

CSS

Clase 17 - CSS, Hora de "Tunear" mi web app

DOM (Document Object Model)

El modelo de objeto de documento (DOM) es una interpretación para los documentos HTML. Facilita la representación y acceso a las partes de una página, a fin de modificar, tanto su estructura, estilo y contenido. El DOM ve a la página web como un grupo de nodos estructurados y ordenados, que tienen propiedades y métodos haciendo que sea mucho más sencillo acceder a las partes para programar o estilizar (también está muy relacionado con JavaScript).

Definición:

CSS (Cascading Style Sheets) es un lenguaje web para aplicar formato visual (color, tamaño, separación y ubicación) al HTML. Es así que puedes cambiar por completo el aspecto de cualquier etiqueta HTML.

Es un lenguaje que trabaja en conjunto con html, diseñado para definir los aspectos visuales de los elementos (etiquetas), tales como: color, tamaño, apariencia, etc. Básicamente, trata de compensar las limitaciones de diseño y la complejidad de HTML.

Sintaxis

```
selector {  
    attribute1: value;  
    attribute2: value;  
    ...  
}  
  
h1 {  
    color: red;  
}
```

Selectores

- Element: Nombre de la etiqueta
- Class: Próxima clase
- ID: Próxima clase

Atributos

Atributos: son los modificadores que se pueden agregar a cada estilo (un estilo puede estar compuesto por uno o varios atributos)

Hay muchísimos pero vamos a ver:

- color
 - color del texto
 - en inglés: blue, green, grey...
 - Hexadecimal: Usar color picker de Google u cualquier otro.
 - RGB: Usar color picker de Google u cualquier otro.
- background-color
 - color de fondo
- font-size
 - en px
- width
 - en pixel
- height
 - en pixel
- font-family
 - font-family: family-name|generic-family
 - family-name / generic-family A prioritized list of font family names and/or generic family names
 - ej. font-family: Verdana, Geneva;
- padding
 - margen interior
- margin
 - margen exterior
- border
 - borde
 - border: ancho tipo color;
 - border: 2px solid black;

Reglas sintácticas

- Cada declaración CSS está formada por un juego de pares propiedad: valor;
- No se ve afectado por el espacio en blanco. Las propiedades se pueden escribir de corrido o una debajo de la otra.
- Siempre que la propiedad representa un número, el valor debe indicar en qué unidad se expresa.

Ejemplo básico

```
body {
  background-color: grey;
}
```

Ejemplo 1

The screenshot shows a code editor with two tabs: 'HTML' and 'CSS'. The 'HTML' tab contains the following code:

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <article>
      <h1>My First Heading</h1>
      <p>Lorem ipsum dolor sit amet consectetur adipiscing, elit libero nunc montes rhoncus nullam mauris, luctus ridiculus condimentum tincidunt scelerisque. Pharetra litora eu mattis et duis in, ridiculus turpis dapibus praesent orci feugiat varius, sodales torquent morbi dis habitant. Leo metus blandit eleifend litora montes lectus dapibus cum integer quam facilisis platea potenti purus nulla pellentesque tempor class nostra, odio placerat nullam lacus felis sagittis eu fringilla aenean commodo rutrum tincidunt phasellus hac duis in nisi.</p>
    </article>
  </body>
</html>
```

The 'CSS' tab contains the following code:

```
body {
  background-color: grey;
  font-family: Verdana, Geneva, sans-serif;
  padding: 20px;
}

p {
  color: #FFFFFF;
  line-height: 2;
}

h1 {
  font-weight: 1000;
  color: white;
}
```

To the right of the code editor is a preview window showing the rendered HTML. The title is 'Page Title'. The heading is 'My First Heading' in a large, bold, black font. The paragraph below it is in a smaller, regular black font. The entire page has a grey background.

```
body {
  background-color: grey;
  font-family: Verdana, Geneva, sans-serif;
  padding: 20px;
}

p {
  color: #FFFFFF;
  line-height: 2;
}

h1 {
  font-weight: 1000;
  color: white;
}
```

Padre e hijos

Cuando tienes una etiqueta “dentro” de otra, lo que haces es aplicar el concepto de padres e hijos. Esto habilita a agregar atributos específicos a “hijos”, sin alterar los del “padre”. Un padre puede tener muchos hijos, y todos ellos heredan sus características, pudiendo tener también características particulares. Esto habilita a agregar atributos específicos a “hijos”, sin alterar los del “padre”. Un padre puede tener muchos hijos, y todos ellos heredan sus características, pudiendo tener también características particulares.

Sintaxis

```
// atributos para definir al hijo
padre hijo {
  attribute1: value;
  attribute2: value;
  ...
```

```

}

// atributos para definir al nieto
abuelo padre hijo {
    attribute1: value;
    attribute2: value;
    ...
}

// y así sucesivamente

```

Ejemplo básico

```

body p {
    background-color: grey;
}

```

Ejemplo 2

The screenshot shows a code editor with two panes. The left pane contains an HTML file with the following content:

```

<!DOCTYPE html>
<html>
    <head>
        <title>Page Title</title>
    </head>
    <body>
        <article>
            <h1>Titulo del articulo</h1>
            <p>Párrafo de un artículo.</p>
        </article>
        <p>Párrafo suelto.</p>
    </body>
</html>

```

The right pane shows a browser preview with the following styling applied:

- body {**
- background-color: grey;**
- font-family: Futura, "Trebuchet MS", Arial, sans-serif;**
- padding: 20px;**
- }**
- p {**
- background-color: yellow;**
- color: #FFFFFF;**
- line-height: 2;**
- }**
- article p {**
- background-color: green;**
- }**

The browser preview shows the title "Titulo del articulo" in green and the two paragraphs below it in yellow.

```

body {
    background-color: grey;
    font-family: Futura, "Trebuchet MS", Arial, sans-serif;
    padding: 20px;
}

p {
    background-color: yellow;
    color: #FFFFFF;
    line-height: 2;
}

article p {
    background-color: green;
}

```

```
h1 {  
    font-weight: 1000;  
    color: white;  
}
```

¿Dónde va todo esto? Modos Implementación

- Vinculado: incluido en archivos .css en el head del archivo
- Empotrado: utilizando la etiqueta <style></style>
- Inline: definido dentro de la etiqueta con la propiedad style NO RECOMENDADO!!!!

Forma EXTERNA, vinculado

dentro de la etiqueta , llamas al archivo CSS que necesites (recuerda el uso de rutas relativas y absolutas)

```
<link rel="stylesheet" href="archivo.css" />
```

Forma INTERNA, empotrado

Es recomendable que esté dentro de la etiqueta head. Puede estar en body , pero sería más desprolijo.

```
<style>  
/* comentario de CSS, dentro de esta etiqueta, va el  
código CSS, */  
</style>
```

Otra forma INTERNA, inline

muy poco recomendable, consiste en usar para “parches” específicos, o pruebas. Se hace difícil mantenerlo.

No lleva selectores

```
<h1>Un encabezado sin formato</h1>  
<h2 style="CODIGO CSS">H2 con formato CSS</h2>  
<p>Párrafo sin formatear</p>  
<p style="CODIGO CSS">Párrafo formateado</p>  
<p>Otro párrafo sin formatear</p>
```

Actividad

Creamos 2 páginas noticia1.html y noticia2.html que sean iguales en cuanto a etiquetas pero con contenido distinto. Las noticias deben contener un título, una foto, y 2 párrafos cada una.

- Miramos cómo lucen antes de aplicar CSS.
- Vinculamos nuestro archivo CSS del ejemplo 1.
- Miramos cómo lucen las páginas luego de importar nuestro CSS
- Aplicamos CSS para que el fondo de la noticia 2 sea amarillo.
- Aplicamos CSS para que el título de la noticia 1 tenga fuente Verdana