
UF3. Gestión de Estilos con CSS

Introducción a CSS *Cascading Style Sheets*

Índice

0. ¿Qué es CSS?
1. **Sintaxis básica**
2. Formas de añadir CSS al HTML
3. Inicialización de
4. Tipos de **Selectores**
 - ❑ Agrupaciones de selectores
 - ❑ Algunas **Pseudo-clases y pseudo-elementos**

Anexos: Algunas **Propiedades** para empezar

- ❑ Propiedades de Fuentes
- ❑ Propiedades de Textos
- ❑ Propiedades de Colores y Fondos
- ❑ Propiedades de Bordes
- ❑ Propiedades de Márgenes

0. ¿Qué es CSS?

- CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos.
- CSS es la mejor forma de separar los contenidos y su presentación.
- CSS se utiliza para definir el aspecto de todos los contenidos, es decir, el color, tamaño de letra, la separación entre párrafos, la tabulación de una lista, etc.

0. ¿Qué es CSS?

■ CSS nos permite:

- ❑ Separar presentación de contenido
- ❑ Mayor flexibilidad, escalabilidad, y posibilidades en la presentación de contenidos web
- ❑ Dinamismo (junto a *JavaScript* y *DOM*)
- ❑ Correctamente utilizado reduce el peso de las páginas

1. Sintaxis Básica

Declaración



Selector { propiedad: valor; ... }

Ejemplo:

```
h1 {    color: #CC9900;
        background-color: red;
    }
```

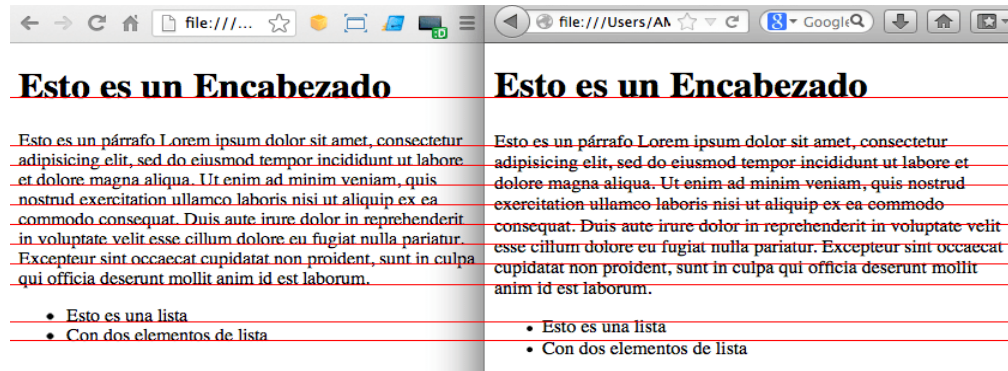
2. Formas de añadir CSS al HTML

- En un documento externo, el cual vinculo al html con `<link ...>`
`<link rel="stylesheet" type="text/css" href="estilo.css"/>`
- Dentro de la sección `<head>` del documento html
`<style type="text/css">`
 `p { text-align:right; }`
`</style>`
- Directamente al elemento que le afecta (en línea): **atributo style**
`<p style="text-align: right;">Estilo propio </p>`

Ideas clave...

- Los atributos en línea tienen más “peso” que los otros (hay que evitar utilizarlos)
- Si hay dos reglas dentro de un documento o header, siempre la última regla de estilo es la que se aplica sobre el HTML (por norma general, y ante selectores iguales)
- Es bastante usual utilizar documentos externos en todo el proyecto.
- Un documento HTML se puede vincular a más de una hoja CSS externa.

3. Inicialización de estilos del navegador



- **PROBLEMA: Diferentes navegadores => diferentes estilos por defecto**
- **SOLUCIÓN: Inicializar los valores de estilo a 0**
 - * `{ padding: 0px; margin: 0px; list-style: none; text-decoration: none; border: none; outline: none; }`
- **Hay hojas de estilo CSS ya creadas para realizar esto. Las + conocidas:**
 - ❑ **reset.css** de Eric Meyer <http://meyerweb.com/eric/tools/css/reset/>
 - ❑ **normalize.css** de Nicolas Gallagher <http://necolas.github.io/normalize.css/>
- + Información: <https://octuweb.com/normalize-vs-reset-css/>
<https://franciscoamk.com/usar-reset-css-o-normalizar/>

4. Selectores

4.1. Selectores base

- Selector Universal

- * {font-family: Verdana;}

- Selectores de Elementos html

- h1** {font-family: Arial, Helvetica;
font-size: 10pt; font-style: italic;}

- Selectores de clase

- .enfaticada {font-weight: bold;}

- Selectores de clase, más específicos que sólo afectan al elemento html que especifiquemos

- p.enfaticada {font-weight: bold;}

- Selectores de ID

- #enfaticada** {font-weight: bold;}

4.1. Selectores base

Ejemplo selector de clase:

.pregunta {font-weight: bold; font-style: italic}

Ahora, a los párrafos que queramos agregar estos estilos les agregamos el atributo CLASS dentro de la etiqueta HTML:

```
<p class="pregunta">Esta es una pregunta</p>
```

Nota: Varios elementos pueden pertenecer a una misma clase.
Un elemento puede pertenecer a más de una clase.

```
<p class="pregunta resaltar bordear">Esto es otra pregunta</p>
```

Ejemplo selector de ID:

#especial{color: #FFFF00; background-color: #000000}

Ahora, el párrafo que queramos que tenga este estilo concreto le agregamos el atributo ID dentro de la etiqueta HTML:


```
<p id="especial">Este es un parrafo especial</p>
```

Nota: Los id deben ser únicos para cada elemento
(no debería haber dos elementos con el mismo id, aunque los navegadores “lo toleren”)

4.2. Esquema de trabajo

ESQUEMA DE TRABAJO PARA REALIZAR UNA HOJA DE ESTILOS CSS

Siempre ir de lo más general a lo más concreto

- 
1. Selector universal, para los estilos que más afecten a la página HTML, p.e. fuente más usada, color fondo, alineación. Y también para hacer settings iniciales:

```
* { margin: 0; padding: 0;
  font-family: Arial;
  background-color: #A02001;
  text-align: center}
```
 2. Selectores de elemento, para especificar estilos comunes a los elementos base de HTML

```
li { font-family: Courier;
     font-size: 14px;
     text-align: left}
```
 3. Selectores de clase, para agrupar elementos diferentes que comparten estilos

```
.verde{ color: green; }
```
 4. Selectores de ID, para elementos muy concretos, que tengamos que aplicar estilo especial

```
#especial{color: #FFFF00; background-color: #000000}
```

Nota: Se desaconseja utilizar estilos en línea.

Con el tiempo, mantener páginas con estilos en línea se complica.

4.3. Prioridad de las selectores

- Los estilos “en línea” siempre mandan sobre cualquier otro:

```
<div class="miclase" id="midiv" style="color: purple">  
</div>
```

Nota: se desaconseja su uso!!

- En ausencia de estilos en línea, la prioridad viene dada por...
“selectores más específicos sustituirán a los más generales”

```
* { color: red; }  
div { color: green; }  
.miclase { color: brown; }  
div.miclase { color: blue; }  
#midiv { color: yellow; }
```



- En igualdad de especificidad, el último sobreescribe

```
#midiv { color: yellow; } // en hoja externa  
#midiv { color: pink; } // en la cabecera del doc
```

- La declaración **!important** sirve para deshacer estas reglas

```
#midiv { color: yellow !important; } // hoja externa  
#midiv { color: pink; } // en la cabecera del doc
```

4.4. Agrupación de Selectores

■ Agrupación de selectores

`h1, b, .negrita {font-color: blue; font-style: bold;}`

La coma es una forma abreviada de aplicar estilos a varios elementos

■ Selectores vinculados a otros selectores

- *Elementos P que sean de una clase determinada*

`p.miclase {font-size: 20px;}`

- *Elementos P que tengan un ID determinado*

`p#parrafo1 {font-size: 20px;}`

- *Elementos P que sean hijos de otro elemento*

`li > a {font-size: 20px;}`

- *Elementos que sean descendientes de otro elemento:*

`article a {font-size: 20px;}`

- *Elementos que no cumplan el selector especificado*

`p:not(.micl1) {font-size: 20px;}` (:not acepta selectores simples)

- *Elementos adyacentes: `div+p` (selecciona el p al lado de div)*

- *Elementos hermanos: `div~p` (selecciona todos los p al lado de div)*

4.5. Selectores por atributo

- En **CSS** se permite referenciar atributos

- *Elementos que tengan un atributo dado:*

```
p[name] {font-size: 20px;}
```

- *Elementos que el atributo dado con tenga el valor especificado:*

```
p[name="mitexto"] {font-size: 20px;}
```

- *Elts que tengan el atributo con esta cadena dentro de su lista de valores:*

```
p[class~="enlace1"] {font-size: 20px;}
```

Otros menos usados:

- *Elementos que tengan el atributo que empiece por la cadena:*

```
p[name^="mi"] {font-size: 20px;}
```

- *Elementos que tengan el atributo que termine por la cadena:*

```
p[name$="texto"] {font-size: 20px;}
```

- *Elementos que tengan el atributo que contenga la cadena:*

```
p[name*="text"] {font-size: 20px;}
```

Vamos a hacer pruebas: <http://www.w3schools.com/cssref/trysel.asp>

4.6. pseudo-clases y pseudo-elementos

■ Pseudo-Elementos (en css3 se expresa con puntos)

p::first-line **p::first-letter** Se aplica primera línea/letra de elemento

p::selection Selecciona texto seleccionado por el usuario.

p::before **p::after** Selecciona parte anterior/posterior a contenido de elemento para insertar nuevo contenido generado. Ejemplo: `p::after { content: " - by ESS"; }`

■ Pseudo-Clases

:not(selector simple) NEGACIÓN, se aplica a todo, menos a lo especificado **+info**

p:first-child **p:last-child** Se aplica al primer/ultimo **p** de su padre

body :nth-child(n) Se aplica al n-esimo hijo de body (acepta **even odd**)

body p:nth-of-type(n) Se aplica al n-esimo **p** de body (acepta **even odd**)

:root Selecciona el elemento raíz (en html5 es el elemento HTML)

■ Pseudo-Estados

a:link Enlace no visitado **a:visited** Enlace visitado

p:hover Puntero sobre elemento **p:active** Se está clicando sobre elto

input:focus Elemento toma el foco

:enabled **:disabled** **:checked** Selecciona elementos de formulario por estado

Extra: sitios web interesantes

- http://www.w3schools.com/cssref/css_selectors.asp
- <http://www.w3schools.com/cssref/trysel.asp>
- http://librosweb.es/css_avanzado/capitulo_3/selectores_de_css_3.html
- <http://css-tricks.com/almanac/selectors/n/not/>
- <http://css-tricks.com/useful-nth-child-recipes/>
- <http://www.google.com/fonts>
- <http://geeksroom.com/2013/05/como-utilizar-google-web-fonts-en-tu-sitio/74508/>
- <http://www.dafont.com/es/>
- <http://www.fontsquirrel.com/>

Anexo: Propiedades para Fuentes

Con estas propiedades controlamos el estilo de la fuente: familia, tamaño, grosor, etc.

- **font-family : nombre de las familias de las fuentes | familia genérica**

Sus posibles valores son los nombres de las fuentes, pudiendo especificar un conjunto. Se selecciona la primera que se encuentra.

- **font-size : XX *unidad* | % | larger | smaller | x-small | medium | large | x-large**

Especifica el tamaño de la fuente. Puede ponerse como una longitud en unidades (absoluta o relativa), aunque también admite palabras reservadas.

☞ ***unidad*** puede ser: **px, in, cm, mm, pt, em** (anchura letra m), **ex** (altura x), **rem**

- **font-weight : lighter | normal | bold | bolder | 100 | 200 | 300 | | 900**

Especifica el peso, densidad o grosor de la fuente, con valores entre 100 y 900 siendo cada vez más densos. El valor 400 equivale a normal, y el 700 a bold.

- **font-style : normal | italic | oblique**

Especifica el estilo de la fuente: normal, en itálica (cursiva) o en oblicua (un poco más inclinada que la itálica).

☞ ***¿Cómo cargar nuestra propia fuente con CSS3?***

```
@font-face {  
    font-family: 'MiPropiaFuente';  
    src: url('fuentes/mifuentes.ttf') format("truetype");  
    src: url('fuentes/mifuentes.otf') format("opentype");  
}
```

☞ ***Otra opción: utilizar google fonts :*** <http://www.google.com/fonts/>

Anexo: Propiedades para Texto

Son propiedades que afectan a la presentación visual de caracteres, espacios, palabras y párrafos.

- **text-align : left | right | center | justify**

Fija la **alineación del bloque**, al margen izquierdo, al derecho, centrado o a ambos. **Se hereda.**

- **text-decoration : underline | overline | line-through | blink | none**

Fija una o más **características "decorativas" del texto**, como subrayado, líneas superiores, caracteres tachados o parpadeantes.

- **vertical-align : baseline | sub | super | top | text-top | middle | bottom | text-bottom**

Alineación vertical del texto en relación con la línea base del texto. **No se hereda.**

- **text-transform : uppercase | lowercase | capitalize | none**

Especifica, si las **letras del texto** deben transformarse en mayúsculas, minúsculas ,,

- **text-indent : XX *unidad* | %**

Fija la **sangría o indentación** de la primera línea del texto. **Se hereda.**

- **letter-spacing : normal | XX *unidad***

Especifica el **espaciado entre letras**. En el caso de especificar una longitud esta se sumara a la normal.

- **word-spacing : normal | XX *unidad***

Especifican el **espaciado entre palabras**. En el caso de especificar una longitud esta se sumara a la normal. **Se hereda.**

- **text-shadow : rgb(0, 0, 150) 3px 3px 5px; //color desp hor, desp vert, radio difum**

- **text-stroke: *grosor color*** Dibuja el contorno de las letras. **Todavía no soportado.**

De momento usar: **-webkit-text-stroke-width: 1px; -webkit-text-stroke-color: black;**

(Con text-shadow también se puede simular)

Anexo: Colores y Fondos

Son las propiedades que permiten fijar el color del primer plano y fondo de un elemento.

- **color** : *NombreColor* | #ValorHexa | rgb(R%,G%,B%) | rgb(R,G,B) | rgba(R,G,B,A)

Especificamos el color del primer plano del contenido del texto. Se hereda.

- **background-color** : *NombreColor* | #ValorHexa | rgb(R%,G%,B%) | rgb(R, G, B) | rgba(R,G,B,Alfa) | transparent

Especifica el color del fondo del elemento, que puede ser un color o la palabra reservada **transparent** (el fondo del elemento padre se ve). Se hereda.

☞ *NombreColor* es un nombre de color conocido: **black, blue, red, gray, green, yellow** ...

+ nombres reconocidos: https://www.w3schools.com/cssref/css_colors.asp

☞ Otras formas de especificar color:

#FF0000 - Será el color rojo **#00FF00** - Será el color verde **#0000FF** - Será el color azul

rgb(0%,100%,0%) - color verde

rgb(0,255,0) - color verde

rgba(0,255,0, 0.5) – añade canal Alfa, con el que podemos especificar la OPACIDAD del color (1 es totalmente opaco, 0 es transparente)

Anexo: Colores y Fondos

- **background-image : none | url(dirección)**

Designa una imagen para rellenar el fondo del elemento por medio de una URL o la palabra reservada none para indicar que no se utilizara ninguna imagen.

- **background-repeat : repeat | repeat-x | repeat-y | no-repeat**

Establece si la imagen de fondo **se repetirá** horizontal y verticalmente para rellenar todo el fondo del elemento (**repeat**), sólo horizontalmente o sólo verticalmente (**repeat-x** o **repeat-y**).

- **background-position : x y | x% y% | left top | left center | left bottom | right top |... | center top"...**

Sets the starting position of a background image. By default, is placed at the top-left corner.

- **background-size: auto| 40% 40%| 50px 50px | cover | contain;**

Sets the size of the image. With cover and contain, expand the image to full size of container

- **background-attachment: scroll | fixed | local;**

Sets whether a background image scrolls with the rest of the page, or is fixed.

https://www.w3schools.com/cssref/playit.asp?filename=playcss_background-size

Background con varias imágenes

background-image: {imagen1} , {imagen2}; //imagen 2 se superpone a imagen

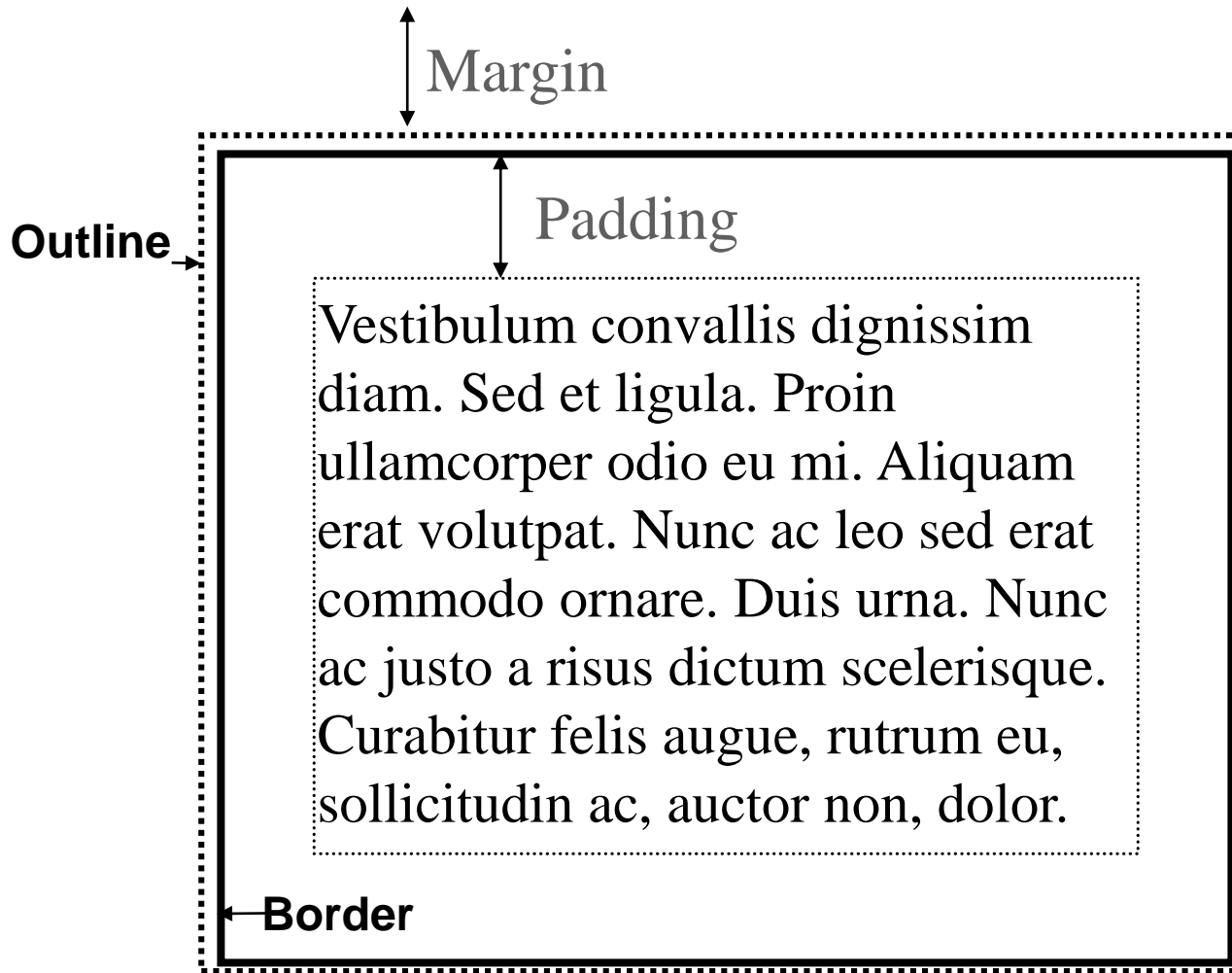
background-repeat: {repeticionImagen1} , {repeticionImagen2};

background-attachment: {fijacionImagen1} , {fijacionImagen2};

background-position: {posicionImagen1} , {posicionImagen2};

+info: https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Backgrounds_and_Borders/Using_multiple_backgrounds

El “Modelo de Cajas” de HTML/CSS...



Retomaremos este modelo cuando veamos maquetación con CSS

Anexo: Márgenes y Padding

Con estas propiedades especificamos los márgenes de cualquier elemento y con el padding controlamos la distancia entre el borde y el contenido.

- **margin-top** : **XX unidades** | % | **auto** (el valor por defecto es 0)

Fijamos el margen superior.

Con la propiedad **auto** el navegador da un valor adecuado dependiendo del tipo de elemento.

... de igual forma... **margin-bottom** **margin-left** **margin-right**

- **“Shorthand”**: **margin** : ancho1 ancho2 ancho3 ancho4

Podemos especificar todos los valores de los márgenes de una sola vez.

Se puede poner 1 valor(mismo margen a todas) , 2 valores (top/bottom left-right) o 4 valores. Esto es un ejemplo de propiedad “SHORTHAND”

https://developer.mozilla.org/es/docs/Web/CSS/Shorthand_properties

- **padding-top** : **XX unidades** | %

Distancia entre el borde superior y el contenido.

- ... de igual forma... **padding-bottom** **padding-left** **padding-right**

... y también tenemos **padding** = XX unidades da valor a todas a la vez

Anexo: Bordes (**border** y ahora también **outline**)

Con estas propiedades especificamos el ancho, color y estilo del area de borde.

- **border-top-width : thin | medium | thick | XX *unidad***

Fijamos el ancho del borde superior.

- ...**border-bottom-width border-left-width border-right-width**

- ...y también... **border-width : XX unidades** *da valor a todas a la vez*

- **border-style : none | dotted | dashed | solid | double | groove |
ridge | inset | outset**

Especificamos el estilo de visualización del borde

- **border-color : *NomColor* | #ValorHexa | rgb(R%,G%,B%) | rgb(R,G,B)**

Especificamos el color del borde.

☞ Además tenemos todas éstas propiedades en sus versiones “**shorthand**” :

- ***border: 5px solid red;***

Mediante esta propiedad podemos especificar de forma conjunta todas las propiedades de los 4 bordes, hay que poner 3 parámetros *anchura | estilo | color*

- ***border-top border-bottom border-left border-right***

También, dado el caso, podemos especificar uno de los lados del borde. También con los 3 parámetros: *anchura | estilo | color*

- ***outline: 5px dotted red;*** línea por fuera del borde. Param's: *anchura | estilo | color*