A</article>  
→ <div id="div2" class="movidas">B  
    </div>  
    <article style="color:white">  
        <input type="number">  
    </article>  
    <p class="uno dos">parrafo</p>  
</div>
```



A

B

parrafo



# Selectores

## Selectores avanzados

**el[attr\$=val]:** Si tiene un atributo `attr` con un valor que termina por `val`.

```
*[data-movidas$="as"] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

→ `<div id="div1" data-movidas="muchas">`  
    `<article>A</article>`  
    `<div id="div2" class="movidas">B`  
    `</div>`  
    `<article style="color:white">`  
        `<input type="number">`  
    `</article>`  
    `<p class="uno dos">parrafo</p>`  
`</div>`

A

B

parrafo

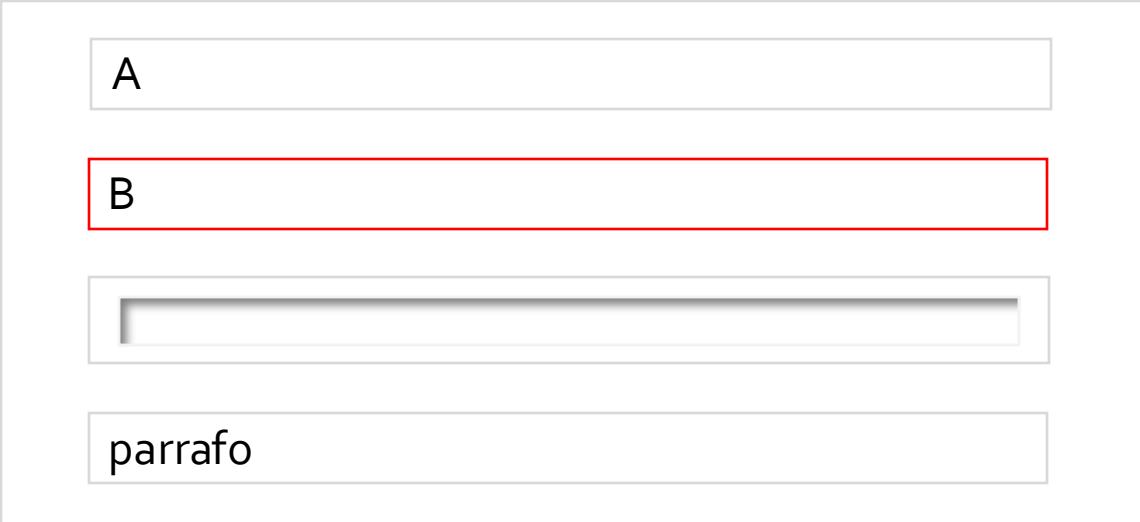
# Selectores

## Selectores avanzados

`el[attr operador val i]`: Indica que **NO** se distinguirá entre mayúsculas y minúsculas.

```
*[class$="AS" i] {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1">  
    <article>A</article>  
→ <div id="div2" class="movidas">B  
    </div>  
    <article style="color:white">  
        <input type="number">  
    </article>  
    <p class="uno dos">parrafo</p>  
</div>
```



A

B

parrafo

# Selectores

## Variables

Referencia a la raíz del documento. Variables Globales. Se puede escoger otro ámbito.

**--nombre-variable:** Nos permite declarar variables para poder utilizar en el futuro.

```
:root {  
  /* Bloque de declaración; */  
  --toGuapo: red;  
}
```

```
*[class$="AS" i] {  
  /* Bloque de declaración; */  
  border: 1px solid var(--toGuapo, green);  
}
```

```
<div id="div1">  
  <article>A</article>  
  <div id="div2" class="movidas">B  
  </div>  
  <article style="color:white">  
    <input type="number">  
  </article>  
  <p class="uno dos">parrafo</p>  
</div>
```

A

B

parrafo

# CSS

## Selectores avanzados

Selector



propiedad\_1 : valor\_1 ;

propiedad\_2 : Valor\_2 ;

...

propiedad\_n : Valor\_n [;]



# Pseudoselectores

## Selectores avanzados

### Pseudoclases

Estado o situación

<code>:active</code>	<code>:required</code>
<code>:checked</code>	<code>:visited</code>
<code>:disabled</code>	<code>:last-child</code>
<code>:empty</code>	<code>:first-child</code>
<code>:enabled</code>	<code>:link</code>
<code>:focus</code>	<code>:nth-child(x)</code>
<code>:hover</code>	<code>:nth-last-child(x)</code>

### Pseudoelementos

Parte concreta del  
contenido del elemento

`::after`  
`::before`  
`::first-letter`  
`::first-line`  
`::selection`

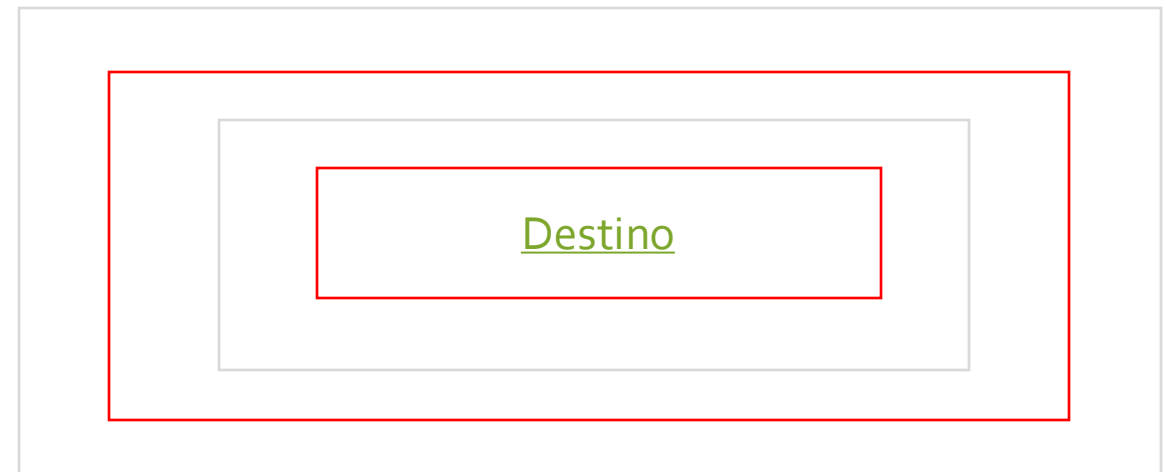
# Selectores

## Selectores avanzados

**Selector de descendientes:** Elementos por debajo de otro elemento.

```
#div1 div{  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1"> Contenido  
→ <div id="div2" class="movidas">  
    <article>  
        → <div class="movidas">  
            <a  
href="URL">Destino</a>  
            </div>  
        </article>  
    </div>  
</div>
```



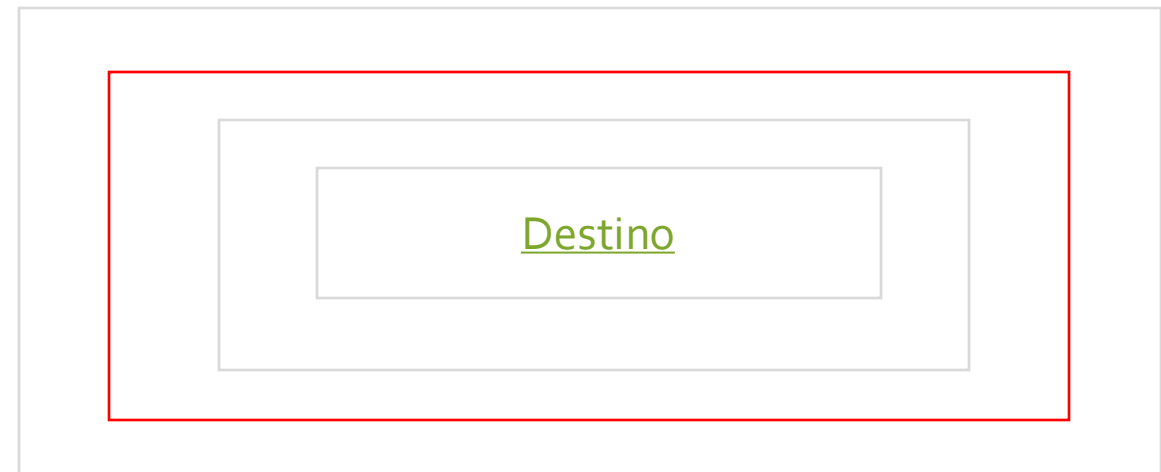
# Selectores

## Selectores avanzados

**Selector de descendientes directos (hijos):** Elementos hijos de otros.

```
#div1 > div{  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1"> Contenido  
→ <div id="div2" class="movidas">  
    <article>  
        <div class="movidas">  
            <a  
href="URL">Destino</a>  
        </div>  
    </article>  
</div>  
</div>
```



# Selectores

## Selectores avanzados

**Selectores adyacentes:** Elementos que son contiguos y el 1º precede al 2º.

```
#div2 + article{  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1"> Contenido  
    <article></article>  
    <div id="div2" class="movidas">  
        </div>  
→ <article>  
    <div class="movidas"></div>  
    </article>  
    <article></article>  
</div>
```





# Selectores

## Selectores avanzados

**Selectores hermanos:** Elementos que comparten padre y el 1º precede al 2º.

```
#div2 ~ .movidas {  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1"> Contenido  
  <article class="movidas"></article>  
  <div id="div2" class="movidas">  
  </div>  
  <article>  
    <div class="movidas"></div>  
  </article>  
  <div class="movidas"></div>  
</div>
```



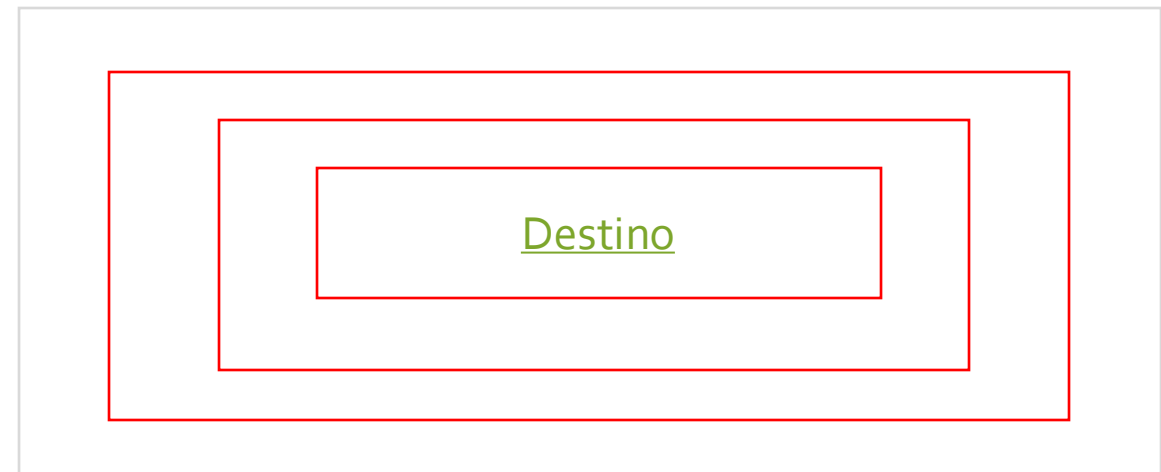
# Selectores

## Selectores avanzados

**Combinación de selectores:** Varios selectores con las mismas propiedades

```
#div1 div, article{  
    /* Bloque de declaración; */  
    border: 1px solid red;  
}
```

```
<div id="div1"> Contenido  
→ <div id="div2" class="movidas">  
    → <article>  
        → <div class="movidas">  
            <a  
href="URL">Destino</a>  
            </div>  
        </article>  
    </div>  
</div>
```



# Selectores

## Selectores avanzados

**Nesting:** Varios selectores pueden anidarse dependiendo los interiores de los exteriores.

```
#div1 {  
  /* propiedades y/o reglas; */  
  border: 1px solid red;  
  [&] .movidas{  
    /* propiedades y/o reglas; */  
    border: 1px solid green;  
  }  
}
```

```
→ <div id="div1">  
  → <div id="div2" class="movidas">  
    <article>  
      → <div class="movidas">  
        </div>  
      </article>  
    </div>  
  </div>
```



## CSS

## Selectores avanzados

## Cascada

Cuando existen varias propiedades iguales sobre un mismo elemento, siempre se aplica la última que aparece.

```
<div> Contenido </div>
```

```
-----
```

```
archivo.css
```

```
-----
```

```
div {  
    background-color:blue;  
}
```

```
div {  
    background-color:red;  
}
```

```
<style>  
    div {  
        background-color:pink;  
    }  
</style>
```

```
<link src="archivo.css"  
rel="stylesheet"  
type="text/css">
```

```
<link src="archivo.css"  
rel="stylesheet"  
type="text/css">
```

```
<style>  
    div {  
        background-  
color:pink;  
    }
```

```
</style>  
<div style="background-  
color:green">cont</div>
```

## CSS

## Selectores avanzados

## Especificidad

Cuando existen varias propiedades iguales sobre un mismo elemento, siempre se aplica la más específica.

```
<div id="mola" class="mola"> Contenido </div>
```

```
-----
```

```
archivo.css
```

```
-----
```

```
#mola {  
    background-color:blue;  
}
```

```
div {  
    background-color:red;  
}
```

```
.mola {  
    background-color:red;  
}
```



id

atributo

class

elemento



\*

## CSS

## Selectores avanzados

## important!

La propiedad declarada con `important!` sobrescribe cualquier otra propiedad.

```
<link src="archivo.css"
rel="stylesheet"
type="text/css">
<style>
```

```
  div {
    background-color: pink
    important!;
  }
```

```
</style>
<div style="background-
color: green">cont</div>
```

```
<div id="mola" class="mola"> Contenido </div>
```

```
-----
```

```
archivo.css
```

```
-----
```

```
#mola {
  background-color: blue;
}
```

```
div {
  background-color: red important!;
}
```

```
.mola {
  background-color: red;
}
```

# CSS

Estilos del sitio web [2]

important!

- Siempre intentar especificidad o cascada.
- Solo para sobreescribir CSS externo como Bootstrap.

# Herencia

## Estilos del sitio web [2]

### Propiedades heredables

- Propiedades relativas a la fuente.
- Propiedades relativas al texto.
- Propiedades relativas a listas.
- **inherit.**

Heredará del padre más cercano que tenga definida la propiedad.

¿Si queremos que una propiedad heredable no se herede?

### Propiedades no heredables

- Resto de propiedades.
- **unset**

¿Si queremos que estas propiedades hereden del contendor?