

Diseño y Programación Web

Tema 4 **CSS (II)**

Enrique Moguel
enrique@unex.es

CSS

– Propiedades CSS:

- Tenemos varias propiedades para definir el **estilo del texto**:
 - **Color** permite definir el color del texto.
 - **Text-align** permite definir la alineación horizontal.
 - » *Left, right, center, justified*
 - **Text-decoration** permite añadir o eliminar decoraciones al texto.
 - » *Overline, line-through, underline, none*

CSS

– Propiedades CSS:

- Tenemos varias propiedades para definir el estilo del texto.
 - **Text-transform** permite modificar el texto.
 - » *Uppercase, lowercase, capitalize*
 - **Text-indent** permite sangrar la primera línea de texto.
 - **Letter-spacing** permite definir el espacio entre caracteres.
 - **Line-height** permite definir el espacio entre líneas.
 - **Word-spacing** permite definir el espacio entre palabras.
- http://www.w3schools.com/css/exercise.asp?filename=exercise_text1

CSS

– Propiedades CSS:

- Hay propiedades para trabajar con las fuentes.

Difference Between Serif and Sans-serif Fonts



- Dos tipos de familias de fuentes.
 - » Genéricas.

Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

CSS

– Propiedades CSS:

- Con la propiedad **font-family** se puede especificar la fuente específica a utilizar.
 - Debe tener alternativas por si un buscador no encuentra una fuente determinada.

```
p {  
    font-family: "Times New Roman", Times, serif;  
}
```

- Con la propiedad **font-style** podemos definir el estilo de la fuente.
 - *Normal, italic, oblique*

CSS

– Propiedades CSS:

- Con **font-size** definimos el tamaño del texto.
- Con **font-weight** definimos el peso de la fuente.
 - *Normal, bold*
- http://www.w3schools.com/css/exercise.asp?filename=exercise_font1
- <https://fonts.google.com>

CSS

– Propiedades CSS:

- Todas las propiedades de texto y fuentes pueden aplicarse a los enlaces HTML.
- También se pueden aplicar estilos específicos para los distintos estados de un enlace.
 - **a:link** – Normal, no visitado.
 - **a:visited** – Normal, visitado.
 - **a:hover** – Cuando el ratón esta sobre el enlace.
 - **a:active** – Cuando el enlace es pulsado.

CSS

– Propiedades CSS:

- Podemos combinar distintas propiedades para mostrar los enlaces como botones.

```
a:link, a:visited {  
    background-color: #f44336;  
    color: white;  
    padding: 14px 25px;  
    text-align: center;  
    text-decoration: none;  
    display: inline-block;  
}
```

```
a:hover, a:active {  
    background-color: red;  
}
```


CSS

– Propiedades CSS:

- http://www.w3schools.com/css/tryit.asp?filename=trycss_link2
- http://www.w3schools.com/css/tryit.asp?filename=trycss_link_advanced2
- http://www.w3schools.com/css/exercise.asp?filename=exercise_link1

CSS

– Tablas HTML:

- La apariencia de una **tabla** HTML puede modificarse significativamente mediante CSS.

```
table, th, td {  
    border: 1px solid black;  
}
```

```
table {  
    border-collapse: collapse;  
}  
  
table, th, td {  
    border: 1px solid black;  
}
```

```
tr:hover {background-color: #f5f5f5}
```

CSS

– Tablas HTML:

- http://www.w3schools.com/css/tryit.asp?filename=trycss_table_fancy
- http://www.w3schools.com/css/tryit.asp?filename=trycss_table_caption-side
- http://www.w3schools.com/css/exercise.asp?filename=exercise_table1

CSS

– Display:

- La propiedad **display** es la más importante para gestionar el **layout** de una página.
- Especifica si un elemento se muestra y como debe mostrarse.
- Todos los elementos HTML tienen un valor por defecto de la propiedad **display**.
 - *Block / inline*

CSS

– Display:

- **Block**

- Empieza en una nueva línea y ocupa todo el ancho de la página.

<div>

<h1> - <h6>

<p>

<form>

<header>

<footer>

<section>

CSS

– Display:

- **Inline**

- No empieza en una nueva línea y ocupa solo el ancho que necesita.

``

`<a>`

`<image>`

- **None**

- Oculta un elemento para el usuario.
- Se puede manipular con *JavaScript*.

CSS

– Display:

```
li {  
    display: inline;  
}
```

```
a {  
    display: block;  
}
```

```
h1.hidden {  
    display: none;  
}
```

```
h1.hidden {  
    visibility: hidden;  
}
```

- http://www.w3schools.com/css/tryit.asp?filename=trycss_display
- http://www.w3schools.com/css/tryit.asp?filename=trycss_display_js
- http://www.w3schools.com/css/exercise.asp?filename=exercise_display_visibility1

CSS

- Otra propiedad muy relevante para gestionar el **layout** de una página es la propiedad **position**.
 - Especifica el método de posicionamiento a utilizar:
 - **static** - por defecto, se posiciona en relación al flujo normal de la página.
 - **relative** - relativo a su posición normal, permite moverlo de su posición habitual.
 - **fixed** – posición relativa al *viewport*, siempre en el mismo sitio incluso con *scroll*.
 - **absolute** – posición relativa a su padre.

CSS

```
div.static {  
  position: static;  
  border: 3px solid #73AD21;  
}
```

```
div.relative {  
  position: relative;  
  left: 30px;  
  border: 3px solid #73AD21;  
}
```

```
div.fixed {  
  position: fixed;  
  bottom: 0;  
  right: 0;  
  width: 300px;  
  border: 3px solid #73AD21;  
}
```

```
div.relative {  
  position: relative;  
  width: 400px;  
  height: 200px;  
  border: 3px solid #73AD21;  
}  
  
div.absolute {  
  position: absolute;  
  top: 80px;  
  right: 0;  
  width: 200px;  
  height: 100px;  
  border: 3px solid #73AD21;  
}
```

CSS

– Utilizando estas propiedades podemos tener páginas en los que uno elementos se solapen con otros.

- La propiedad **z-index** permite indicar que elementos se colocan “*encima*” de otros.

```
img {  
    position: absolute;  
    left: 0px;  
    top: 0px;  
    z-index: -1;  
}
```

- http://www.w3schools.com/css/tryit.asp?filename=trycss_overflow

CSS

- Con la propiedad **overflow** podemos controlar el comportamiento de un elemento cuando su contenido es demasiado grande.
 - **visible** – por defecto, el contenido se muestra fuera del contenedor.
 - **hidden** – el contenido sobrante no se muestra.
 - **scroll** – se añade una barra de *scroll* para mostrar el contenido sobrante.
 - **auto** – si el contenido no cabe se añade una barra de *scroll*.

CSS

You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.

```
img {  
  position: absolute;  
  left: 0px;  
  top: 0px;  
  z-index: -1;  
}
```

```
div {  
  overflow: hidden;  
}
```

You can use the overflow property when you want to have better control of the layout. The overflow

CSS

You can use the overflow property when you want to have better control of the

```
div {  
    overflow: scroll;  
}
```

```
div {  
    overflow: auto;  
}
```

You can use the overflow property when you want to have better control of the

CSS

- Se pueden usar las propiedades **overflow-x** y **overflow-y** para definir distintos comportamientos para el desbordamiento horizontal y vertical.

```
div {  
    overflow-x: hidden; /* Hide horizontal scrollbar */  
    overflow-y: scroll; /* Add vertical scrollbar */  
}
```

- http://www.w3schools.com/css/exercise.asp?filename=exercise_overflow1

CSS

- La propiedad **float** se utiliza para especificar si un elemento debe flotar.
- La propiedad **clear** se utiliza para controlar el comportamiento de elementos flotantes.

```
img {  
    float: right;  
    margin: 0 0 10px 10px;  
}
```

```
div {  
    clear: left;  
}
```

CSS

```
.div1 {  
  float: left;  
  width: 100px;  
  height: 50px;  
  margin: 10px;  
  border: 3px solid #73AD21;  
}  
  
.div2 {  
  border: 1px solid red;  
}  
  
.div3 {  
  float: left;  
  width: 100px;  
  height: 50px;  
  margin: 10px;  
  border: 3px solid #73AD21;  
}  
  
.div4 {  
  border: 1px solid red;  
  clear: left;  
}
```

Without clear



div1

div2 - Notice that the div2 element is after div1, in the HTML code. However, since div1 is floated to the left, this happens: the text in div2 is floated around div1, and div2 surrounds the whole thing.

Using clear



div3

div4 - Using clear moves div4 down below the floated div3. The value "left" clears elements floated to the left. You can also clear "right" and "both".

CSS

- http://www.w3schools.com/css/tryit.asp?filename=trycss_layout_float_ex
- http://www.w3schools.com/css/tryit.asp?filename=trycss_float4
- http://www.w3schools.com/css/tryit.asp?filename=trycss_float5
- http://www.w3schools.com/css/tryit.asp?filename=trycss_float6
- http://www.w3schools.com/css/tryit.asp?filename=trycss_inline-block_old
- http://www.w3schools.com/css/tryit.asp?filename=trycss_inline-block