

# NODE JS

## Clase 5: Atributos - colores - box model



## ... Clase 5



Atributos

Colores

Box Model



Nación  
Servicios



# Introducción a CSS

A grayscale photograph of four people (three men and one woman) sitting around a table in a library, working together. They are looking at a laptop and various papers. The background is filled with bookshelves. The image is dimmed, with the title text overlaid in a bright orange color.

## ... CSS (Hojas de estilo en cascada)

Código que se usa para dar estilo estético y animado a un desarrollo web. Al igual que HTML, CSS (Cascading Style Sheets) u Hojas de estilo en cascada en español, no es realmente un lenguaje de programación.



## ... CSS (Hojas de estilo en cascada)

Es un lenguaje de hojas de estilo, es decir, permite aplicar estilos de manera selectiva a elementos en documentos html.



## ... Anatomía de una regla en CSS

Observar el siguiente código CSS de arriba:

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

## ... Anatomía de una regla en CSS

Al campo “p” se lo conoce como **selector**. Establece a quienes aplica la regla que se está por enunciar, más adelante desarrollaremos con más detalles los criterios de selección. En este caso, el selector aplica a todos los elementos “p”, dentro de nuestro **html**, los párrafos.

## ... Anatomía de una regla en CSS

La línea completa “color: red;”, se conoce como declaración, en ella encontramos la propiedad, “color” y el valor, en este caso “red”.



## ... Sintaxis en CSS

Otros detalles importantes de la sintaxis CSS:

- Cada una de las reglas (aparte del selector) deben estar encapsuladas entre llaves `{ }`.
- Dentro de cada declaración hay que usar los dos puntos `:` para separar la propiedad de su valor.
- Dentro de cada regla se debe usar el punto y coma `;` para separar una declaración de la siguiente.

## ... Sintaxis en CSS

De este modo, para modificar varios valores de propiedad a la vez, solo se necesita escribirlos separados por punto y coma (;), así:

```
p {  
  color: red;  
  width: 500px;  
  border: 1px solid black;  
}
```

# Selectores



## ... Selectores de elemento

Se pueden seleccionar elementos **HTML** según su tipo. Por ejemplo, el selector "p" selecciona todos los párrafos del documento **HTML** y se les puede aplicar estilos específicos.



## ... Seleccionar varios elementos

También se pueden seleccionar varios elementos y aplicar una sola regla a todos ellos. Esto se aplica incluyendo selectores separados por comas (,). Por ejemplo:

```
p,li,h1 {  
  color: red;  
}
```

## ... Selectores de clase

Para seleccionar elementos **HTML** según su atributo de clase.

Se emplea el punto (.) seguido del nombre de la clase. Por ejemplo, el selector **".button"** selecciona todos los elementos que tienen la clase **"button"** y se les puede aplicar un estilo particular.

## ... Selectores de ID

Para seleccionar elementos HTML según su atributo de identificación (ID).

Se definen utilizando el símbolo numeral (#) seguido del nombre del ID.

Por ejemplo, el selector **"#heading"** selecciona el elemento con el ID **"heading"**, para luego aplicar los distintos estilos personalizados.

## ... Selectores de Atributo

Permiten seleccionar elementos HTML según sus atributos, a partir de un atributo específico o cierta condición. Por ejemplo, el selector **"[type='text']"** selecciona todos los elementos de entrada de tipo texto.



## ... Selectores descendientes y combinados:

Permiten seleccionar elementos específicos que son descendientes directos o indirectos de otros elementos. Por ejemplo, el selector "div p" selecciona todos los párrafos que están dentro de un elemento div.

## ... Diferentes tipos de selectores

Existen muchos tipos diferentes de selectores. Antes solo viste los selectores de elementos, los cuales seleccionan todos los elementos de un tipo dado en los documentos **HTML**. Sin embargo es posible hacer selecciones más específicas que esas, pudiendo extender a atributos, clases, pseudoclases, elementos por id y elementos en particular.

Existen muchos más selectores para explorar, y podrás encontrar una lista más detallada en esta guía de Selectores recomendada:

**[https://developer.mozilla.org/es/docs/Learn/CSS/Building\\_blocks/Selectors](https://developer.mozilla.org/es/docs/Learn/CSS/Building_blocks/Selectors)**



## ... Clase 5



Atributos

Colores

Box Model



Nación  
Servicios



# Colores en desarrollo Web

## ... Colores en CSS

Los colores en **CSS** se pueden indicar de cinco formas diferentes:

- Palabras clave
- Colores del sistema
- RGB hexadecimal
- RGB numérico
- RGB porcentual.

Aunque el método más habitual es el del **RGB hexadecimal**, a continuación se detallan todas las alternativas que ofrece **CSS**.

# Palabra clave (key word)

CSS define 17 palabras clave para referirse a los colores básicos. Las palabras se corresponden con el nombre en inglés de cada color:

aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, orange, purple, red, silver, teal, white, yellow

<b>maroon</b> #800000	<b>red</b> #ff0000	<b>orange</b> #ffa500	<b>yellow</b> #ffff00	<b>olive</b> #808000
<b>purple</b> #800080	<b>fuchsia</b> #ff00ff	<b>white</b> #ffffff	<b>lime</b> #00ff00	<b>green</b> #008000
<b>navy</b> #000080	<b>blue</b> #0000ff	<b>aqua</b> #00ffff	<b>teal</b> #008080	
<b>black</b> #000000	<b>silver</b> #c0c0c0	<b>gray</b> #808080		

## ... Palabra clave, limitaciones...

Aunque es una forma muy sencilla de referirse a los colores básicos, este método prácticamente no se utiliza en las hojas de estilos de los sitios web reales, ya que se trata de una gama de colores muy limitada.

## ... Palabra clave, limitaciones...

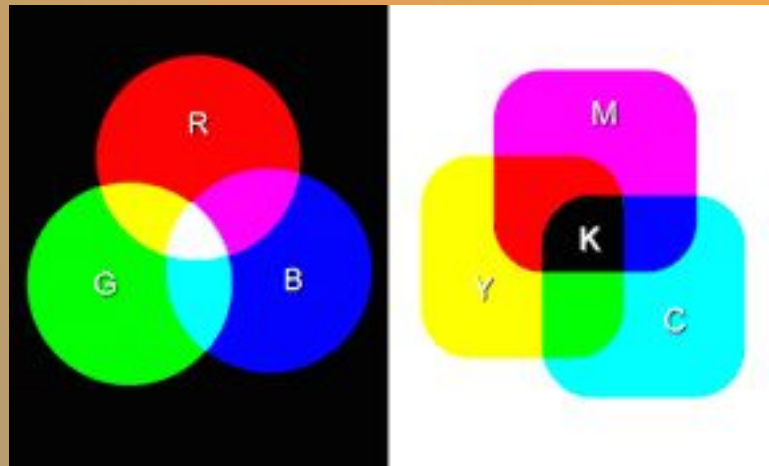
Además de la lista básica, los navegadores modernos soportan muchos otros nombres de colores. La lista completa se puede ver en:

[\*\*wikipedia.org/wiki/Websafe.\*\*](https://wikipedia.org/wiki/Web-safe_colors)



# RGB decimal

En el campo del diseño gráfico, se han definido varios modelos para hacer referencia a los colores. Los dos modelos más conocidos son **RGB** y **CMYK**.



## ... RGB decimal

Simplificando su explicación, el modelo **RGB** consiste en definir un color indicando la cantidad de color rojo, verde y azul que se debe mezclar para obtener ese color. Técnicamente, el modelo **RGB** es un modelo de tipo "**aditivo**", ya que los colores se obtienen sumando sus componentes.

## ... RGB decimal

Si todas las componentes valen 0, el color creado es el negro y si todas las componentes toman su valor máximo, el color obtenido es el blanco. En **CSS**, las componentes de los colores definidos mediante **RGB** decimal pueden tomar valores entre **0** y **255**. El siguiente ejemplo establece el color del texto de un párrafo:

```
p { color: rgb(71, 98, 176); }
```

## ... RGB porcentual

Las componentes **RGB** de un color también se pueden indicar mediante un porcentaje. El funcionamiento y la sintaxis de este método es el mismo que el del **RGB** decimal. La única diferencia es que en este caso el valor de las componentes **RGB** puede tomar valores entre 0% y 100%.

El mismo color del ejemplo anterior se puede representar de forma porcentual: **p { color: rgb(27%, 38%, 69%); }**

## ... RGB hexadecimal

Aunque es el método más complicado para indicar los colores, se trata del método más utilizado con mucha diferencia. De hecho, prácticamente todos los sitios web reales utilizan exclusivamente este método.

## ... RGB hexadecimal

Definir un color en **CSS** con el método **RGB** hexadecimal requiere realizar los siguientes pasos:

- Determinar las componentes **RGB** decimales del color original, por ejemplo: R = 71, G = 98, B = 176
- Transformar el valor decimal de cada componente al sistema numérico hexadecimal. Se trata de una operación exclusivamente matemática, por lo que se puede utilizar una calculadora.

## ... RGB hexadecimal

En el ejemplo anterior, el valor hexadecimal de cada componente es: **R = 47, G = 62, B = B0**. Para obtener el color completo en formato **RGB hexadecimal**, se concatena los valores hexadecimales de las componentes **RGB** en ese orden y se les añade el prefijo **#**. De esta forma, el color del ejemplo anterior es **#4762B0** en formato **RGB hexadecimal**.

## ... RGB hexadecimal

Siguiendo el mismo ejemplo de las secciones anteriores, el color del párrafo se indica de la siguiente forma utilizando el formato **RGB hexadecimal**:

```
p { color: #4762B0; }
```



## ... Excepción en el RGB hexadecimal

El formato **RGB hexadecimal** es la forma más compacta de indicar un color, ya que incluso es posible comprimir sus valores cuando todas sus componentes son iguales dos a dos:

- #AAA = #AAAAAA
- #FFF = #FFFFFF
- #A0F = #AA00FF
- #369 = #336699

## ... Excepción en el RGB hexadecimal

En el siguiente ejemplo se establece el color de fondo de la página a blanco, el color del texto a negro y el color de la letra de los titulares se define de color rojo:

```
body { background-color: #FFF; color: #000; }
```

```
h1, h2, h3, h4, h5, h6 { color: #C00; }
```



## ... Clase 5



Atributos

Colores

Box Model



Nación  
Servicios



A grayscale photograph of four people (two women and two men) sitting at a long table in a library, surrounded by bookshelves. They are looking at papers and a laptop, appearing to be in a collaborative work or study session. The image is dimmed to serve as a background for the text.

# Fuentes, texto, box model

## ... Fuentes en el desarrollo Web

La propiedad **font-family** se refiere a la(s) fuente(s) que se desea usar en el texto. Esta regla define una fuente base global y un tamaño de fuente para usar en toda la página.

Dado que **<html>** es el elemento primario (o padre) de toda la página, todos los elementos contenidos dentro de él heredan las propiedades **font-size y font-family**.

# Fuentes en el desarrollo Web



Nación  
Servicios



```
html {  
  font-size: 10px;  
  font-family: "Open Sans", sans-serif;  
}
```

```
h1 {  
  font-size: 60px;  
  text-align: center;  
}
```

```
p, li {  
  font-size: 16px;  
  line-height: 2;  
  letter-spacing: 1px;  
}
```

# Separación entre componentes, box model

Una cosa que es posible notar sobre la escritura de **CSS** es que se puede tratar a los componentes como si fueran un conjunto de cajas, las cuales van variando su tamaño, color, posición, entre otros parámetros.



## ... Separación entre componentes, box model

A partir de esta lógica de cajas, es que debemos manejar los siguientes parámetros:

- **padding (relleno):** espacio alrededor del contenido. En el ejemplo siguiente, es el espacio alrededor del texto del párrafo.
- **border (marco):** línea que se encuentra fuera del relleno.
- **margin (margen):** espacio fuera del elemento que lo separa de los demás.



## ... Separación entre componentes, box model

En esta sección también se utiliza:

- **width (ancho del elemento)**
- **background-color:** color de fondo del contenido y del relleno
- **color:** el color del contenido del elemento (generalmente texto)
- **text-shadow:** coloca una sombra difuminada en el texto dentro del elemento
- **display:** selecciona el modo de visualización para el elemento (se desarrollará más adelante)

# Cascada y herencia

## ... Cascada y herencia

El nombre de hojas de estilo en “**cascada**” no es casual, expresa que los estilos que especifican con reglas se pueden heredar de una manera jerárquica. Para ejemplificar, veamos esta regla:

```
body {  
    font-family: Arial;  
}
```



## ... Cascada y herencia

Aquí estamos diciendo que la etiqueta **<body>**, que es la que envuelve el contenido de cualquier página web, tenga un tipo de letra “Arial”.

## ... Cascada y herencia

Esto no tendría mucho sentido si no fuera porque gracias a la capacidad de herencia de las reglas de este modo por defecto cualquier elemento hijo como un título o un párrafo va a **“heredar”** ese estilo, salvo que especifique lo contrario como, por ejemplo, en esta regla:

## ... Cascada y herencia

```
p {  
    font-family: Verdana;  
}
```

En esta última regla aplica el principio de “especificidad”. Podríamos pensar que se está planteando un conflicto entre la regla general de `<body>` con lo que dice la regla de `<p>`.

## ... Cascada y herencia

Pero se resuelve fácilmente puesto que se aplica la regla más específica y referirse a un párrafo es más específico que referirse a “los elementos hijos que pueden dentro de `<body>`”.

A grayscale photograph of four people (three men and one woman) sitting around a desk in a library, looking at a laptop and papers. The background is filled with bookshelves. The image is dimmed to serve as a background for the title.

# Vínculo entre HTML y CSS



## ... CSS inline

La primera opción consiste en usar el atributo “style” en un elemento **HTML** tal como se puede ver aquí:

```
<h1 style="font-family:Verdana; color:red"> Título de nivel 1 </h1>
```

## ... CSS inline

En este caso, no hay selector puesto que no hay elemento que seleccionar, la aplicación es directa. Este tipo de reglas tienen la mayor prioridad de todas las reglas.

Esta implementación no se recomienda, salvo en desarrollos ínfimos. Caso contrario, es poco práctica y reduce la legibilidad del código.

## ... CSS interno

La segunda opción es escribir las reglas dentro de un elemento **<style>** dentro de esta etiqueta que, a su vez, se encontraría dentro del elemento **<head>** de la página **HTML**.

## ... CSS interno

Obviamente la desventaja es que en este caso las reglas solo son visibles para esa página concreta en las que se han declarado y no se pueden usar para otras páginas.

## ... CSS externo

La última opción es la más usada y consiste en crear uno o varios ficheros con extensión **".css"** en los cuales se declaran las reglas **CSS**.

La manera de vincular estas reglas a una página **HTML** consiste en hacer referencia el fichero **.css** desde esa página. De este modo se puede crear un fichero global para una web entera y usarlo en todas las páginas que se quiera.

# ¡Muchas gracias!

