

APLICANDO CSS

“Lo bueno necesita aportar pruebas; lo bello, no”

(Bernard Le Bouvier de Fontenelle)

INTRODUCCION A CSS

Las hojas de estilo en cascada (CSS) definen la presentación de un documento HTML. HTML es ante todo un lenguaje para definir la estructura de un documento, pero no define como se verán esos elementos. CSS es el lenguaje que le permitirá agregar estilos a la estructura creada HTML.



Nota

Si alguna vez uso un procesador de texto como Word vera que después de escribir el texto en el documento deberá aplicarle la presentación del mismo como por ejemplo la modificación de la fuente tipográfica o el color de un título. Esto es precisamente lo que realiza la CSS sobre el HTML.

DEFINIR REGLAS DE ESTILO CSS

Cada regla de estilo CSS (también denominada estilo) tiene dos partes principales: un selector (por ejemplo, h1) y una o más declaraciones que aparecen entre una llave de apertura ({) y una llave de cierre (}) (por ejemplo una declaración podría ser, color:#FFCCDD). La declaración consta de una propiedad (color) y su valor (#FFCCDD). La sintaxis de una regla de estilo es la siguiente:

```
selector { propiedad : valor ; propiedad : valor ; ... }
```

Por ejemplo, la siguiente regla de estilo CSS especifica que el texto de todos los elementos h1 deben estar centrados y tener un color de fuente azul.

```
h1 {text-align:center; color: #99CCFF;}
```

El punto y coma después del par propiedad-valor separa los posibles otros pares de propiedad-valor en una misma declaración.



Nota

En esta parte del curso trabajaremos con colores en forma hexadecimal. Una buena opción es trabajar con paletas de colores construidas por expertos diseñadores. Puede visitar el sitio <https://coolors.co>, en dicho sitio vera varios conjunto de colores para pasar a otras paletas puede hacerlo sencillamente con la tecla barra espaciadora.

TIPOS DE SELECTORES

Hay varios tipos de selectores. El selector basado en una etiqueta html esta sólo está compuesto por el nombre del elemento HTML, como por ejemplo h1. Estos estilos se aplican a todas las instancias de un elemento HTML determinado. Un selector basado en una clase está precedido por un punto (.) y define a que subconjunto de elementos se aplican (por ejemplo, a algunos de los elementos p) pero también se pueden aplicar a tipos diferentes de elementos, como a algunos elementos p y a algunos elementos h1. Un selector basado en un identificador está precedido por un numeral (#) seguido de una cadena de caracteres definidos por el desarrollador, pero solo puede ser aplicado a un solo elemento.

Por ultimo encontraremos los selectores descendientes, que se aplican a los elementos que se encuentran dentro de otros elementos. Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento.

Ahora veremos algunos ejemplos básicos para el manejo de selectores CSS y su aplicación dentro de la estructura del HTML.

Ejemplo de selector HTML:

En CSS:

```
ul {  
    list-style: none;  
}
```

Su aplicación en HTML:

```
<ul>  
  <li>Manzanas</li>  
  <li>Peras</li>  
  <li>Bananas</li>  
</ul>
```

Ejemplo de selector basado en un identificador:

En CSS:

```
#container {  
    width: 960px;  
    margin: 0 auto;  
}
```

Su aplicación en HTML:

```
<div id="container"></div>
```

Ejemplo de selector de clase:

```
.red {  
    color: #ff0000;  
}
```

Aplicado en HTML seria lo siguiente:

```
<div class="red"></div>
```

Ejemplo de selector descendiente:

```
#container .box {  
    padding-bottom: 15px;  
}
```

Un ejemplo de aplicación en HTML quedaría de esta forma:

```
<div id="container">  
    <div class="box"></div>  
</div>
```

ESCRIBIR ESTILOS CSS

Existen 3 formas de insertar CSS:

- Una hoja de estilo externo (Archivo con extensión .css).
- Una hoja de estilo interna (Dentro del HTML con el tag <style>).
- Estilo inline (usando el atributo style sobre un elemento HTML).

En general, las reglas que se aplican al sitio web entero se escriben en una hoja de estilos externa. Las reglas de estilo que se aplican sólo a una página se escriben en el elemento style de la página. Las reglas de estilo que se aplican a un único elemento de la página se escriben como estilo inline. Muchos diseñadores y programadores encuentran que escribir reglas de estilo en una o varias hojas de estilos externas facilita el mantenimiento de los estilos.

CREAR ESTILOS INLINE

Una regla del estilo inline se define en la etiqueta de apertura de un elemento mediante el atributo style. Utilice un estilo inline si desea definir las propiedades de un único elemento de una página web y no desea volver a usar ese estilo.

En el ejemplo siguiente se muestra un estilo inline:

```
<p style="font-weight: bold; font-style: italic; color: #FF0000">
```

CREAR ESTILOS CSS INTERNOS (ESPECÍFICOS DE LA PÁGINA)

Las reglas de estilo CSS se pueden definir dentro de la etiqueta head de una página web. En ese caso, las reglas de estilo se aplican sólo a los elementos de esa página.

Ejemplo:

```
<head>
<style >
p{ font-weight: bold; font-style: italic; color: #FF0000}
</style>
</head>
```

HOJAS DE ESTILOS EXTERNAS

Una hoja de estilos externa es un archivo de texto con una extensión de archivo .css que contiene sólo reglas de estilo. Puede vincular una hoja de estilos a una página web utilizando la etiqueta <link> dentro de la etiqueta <head>, como se muestra en el ejemplo siguiente.

```
<link rel="stylesheet" href="style.css">
```

Para este ejemplo necesitamos un archivo de texto denominado style.css que contiene todas las reglas de estilo que necesitaremos en nuestro sitio web. Las reglas de estilo incluidas en una hoja de estilos externa se escriben de la misma manera que en una etiqueta <style>. Sin embargo, no se escriben entre etiquetas <style> y </style>. En el ejemplo siguiente se muestra el contenido completo del archivo style.css.

```
p{ font-weight: bold; font-style: italic; color: #FF0000}
```

Puede vincular una hoja de estilos externa a varias páginas HTML para aplicar los estilos a todo el sitio web. Las hojas de estilos separan las reglas de formato del contenido. De esta manera resulta más fácil administrar las reglas de estilo.

PRIORIDADES DENTRO DE UNA CSS

Las reglas de estilo CSS se aplican "en cascada" siguiendo un orden de prioridad. Las reglas de estilo globales se aplican primero a los elementos HTML y las reglas de estilo locales las invalidan. Por ejemplo, un estilo definido en una etiqueta <style> de una página web invalida un estilo definido en una hoja de estilos externa. De forma similar, un estilo inline definido en una etiqueta HTML de la página invalida cualquier estilo definido dentro de la etiqueta <style> o en una hoja de estilo externa.

COMENTARIOS EN CSS

CSS también nos permite incluir comentarios en nuestros archivos que el navegador ignorará. Un comentario en CSS es contenido dentro de estos símbolos: /* */.

MODIFICAR EL TEXTO CON CSS

El lenguaje CSS ofrece varias propiedades para modificar el estilo del texto, por ejemplo, podremos usar la propiedad `color` y hacer uso de un color en notación hexadecimal (también lo podemos hacer usando un keyword como las palabras inglesas `red`, `orange`, `green` pero lo normal es hacerlo en hexadecimal). La notación hexadecimal viene acompañada de un signo numeral (#) seguido de seis dígitos, el primer par de caracteres representa rojo, el segundo verde, el tercero azul.

Otra propiedad es `font-family` que indica las tipografías a utilizar con el texto. Los valores que pueden tomar son nombres de diferentes tipografías separadas por coma. Si el nombre de la tipografía tiene más de una palabra (por ende, contiene uno o más espacios) la ponemos entre comillas.

La propiedad `font-size` cambia el tamaño en el que la tipografía es presentada. La propiedad `font-weight` define el peso de la tipografía. Los valores más usados son `bold` para poner 'negritas' y `normal` para contrarrestar este efecto en caso de ser heredado.

Otra propiedad importante es `text-align` que nos permite alinear el texto para donde nos guste, izquierda, derecha centrado o justificado.

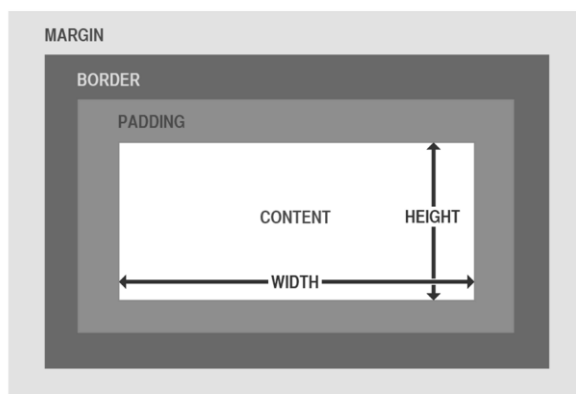
La propiedad `line-height` modifica el alto de línea, es decir la separación entre línea y línea en un párrafo de texto continuo. Este es un valor importantísimo ya que afecta la legibilidad del texto de forma crítica. Una particularidad de `line-height` es que puede ser escrito sin unidad, si elegimos no poner unidad el alto de línea es relativo al tamaño de texto computado independientemente de la unidad de tamaño que hallamos elegido.

Por último, la propiedad `text-decoration` es generalmente usada para manipular el estilo de los hipervínculos, el valor `none` (quita el subrayado) y el valor `underline` (vuelve a colocar el subrayado).

¿QUÉ ES EL MODELO DE CAJA?

El modelo de caja es un conjunto de propiedades que definen como se muestran y se alinean los elementos HTML y cómo interactúan con su entorno en una página web. Este modelo permite dividir una página en una serie de cajas.

Las cajas de las páginas en principio no son visibles a simple vista porque no muestran ningún color de fondo ni ningún borde. Cada caja posee los siguientes elementos:



Contenido (content): Se trata del contenido HTML del elemento (el texto de un párrafo, una imagen, el texto de una lista de elementos, etc.)

Relleno (padding): Es el espacio libre opcional existente entre el contenido y el borde. CSS define cuatro propiedades para definir el relleno:

padding-top (relleno superior)
padding-right (relleno derecho)
padding-bottom (relleno inferior)
padding-left (relleno izquierda)

Borde (border): Línea que encierra completamente el contenido y su relleno. CSS permite modificar el ancho, el color y hasta el estilo que deberán tener los cuatros bordes (superior, derecho, inferior e izquierdo) de nuestra caja, usando las siguientes propiedades.

border-top-width;
border-right-width;
border-bottom-width;
border-left-width;
border-top-color;
border-right-color;
border-bottom-color;
border-left-color;
border-top-style;
border-right-style;
border-bottom-style;
border-left-style;

Margen (margin): Es la separación opcional existente entre la caja y el resto de cajas adyacentes. Hay cuatro propiedades en CSS para controlar cada uno de los márgenes:

margin-top (margen superior)
margin-right (margen derecho)
margin-bottom (margen inferior)
margin-left (margen izquierdo)

NOTACION ABREVIADA

CSS permite crear estilos mediante una sintaxis abreviada conocida como notación CSS abreviada o shorthand. La notación CSS abreviada permite especificar los valores de diversas propiedades con una sola declaración. Por ejemplo, la propiedad border permite definir las propiedades border-width, border-style y border-color en una única línea. Un aspecto esencial que debe tenerse en cuenta cuando se utiliza CSS en forma abreviada es que a los valores omitidos en una propiedad CSS en forma abreviada se les asignan sus valores predeterminados.

Ejemplo:

```
border-width: 1px;  
border-style: solid;  
border-color: #000;
```

Se podría escribir de la siguiente manera:

```
border: 1px solid #000;
```

IMPLEMENTANDO EL BOX MODEL

Un simple ejercicio despejara todas nuestras dudas con respecto al modelo de cajas de CSS:

Para calcular el ancho y el alto de cada caja se usarán las siguientes fórmulas matemáticas:

Total width = left margin + left border + left padding + width + right padding + right border + right margin

Total height = top margin + top border + top padding + height + bottom padding + bottom border + bottom margin

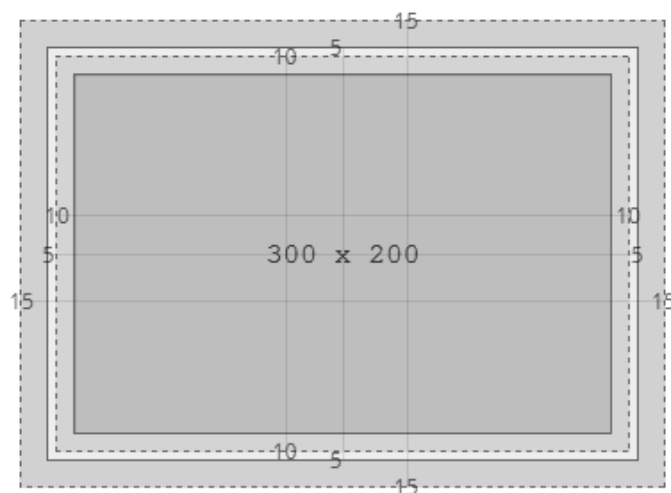
Ahora definimos una pequeña clase en nuestra CSS:

```
.box {  
  width: 300px;  
  height: 200px;  
  padding: 10px;  
  border: 5px solid #000;  
  margin: 15px;  
}
```

El tamaño total del elemento es:

Total width = 15 + 5 + 10 + 300 + 10 + 5 + 15 = 360px

Total height = 15 + 5 + 10 + 200 + 10 + 5 + 15 = 260px

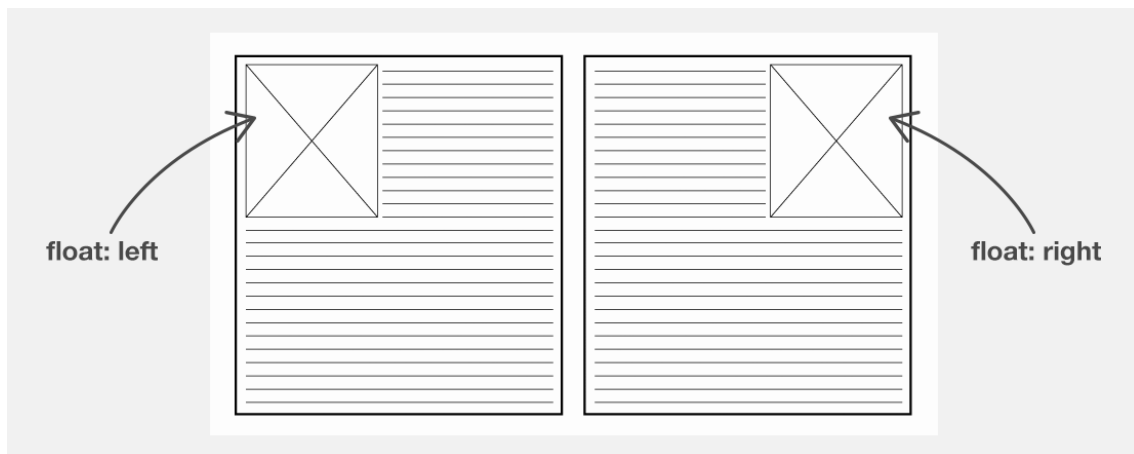


PROPIEDAD FLOAT

A diferencia de html, css permite un buen control sobre el posicionamiento de elementos. Esto nos permitiría por ejemplo posicionar elementos distintos dentro de una página como armar un menú de navegación, o colocar información al lado de un video o imagen. La propiedad float especifica si un elemento debe salir del flujo normal y aparecer a la izquierda o a la derecha de su contenedor. Los posibles valores para float son:

- float: left;
- float: right;
- float: none;

Left significa izquierda y right derecha. None significa que el elemento no flota, por lo que se vuelve a colocar en su posición asignada en el flujo normal. Este último valor es que esta por defecto.



Ahora veremos un ejemplo de posicionamiento en una clase:

```
.columna-izquierda{  
  float: left  
}
```

Al crear un elemento flotante, el navegador sitúa a los elementos que se encuentran a continuación al lado de este mientras haya espacio, aunque no pertenezca al mismo bloque. Para impedir que esto ocurra es necesario que la flotación se interrumpa. La propiedad clear hace que el elemento definido con esta propiedad no tenga elementos flotantes a su lado. Podemos especificar si queremos que no haya elementos flotantes a la izquierda, a la derecha o en ambos lados. El valor típico es both que hace que no haya elementos flotantes ni a su derecha ni a su izquierda.

PROPIEDADES MÁS USADAS EN LAS HOJAS DE ESTILOS

Una vez vista la sintaxis de las hojas de estilo se abordarán las propiedades principales de las hojas de estilos.

a:active: Especifica como mostrar un enlace que se ha hecho un clic con el Mouse.

a:hover: Especifica como mostrar un enlace que tiene el Mouse encima, pero aún no se ha hecho un clic.

a:link: Especifica como mostrar un enlace que no se ha visitado aun.

a:visited: Especifica como mostrar un enlace que ya se ha visitado.

background-color: Permite definir el color de fondo en RGB o en nombre de color.

background-image: Define qué imagen se usará como fondo. Debe colocarse la URL de dicho elemento.

background-position: Controla la posición de las imágenes de fondo. Se puede utilizar porcentajes o medidas o palabras claves como: top, right, bottom, left y center.

background-repeat: Indica cómo se repite la imagen que se usa de fondo. Sus posibles valores son: no-repeat (solo se muestra una sola vez, no se repite), repeat (repite la imagen).

background-size: Indica el tamaño de la imagen de fondo, si el valor es cover se encarga de decirle al navegador que el alto y ancho de la imagen se tienen que escalar de forma automática y proporcional de acuerdo a la ventana de visualización.

border-radius: Permite definir todos los bordes con esquinas redondeadas. El valor de la propiedad define el radio del círculo.

border-top-left-radius: Modifica la esquina superior izquierda del cuadro.

border-top-right-radius: Modifica la esquina superior derecha del cuadro.

border-bottom-right-radius: Modifica la esquina inferior derecha del cuadro.

border-bottom-left-radius: Modifica la esquina inferior izquierda del cuadro.

box-shadow: Establece una sombra difuminada sobre un elemento.

color: Define el color a usar sobre el elemento en formato RGB o el nombre del color.

display: Esta propiedad especifica el tipo de caja de representación que se usara para un elemento. (none, hace invisible un elemento; inline, hace que el elemento genere uno más elementos en la misma línea; block, hace que el elemento genere una caja de bloque).

font-family: Recibe las fuentes que podrán usarse.

font-size: Permite definir el tamaño de la fuente. Si se colocaran en puntos hay que definir la medida Pt.

font-style: Permite definir el estilo de la fuente (normal, normal; italica italic).

font-weight: Permite definir el grosor del texto, por ejemplo en negrita (bold), normal (normal).

height: Establece la altura de un elemento.

letter-spacing: Establece la separación entre las letras del texto. Si se indica una medida, se considera que es el espaciado adicional que se añade a la separación normal de cada letra.

line-height: Establece la distancia en líneas (Interlineado). Se coloca según la unidad (Por ejemplo 1.2px).

list-style-image: Permite indicar una imagen para la marca de cada elemento de la lista. Se le debe asignar la URL de la imagen.

list-style-type: Permite indicar que tipo de marca se desea colocar a cada elemento. none (no tiene marca), disc (la opción por defecto), circle (circulo sin rellenar), square (cuadrado), decimal (la marca es un numero decimal)

margin-bottom: Permite indicar el tamaño del margen inferior. Si se colocan en cm hay que definir la unidad cm.

margin-left: Permite indicar el tamaño del margen izquierdo. Si se colocan en cm hay que definir la unidad cm.

margin-right: Permite indicar el tamaño del margen derecho. Si se colocan en cm hay que definir la unidad cm.

margin-top: Permite indicar el tamaño del margen superior. Si se colocan en cm hay que definir la unidad cm.

overflow: Nos permite definir qué es lo que pasará cuando un objeto contenido se desborda de su contenedor. (visible, esta opción es por defecto y nos permite que el objeto contenido se desborde siempre; hidden, es lo contrario, no permite el desbordamiento; scroll, no permite el desbordamiento pero crea un scroll para poder ver la parte oculta).

padding: Es el relleno es el espacio dentro de un elemento, este espacio está entre el borde del elemento y el contenido del elemento.

text-align: Define la alineación del texto izquierdo (left), derecho (right), centrado (center), justificado (justify).

text-decoration: Permite indicar varios efectos visuales para un texto. Sus posibles valores son: (none) sin nada, (underline) subrayado, con una línea por encima (overline), parpadeando (blink).

text-shadow: Nos permite definir una sombra a un texto.

width: Establece el ancho del elemento.