

Introducción a CSS

Luis Valencia Cabrera (lvalencia@us.es),
Víctor Ramos González (vramos1@us.es)

Departamento de Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

22-02-2023, Bases de Datos

Introducción

En el código HTML se pueden incluir una serie de etiquetas representando tanto la clase de elementos y secciones que queremos incluir en nuestra página como el estilo a aplicar en las mismas.

```
<h1>  
  <font color="blue">Este es un ejemplo</font>  
</h1>
```

Sin embargo, hace ya bastantes años que se desaconseja este tipo de prácticas, y se va tendiendo más a separar los contenidos de la presentación, el marcado de los *contenidos* de los *estilos* de visualización de los mismos.

Introducción

- Al crear una página web se utiliza en **primer lugar** el lenguaje HTML para marcar los contenidos, es decir, designar la función de cada elemento de la página: sección, párrafo, texto destacado, tabla, lista ordenada, control, etc. La idea es que esta parte tenga cada vez más semántica, dotando de sentido a los documentos generados y sus secciones.
- En **segundo lugar**, se emplea CSS para definir el aspecto de cada elemento (color, tamaño, tipo de letra, posición, etc.).
- CSS son las siglas de Cascade Style Sheet

Introducción

- La especificación o norma oficial que se utilizaba hace algún tiempo para establecer estilos con CSS era la versión CSS 2.1, que se puede consultar en <https://www.w3.org/TR/CSS21> (traducida en <http://www.sidar.org/recur/desdi/traduc/es/css/cover.html>)
- Sin embargo, hace años la W3C trabaja en la elaboración de lo que se conocía como CSS3, cuyo estándar ha quedado dividido en muchas secciones, que se pueden consultar [en este enlace](#). Esta nueva versión incluye multitud de cambios importantes a todos los niveles, siendo mucho más avanzada y compleja que CSS2.

Introducción

Para empezar, podemos definir una página web sencilla como

```
<!DOCTYPE html>
<html>
<head>
<title>Página básica sin estilo</title>
</head>

<body>
<h1> Este es un ejemplo</h1>
<p>Este es el primer párrafo</p>
<p>Este es el segundo párrafo</p>
</body>
</html>
```

Introducción

- Si queremos que el título aparezca en azul, podemos añadir el elemento ``, y dentro del mismo asignar el valor `blue` al atributo `color`.
- Si queremos destacar el segundo párrafo (porque nos parezca importante) aumentando el tamaño de letra podemos añadir el elemento `` y dentro de ese elemento asignar el valor `5` al atributo `size`.

Introducción

Podemos definir una página web sencilla como

```
<!DOCTYPE html>
<html>
<head>
<title>Página básica sin estilo</title>
</head>

<body>
<h1> <font color="blue">
Este es un ejemplo</font></h1>
<p>Este es el primer párrafo</p>
<p><font size="5">
Este es el segundo párrafo</font></p>
</body>
</html>
```

Introducción

- Hemos logrado el cambio de visualización deseado, pero a costa de mezclar los contenidos con el formato de los mismos.
- Sin embargo, una cuestión concierne a la persona de la empresa que decide el tipo de información a presentar, y otra a la persona encargada de definir la presentación/maquetación de la información, que no tendrían por qué ser la misma.
- Además, si toda cabecera principal se debe poner en azul, deberemos indicar la misma instrucción en todas las cabeceras, lo cual es redundante y puede implicar incoherencias de estilo.

Introducción

Una alternativa es separar los contenidos del formato

```
<!DOCTYPE html>
<html>
<head>
<title>Página básica con estilo</title>
<style type="text/css">
  h1 {color:blue;}
  .imp {font-size: x-large;}
</style>
</head>

<body>
<h1> Este es un ejemplo</h1>
<p>Este es el primer párrafo</p>
<p class="imp"> Este es el segundo párrafo</p>
</body>
</html>
```

Introducción

- Como vemos, se han eliminado las referencias a color o tamaño de texto como atributos de las etiquetas `<h1>` y `<p>`.
- En su lugar, se ha dejado sin atributo la etiqueta `<h1>` y se ha calificado uno de los párrafos como importante (`class="imp"`), lo cual tiene una consideración más semántica.

Introducción

- Será la hoja de estilo la que nos diga cómo presentar las cabeceras, los párrafos normales (por defecto) y los párrafos importantes (calificados).
- A simple vista podemos ver que se establece dentro de `<head>` un `<style>` de tipo `text/css`, y que en su interior definimos el formato a aplicar a las etiquetas `<h1>` (`color` de texto `blue`), y a aquello que se califique como importante, `.imp` (tamaño de texto `x-large`).

Incluir código CSS en nuestra página HTML

Hay tres formas de incluir el código de estilo en nuestra página HTML

Primera forma: incluir estilo como atributo de etiqueta.

- Se pueden emplear instrucciones de estilo CSS con el atributo `style` dentro de cualquier etiqueta HTML. Por ejemplo:

```
<p style="font-size: x-large">
```

- Es la opción menos aconsejada y usada, al no separar bien contenido y presentación.
 - Dificulta la uniformidad y estandarización del sitio.
 - Opción a emplear para aplicar un formato de forma excepcional a un solo elemento.

Incluir código CSS en nuestra página HTML

Primera forma

```
<!DOCTYPE html>
<html>
<head>
<title>Página básica sin estilo</title>
</head>

<body>
<h1 style="color: blue"> Este es un ejemplo</h1>
<p>Este es el primer párrafo</p>
<p style="font-size: x-large">
Este es el segundo párrafo</p>
</body>
</html>
```

Definición de estilos CSS

- Los estilos aplicados directamente sobre un elemento HTML incluyen únicamente la sección de declaraciones, ya que el elemento seleccionado es sobre el que estamos actuando con el atributo style.
- Hasta la especificación CSS 2.1 había hace años poco más de 100 propiedades definidas, mientras que en CSS3 hace tiempo que se excedieron con creces las 200, aunque como adelantamos esta especificación se mantiene viva.

Incluir código CSS en nuestra página HTML

Segunda forma

Segunda forma: Incluir estilo en el cuerpo de la etiqueta STYLE.

- Se pueden emplear instrucciones de estilo CSS empleando una etiqueta `<STYLE>` en la sección HEAD de nuestro documento HTML.
- Por ejemplo, como vimos anteriormente.
- Permite aplicar el estilo definido a todos los elementos de un determinado tipo de contenido.

Incluir código CSS en nuestra página HTML

Tercera forma

Tercera forma: incluir el estilo en un fichero .css independiente.

- Podemos definir un documento CSS con el estilo, y utilizarlo así en todas las páginas html de manera uniforme.
- Es la opción *más aconsejada*, por ser la que permite una **mejor separación** entre **contenido** y **presentación**.

Incluir código CSS en nuestra página HTML

Tercera forma

- El código en el fichero aparte sería del mismo tipo de la opción anterior, pero sin la etiqueta `style`. Por ejemplo, podemos tener un archivo `miestilo.css`.

```
h1 {color:blue;}  
.imp {font-size: x-large;}
```

- Para importarlo, bastará con escribir en la cabecera (head) de las páginas que queramos aplicar el estilo:

```
<link rel="stylesheet" href="miestilo.css" type="text/css">
```

Incluir código CSS en nuestra página HTML

Tercera forma

- De esta forma, cuando el navegador carga la página HTML, al ver esta etiqueta descarga los archivos CSS externos enlazados mediante `<link>` y aplica los estilos a los contenidos de la página.
- También se pueden emplear rutas relativas a otras carpetas, o rutas absolutas dentro de un servidor accesible online.
- Además, css permite una instrucción `@import` para importar archivos de estilos.

Incluir código CSS en nuestra página HTML

```
<!DOCTYPE html>
<head>
<title>Ejemplo de estilos CSS en un
archivo externo</title>
<style type="text/css">
  @import "css/estilos.css";
</style>
</head>

<body>
<p>Un párrafo de texto.</p>
</body>
</html>
```

Incluir código CSS en nuestra página HTML

- En el caso anterior, el fichero `estilos.css` lo hemos guardado en el directorio `css`. En este caso el contenido del fichero es una única línea

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

- Es la opción recomendada para separar los ficheros HTML de los ficheros de estilo.

Definición de estilos CSS

La etiqueta `<link>` incluye cuatro atributos para enlazar CSS:

- **rel**: indica el tipo de relación que existe entre el recurso enlazado (en este caso, el archivo CSS) y la página HTML. Para los archivos CSS, siempre se utiliza el valor `stylesheet`.
- **type**: indica el tipo de recurso enlazado. Sus valores están estandarizados y para los archivos CSS su valor siempre es `text/css`.
- **href**: indica la URL del archivo CSS que contiene los estilos. La URL indicada puede ser relativa o absoluta y puede apuntar a un recurso interno o externo al sitio web.
- **media**: indica el medio (impresora, pantalla) en el que se van a aplicar los estilos del archivo CSS. Si no se especifica, toma el valor `all`.

Los comentarios de HTML y de CSS son diferentes y no deben confundirse

- En HTML

```
<!-- Este es un comentario HTML -->
```

- En CSS

```
/* Este es un comentario  
de CSS */
```

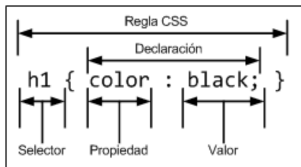
Definición de estilos CSS

- Una sección de definición de estilos CSS (tanto en etiqueta style como en archivo externo) permite definir un conjunto de reglas CSS.
- Las reglas CSS se componen de un **selector** que indica *a qué elementos HTML se aplicará la regla* y un conjunto de **declaraciones**, indicando los *estilos a aplicar* a los elementos seleccionados.

Definición de estilos CSS

- Una declaración se compone de una o varias propiedades CSS con sus valores, siendo:
 - Propiedad: característica a modificar.
 - Valor: establece el valor a aplicar a la característica en cuestión.

```
selector1 { Propiedad : Valor;  
    ...  
    Propiedad : Valor; }
```



Definición de estilos

- En los selectores, la declaración indica *qué hay que hacer* y el selector indica *a qué elementos hay que hacerlo*.
- A un mismo elemento HTML se le pueden aplicar varias reglas CSS y cada regla CSS puede aplicarse a un número ilimitado de elementos. En otras palabras, una misma regla puede aplicarse sobre varios selectores y un mismo selector se puede utilizar en varias reglas.
- El estándar de CSS define muchos tipos de selectores, que permiten seleccionar de forma muy precisa elementos individuales o conjuntos de elementos dentro de una web.
- No obstante, la mayoría de páginas de los sitios web se pueden diseñar utilizando solamente los selectores básicos.

Selectores básicos

Selector universal

- Se utiliza para seleccionar todos los elementos de la página. El siguiente ejemplo elimina el margen y el relleno de todos los elementos HTML:

```
* {margin: 0; padding: 0;}
```

- El selector universal se indica mediante un asterisco (*). A pesar de su sencillez, no se utiliza habitualmente, ya que es difícil que un mismo estilo se pueda aplicar a todos los elementos de una página.
- **padding** se usa para rellenar espacio alrededor de un contenido especificado; **margin** hace referencia al margen.

Selector de tipo o etiqueta

- Selecciona todos los elementos de la página cuya etiqueta HTML coincide con el valor del selector. Para utilizar este selector, solamente es necesario indicar el nombre de una etiqueta HTML (sin los caracteres `<` y `>`) correspondiente a los elementos que se quieren seleccionar.

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

```
h2 {color: blue;}
```

Selectores

Si se quiere aplicar los mismos estilos a dos etiquetas diferentes, se pueden encadenar los selectores. En el siguiente ejemplo, los títulos de sección h1, h2 y h3 comparten los mismos estilos:

```
h1, h2, h3 {  
    color: #8A8E27;  
    font-weight: normal;  
}
```

#8A8E27 es la expresión de un color RGB en expresión hexadecimal.

Selectores

En las hojas de estilo complejas, es habitual agrupar las propiedades comunes de varios elementos en una única regla CSS y posteriormente definir las propiedades específicas de esos mismos elementos.

```
h1, h2, h3 {  
    color: #8A8E27;  
    font-weight: normal;  
}  
  
h1 { font-size: 2em; }  
h2 { font-size: 1.5em; }  
h3 { font-size: 1.2em; }
```

Selectores

Selector descendente

- Selecciona 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.
- El selector del siguiente ejemplo selecciona todos los elementos `` de la página que se encuentren dentro de un elemento `<p>` (`` se usa para seleccionar un trozo de texto):

```
p span { color: red; }
```
- Al resto de elementos `` de la página que no están dentro de un elemento `<p>`, no se les aplica la regla CSS anterior.

Selectores

Selector hijo

- Selecciona los elementos que se encuentran **inmediatamente** dentro de otros elementos. Un elemento es hijo de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento, directamente dentro de él, no dentro de ninguno de sus descendientes.
- El selector del siguiente ejemplo selecciona todos los elementos `<p>` de la página que se encuentren dentro del elemento `<body>`:

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

- Al resto de elementos `<p>` de la página que están dentro de otros elementos como `<section>`, `<div>`, etc. , no se les aplica la regla CSS anterior.

Selector de clase

- Se usa el atributo `class` para aplicar un estilo a algunos elementos y no a otros. Por ejemplo

```
<body>  
  <p class="destacado">Primer párrafo</p>  
  <p>Segundo párrafo</p>  
</body>
```

- A continuación, se crea en el archivo CSS una nueva regla llamada `destacado` con todos los estilos que se van a aplicar al elemento (ojo al punto):

```
.destacado { color: red; }
```

- El selector `.destacado` se aplica a cualquier elemento cuyo atributo `class` sea igual a `destacado`.

Selectores de ID

- El selector de ID permite seleccionar un elemento a través del valor de su atributo `id`. Este tipo de selectores sólo seleccionan un elemento de cada página, ya que el valor del atributo `id` no se puede repetir en dos elementos de la misma página.
- La sintaxis de los selectores de ID es muy parecida a la de los selectores de clase, salvo que se utiliza el símbolo de la almohadilla (`#`) en vez del punto (`.`):

```
#destacado { color: red; }
```

```
<p>Primer párrafo</p>
```

```
<p id="destacado">Segundo párrafo</p>
```

```
<p>Tercer párrafo</p>
```

Para saber más

- <https://www.w3.org/MarkUp/Guide/Style>
- <http://www.tutorialesprogramacionya.com/cssya/>
- <http://www.w3schools.com/css/>
- <http://www.sidar.org/recur/desdi/traduc/es/css/cover.html>
- ...

Nota: estos son solamente algunos recursos, teniendo en cuenta el ritmo al que avanza el desarrollo y diseño web es necesario irse actualizando día a día. Intentaremos recopilar información de utilidad a lo largo del cuatrimestre en el sitio web de la asignatura.