



El futuro digital
es de todos

MinTIC



Introducción a CSS

OPERADO POR:



Misión
TIC 2022

ruta de aprendizaje 1



Introducción a CSS

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>`

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



¿Qué es CSS?

CSS son las siglas de Cascading Style SheetsCSS describe cómo se mostrarán los elementos HTML en la pantalla, el papel o en otros medios.

CSS se utiliza para definir estilos para sus páginas web, incluido el diseño, el diseño y las variaciones en la visualización para diferentes dispositivos y tamaños de pantalla.

CSS ahorra mucho trabajo. Puede controlar el diseño de varias páginas web a la vez. Las hojas de estilo externas se almacenan en archivos **CSS**.





HTML se creó para describir el contenido de una página web,

Cuando se agregaron etiquetas como **** y atributos de color a la especificación **HTML 3.2**, se inició una pesadilla para los desarrolladores web. El desarrollo de grandes sitios web, donde se agregaban fuentes e información de color a cada página, se convirtió en un proceso largo y costoso.

Para resolver este problema, el Consorcio World Wide Web (W3C) creó **CSS**.

CSS eliminó el formato de estilo de la página **HTML**.

como: **<h1>**
Este es un encabezado **</h1><p>**
Este es un párrafo. **</p>**





Las definiciones de estilo normalmente se guardan en archivos .css externos.

Con un archivo de hoja de estilo externo, puede cambiar el aspecto de un sitio web completo cambiando solo un archivo.

Una regla **CSS** consta de un selector y un bloque de declaración.

El selector apunta al elemento **HTML** al que desea aplicar estilo.

El bloque de declaración contiene una o más declaraciones separadas por punto y coma. Cada declaración incluye un nombre de propiedad **CSS** y un valor, separados por dos puntos.

Varias declaraciones **CSS** se separan con punto y coma y los bloques de declaración están rodeados por llaves.





Sintaxis

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

```
Selector{  
propiedad: valor;  
}  
  
body{  
background: black;  
}
```

Tipos de selectores:

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

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





Selectores básicos:

Existen de varios tipos, entre ellos:

- Selectores de **etiqueta** como por ejemplo:

El tema a tener en cuenta aquí es que un selector de etiqueta se aplicará a todos 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:

```
p{  
  color: red;  
}  
  
h1{  
  font-size: 60px;  
}
```

```
.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) para identificar específicamente un elemento en el DOM, recalcando que no se debe usar para estilos, un ejemplo sería:

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

Podemos utilizar cualquier color predefinido para identificar o diferenciar componentes.





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;  
}
```

Estos serian los selectores universales que podemos utilizar en nuestro desarrollo con **CSS**.





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 sobrescriben 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;
}
```




Conflictos de estilos: important -> especificidad -> cascada

```
<!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>
```



Metodología BEM

Es una forma de estructurar clases CSS de forma sencilla, escalable y reutilizable, es una de las metodologías más utilizadas en el mundo. BEM viene de sus siglas Block, Elemento y Modifier:

- Un bloque (Block) es cualquier elemento autónomo y aislado del documento HTML, puede ser un menú, una galería, un formulario, una sección. Se nombra con una palabra o una palabra seguida de un guión medio

```
.menu  
.menu-principal
```

Un elemento (Element) es cada uno de los elementos del bloque, en un menú es un enlace, en una galería es una foto, en un formulario es un campo de texto o botón, en una sección puede ser el título. Se nombra con el bloque al que pertenece seguido de dos guiones bajos y el nombre del elemento

```
.menu__item  
.menu-principal__link
```




Un modificador (modifier) es cuando un bloque se repite en algún otro lugar de la web pero con modificación como color de texto, tamaño de la fuente, decoración.

Se nombra con el nombre del elemento seguido de dos guiones medios y el nombre del modificador

```
.menu__item--active  
.menu-principal__link--big
```

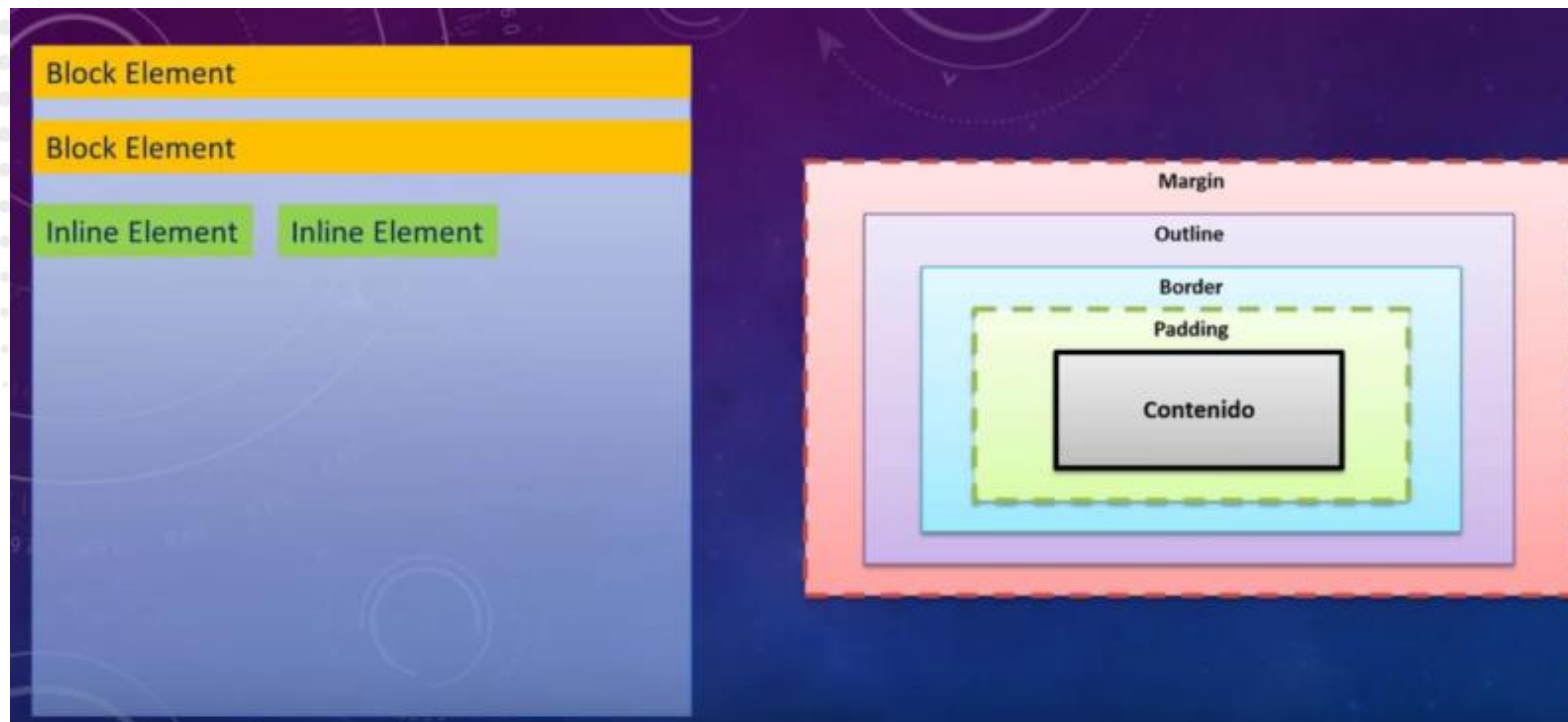
Box-Model

En la web todos los elementos se pueden representar como cajas, una manera de verlo es inspeccionando el código, lo que se resalta al momento de pararse sobre un elemento es un rectángulo, esta forma en la que el navegador dibuja estas cajas se llama el LAYOUT, que hace referencia a la geometría de la web, la posición de un elemento con respecto a las propiedades principales de cada caja es el ancho (width) y el alto (height).



Existen dos tipos de elementos **HTML**:

- Elementos **inline** que solo ocupan su contenido y no se puede modificar su ancho o alto
- Elementos **block** que ocupan todo el ancho disponible y se les puede asignar un ancho y alto. **ViewPort.** por defecto el ancho es todo lo que puedan ocupar y el alto es lo que ocupe su contenido





Margin y Padding

Margin es la separación entre una caja y las cajas adyacentes, mientras que el **padding** es la separación entre una caja y su borde o límite

```
.Caja-1{  
margin: 1em; /* Se aplica a todos los bordes top, right, bottom, left*/  
margin: 10px;  
}
```





El futuro digital
es de todos

MinTIC

GRACIAS

OPERADO POR:

