

Módulo 1

LECCIÓN 1

¿Qué es CSS?

Introducción

Es el único lenguaje de estilos que existe, para cualquier web en la que se le quiera dar estilo se utiliza CSS. Sus siglas significan Cascade Style Sheet que traduce hojas de estilos en cascada.

Aunque no es un lenguaje de programación cuenta con funciones propias y variables.

Existen 4 formas para añadir CSS a un documento HTML

- 1 En el head añadir `<style>` código CSS `</style>`
- 2 En línea con la etiqueta `<p style = "código CSS">`
- 3 Añadir o asociando una hoja de estilos externa al HTML `<link rel="stylesheet" href="style.css">`
- 4 Utilizando `@import` dentro de las etiquetas `<style>` `@import url("style.css")</style>`

Sintaxis

```
Selector{
propiedad: valor;
}

body{
background: black;
}
```

A este conjunto entero se le llama regla, una regla puede tener tantas modificaciones del selector como sea necesario. Siempre cerrar una modificación con ; (punto y coma).

Tipos de selectores

Selectores básicos y selectores combinadores, son los más utilizados a la hora de realizar modificaciones y aplicar reglas.

Luego están los selectores de pseudo-elementos y selectores de pseudo-clases, se revisarán más adelante.

Selectores básicos

Existen de varios tipos, entre ellos:

Selectores de etiqueta como por ejemplo:

```
p{
color: red;
}

h1{
font-size: 60px;
}
```

El tema a tener en cuenta aquí es que un selector de etiqueta se aplicará a todas las etiquetas de ese tipo en el documento.

➤ **Selectores de clase:** Son los selectores de las clases que se asignan a los elementos de html, por ejemplo:

```
.title{
color: blue;
}

.text{
font-size: 12px;
}
```

- **Selectores de id:** no se recomienda para dar estilos, se utilizan más para diferenciar componentes y ser utilizados en HTML para las anclas (referencias a diferentes partes del mismo documento) y en JS para identificar específicamente un elemento en el DOM, recalcando que no se debe usar para estilos, un ejemplo sería:

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

- **Selectores universales:** Son selectores que se aplican a todos los elementos del documento, tampoco se recomienda su uso a no ser que se quiera resetear el navegador, un ejemplo sería:

```
*{
color: red;
}
```

- **Selectores de atributos:** Se selecciona a través de un atributo como href por ejemplo:

```
[href]{
color: green;
}
```

Una modificación de este tipo de selector de atributo, sería un selector de atributo-valor que tiene las siguiente variantes.

- **Un atributo apuntando a un valor específico:**

```
[href=#]{
color: green;
}
```

- Un atributo que tenga un valor al menos una vez.

```
[class~="title"]{
    color: red;
}
```

- Un atributo que tenga exactamente el valor indicado o inicie con el valor con guión.

```
[class |= "text"]{
    color: blue;
}
```

- Un atributo que inicie con un valor indicado:

```
[class ^= "title"]{
    font-size: 60px;
}
```

- Un atributo que finalice con el valor indicado:

```
[class $= "text"]{
    font-size: 12px;
}
```

- Un atributo que contenga el valor:

```
[class *= "title"]
{
    color: blue;
}
```

Selectores combinadores

- **Selector de hermano adyacente:** selecciona a un hermano justo debajo.

```
h1 + h2 {  
color: red;  
}
```

- **Selector de hermano general:** buscar todos los hermanos que compartan el mismo padre.

```
h1 ~ h3 {  
color: green;  
}
```

- **Selectores descendentes:** Aplica el estilo a todos los elementos que sean hijos.

```
div span {  
color: blue;  
}
```

- **Selector de hijo directo:** Se aplica al hijo directo del elemento seleccionado, el hijo directo es el que está en nivel 1 descendente:

```
p > span {  
color: red;  
}
```

Herencia

Se obliga a un elemento a heredar la propiedad de su elemento cercano.

```
p{
color: red;
}

a{
color: inherit;
}
```

Cascada

La manera como se leen las hojas de estilos y la forma en cómo se sobreescriben los estilos a medida que se baja en el documento.

```
p{
color: red;
}

a{
color: inherit;
}

p{
color: green;
}
```

Especificidad

Peso de los estilos cuando hay conflicto de estilos.

Etiqueta = 1

Clases y pseudo clases = 10

ID = 100

Estilos en línea = 1000

!important = sobre escribe cualquiera anterior

```
/*style.css*/

#parrafo{
color: blueviolet;
}

.parrafo{
color: red;
}

a{
color: inherit;
}

p{
color: green !important;
}

.parrafo-2{
color: purple;
}
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Especificidad, herencia y
cascada</title>
  <link rel="stylesheet" href="styles-dist.css">
</head>
<body>

<p>Estamos viendo la <a href="#">herencia</a> en
CSS</p>
```

```
<p class="parrafo">Estamos viendo la <a  
href="#">cascada</a> en CSS</p>  
  
<p id="parrafo">Estamos viendo la <a  
href="#">cascada</a> en CSS</p>  
  
<p class="parrafo-2">El párrafo que queremos  
cambiar y no podremos</p>  
</body>  
</html>
```

Conflictos de estilos: important -> especificidad -> cascada

