

Taller de HTML & CSS

Sesión 2

Contenido

- Basic CSS
- CSS Selectors
- Cascade and Specificity
- !important
- Pseudo classes
- Box Model
- FlexBox Layout
- Grid Layout

Basic CSS

```
<h1>The title</h1>
```

```
h1 {  
  color: green;  
  font-size: 40px;  
  font-weight: bold;  
  text-decoration: dotted;  
  text-transform: uppercase;  
  border: 5px solid greenyellow;  
}
```

Notamos 3 partes principales:

- h1 es un selector.
- color, font-size, ... son propiedades.
- green, 60px , ... son los valores de las propiedades respectivas.

Basic CSS

Herencia

Existen propiedades CSS que son heredadas de padre a hijo.

La más usuales son las que tienen relación con la fuente.

```
<body>
  <section>
    <div>
      <h1>Herencia en acción</h1>
    </div>
  </section>
</body>
```

```
section {
  color: red;
}
```

Basic CSS

Practiquemos

Agreguemos un poco de estilo a este html.

```
<h1>Taller de CSS</h1>
<h2>Sesión 2</h2>
<p>En esta sesión aprenderemos <span>CSS</span> y los <span>estilos</span> para nuestra página web.</p>
<div>
  <h3>Nixvoid</h3>
</div>
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>Javascript</li>
</ul>
<ol>
  <li>Primero</li>
  <li>Segundo</li>
  <li>Tercero</li>
</ol>
```

CSS Selectors

Los selectores que usaremos son los siguientes:

- IDs (#idName)
- Classes (.className)
- Type (htmlTag) (div)
- Universal Selector (*)
- Attribute Selector ([attribute])
- Group Selector (anteriores selectores separados por comas)

Cascade and Specificity

Cascade

Primer ejemplo

```
div {  
  color: purple;  
}  
div {  
  color: green;  
}  
  
<div >  
  <h1>The title</h1>  
</div>
```

Segundo ejemplo

```
.main-title {  
  border: 1px solid black;  
  color: orange;  
}  
  
.title {  
  border: 1px dotted black;  
  color: green;  
}  
  
<h1 class="title main-title">Main title</h1>
```

Cascade and Specificity

Specificity

```
.main-title.title{  
  border: 1px solid black;  
  color: orange;  
}
```

```
.title {  
  border: 1px dotted black;  
  color: green;  
}
```

```
<h1 class="title main-title">Main title</h1>
```

Link para leer más : [MDN-Specificity](#)

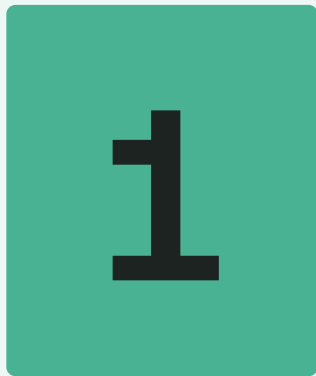
Specificity

inline

ID

classes,
attributes,
pseudo-classes

elements,
pseudo-elements






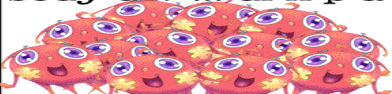



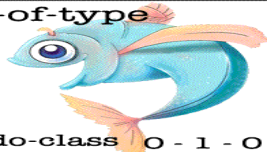











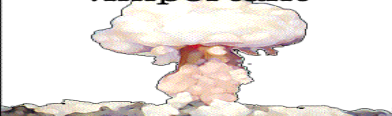
Highest

Lowest



CSS SPECIFISHITY

WITH PLANKTON, FISH AND SHARKS

*  universal selector 0 - 0 - 0	div  1 element 0 - 0 - 1	li > ul  2 elements 0 - 0 - 2	body div ... ul li p a  12 elements 0 - 0 - 12
.myClass  1 class 0 - 1 - 0	*.myClass  1 universal selector 1 class 0 - 1 - 0	[type=checkbox]  1 attribute selector 0 - 1 - 0	:only-of-type  1 pseudo-class 0 - 1 - 0
li.myClass  1 element 1 class 0 - 1 - 1	li[attr]  1 element 1 attribute 0 - 1 - 1	li:nth-of-type(3n)~li  2 elements 1 pseudo-class 0 - 1 - 2	form input[type=email]  2 elements 1 attribute 0 - 1 - 2
li.class:nth-of-type(3n)  1 element 1 class 1 pseudo-class 0 - 2 - 1	input[type]:not(.class)  1 element 1 class 1 attribute 0 - 2 - 1	cl:nth-child(odd)chk[type]...  10 class/attribute/ pseudo-classes 0 - 10 - 0	#myDiv  ID Selector 1 - 0 - 0
#myDiv li.class a[href]  2 types 2 class/attribute 1 ID Selector 1 - 2 - 2	#divitis #myDiv a  2 ID Selectors 1 type selector 2 - 0 - 1	style=""  inline style 1 - 0 - 0	!important  !important 1 - 0 - 0 - 0

X-O-O: The number of ID selectors

O-Y-O: The number of class selectors, attributes selectors, and pseudo-classes

O-O-Z: The number of type selectors and pseudo-elements

*, +, >, ~ : Universal selector and combinators do not increase specificity

:not(x): Negation selector has no value. Argument increases specificity



!important

```
#site-title {  
  color: red;  
  border: 1px dotted black;  
}
```

```
.main-title.title {  
  color: green !important;  
  border: 1px solid greenyellow;  
}
```

```
<h1 id="site-title" class="title main-title">Main title</h1>
```

Pseudo Classes

```
.btn {  
  width: 100px;  
  height: 100px;  
  background-color: yellow;  
  color: red;  
}  
  
.btn:hover {  
  background-color: black;  
  width: 150px;  
  height: 150px;  
  color: white;  
}
```

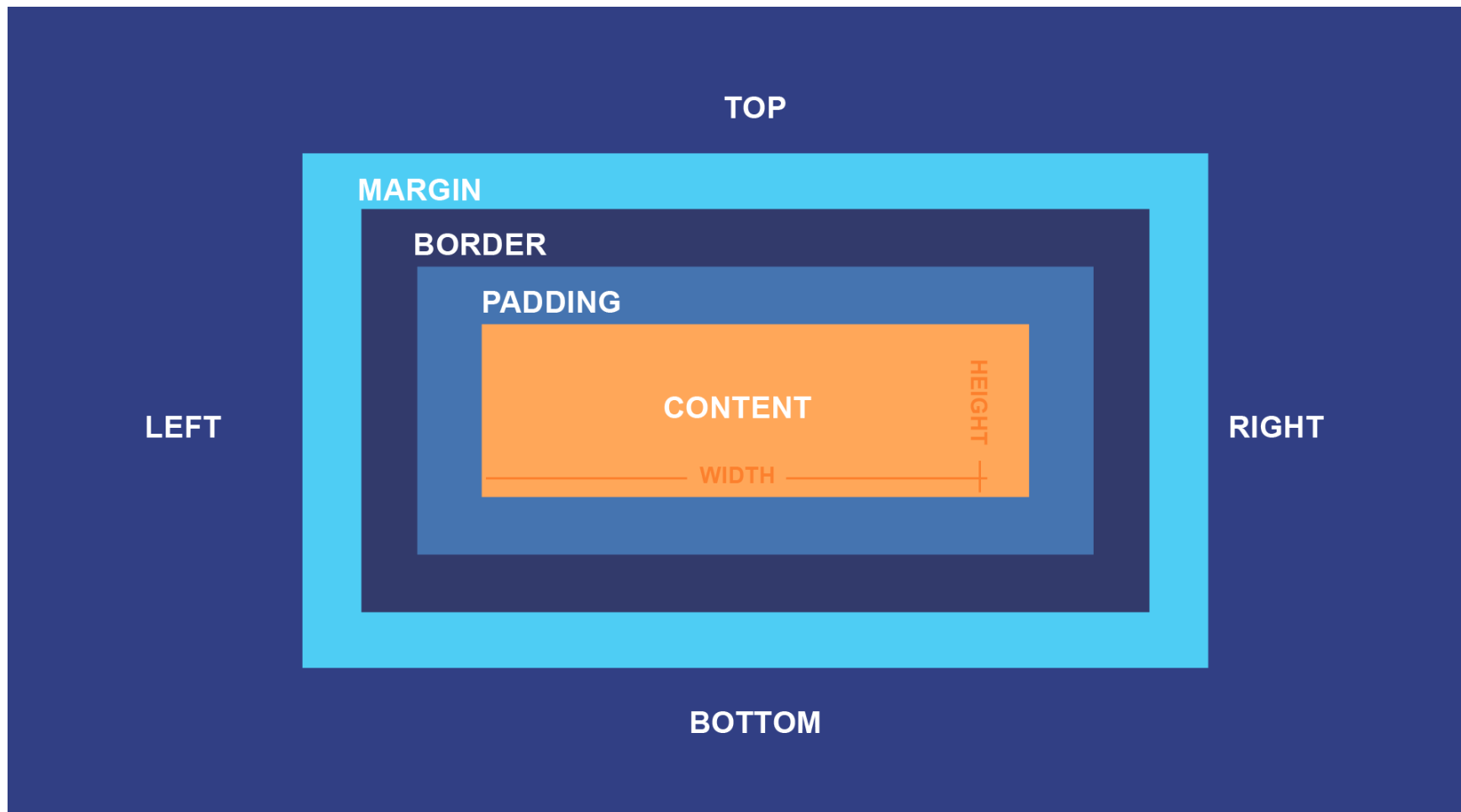
```
<button class="btn">Click me!</button>
```

Indice: MDN-Pseudo-classes

Box Model

El navegador web representa cada elemento como un cuadro rectangular de acuerdo con el CSS Box Model

- Content : esta propiedad se usa para mostrar el texto, las imágenes, etc., que se pueden dimensionar usando la propiedad de ancho y alto.
- Padding : esta propiedad se utiliza para crear espacio alrededor del elemento, dentro de cualquier borde definido.
- Border : esta propiedad se usa para cubrir el contenido y cualquier relleno, y también permite establecer el estilo, el color y el ancho del borde.
- Margin : esta propiedad se utiliza para crear espacio alrededor del elemento, es decir, alrededor del área del borde.



Flexbox

```
.container {  
  display: flex  
}
```

Container:

- flex-direction
- flex-wrap
- justify-content
- align-items
- gap, row-gap, column-gap

Flexbox

```
.item {  
  align-self: flex-end  
}
```

Item:

- order
- flex-grow
- flex-shrink
- flex-basis
- align-self

Recomendación: [CSS-Tricks-flexbox-guide](#)

Grid

```
.container {  
  display: grid;  
  grid-template-columns: 1fr 1fr;  
  grid-template-rows: 100px 1fr max-content;  
}
```

- grid-template-columns
- grid-template-rows
- grid-template-areas
- column-gap
- row-gap
- justify-items
- align-items

Recomendación: [CSS-Tricks-grid-guide](#)