

Diseño y Programación Web

Tema 4 **CSS (III)**

Enrique Moguel
enrique@unex.es

CSS

- Una de las tareas mas habituales a realizar con CSS es la alineación de elementos.
- Vertical y horizontal.
- Existen varias propiedades CSS y mecanismos para alinear elementos.
- Los elementos de tipo block pueden centrarse horizontalmente con **margin: auto;**

```
.center {  
    margin: auto;  
    width: 50%;  
    border: 3px solid green;  
    padding: 10px;  
}
```

CSS

- El texto dentro de un elemento puede centrarse con **text-align: center;**

```
.center {  
    text-align: center;  
    border: 3px solid green;  
}
```

- Una imagen puede centrarse convirtiéndola en elemento block y usando **margin: auto;**

```
img {  
    display: block;  
    margin: auto;  
    width: 40%;  
}
```

CSS

- Para alinear elementos a la izquierda o la derecha se puede usar **position: absolute;**

```
.right {  
    position: absolute;  
    right: 0px;  
    width: 300px;  
    border: 3px solid #73AD21;  
    padding: 10px;  
}
```

- También usando **float;**

```
.right {  
    float: right;  
    width: 300px;  
    border: 3px solid #73AD21;  
    padding: 10px;  
}
```

CSS

- Para centrar elementos de forma vertical se puede usar **padding**;

```
.center {  
    padding: 70px 0;  
    border: 3px solid green;  
}
```

- También **line-height**;

```
.center {  
    line-height: 200px;  
    height: 200px;  
    border: 3px solid green;  
    text-align: center;  
}
```

CSS

- También con **transform;**

```
.center {  
    height: 200px;  
    position: relative;  
    border: 3px solid green;  
}  
  
.center p {  
    margin: 0;  
    position: absolute;  
    top: 50%;  
    left: 50%;  
    transform: translate(-50%, -50%);  
}
```

- http://www.w3schools.com/css/exercise.asp?filename=exercise_align1

CSS

- En CSS existen **pseudo-classes**.
 - Permiten definir un estado especial de un elemento:
 - Cuando el ratón está encima.
 - Cuando ya se ha visitado un link.
 - Cuando tiene el foco.

```
selector:pseudo-class {  
    property:value;  
}
```

CSS

- Las **pseudo-classes** se pueden combinar con clases normales y con selectores complejos.

```
a.highlight:hover {  
    color: #ff0000;  
}
```

```
p {  
    display: none;  
    background-color: yellow;  
    padding: 20px;  
}  
  
div:hover p {  
    display: block;  
}
```


CSS

- La **pseudo-class : first-child** permite identificar elementos de un tipo que son los primeros hijos de elementos de otro tipo.

```
p:first-child {  
    color: blue;  
}
```

```
p i:first-child {  
    color: blue;  
}
```

```
p:first-child i {  
    color: blue;  
}
```

- https://www.w3schools.com/cssref/tryit.asp?filename=trycss_sel_firstchild

CSS

- La **pseudo-class : lang** permite especificar reglas especiales para diferentes idiomas.

```
<html>
<head>
<style>
q:lang(no) {
    quotes: "~" "~";
}
</style>
</head>

<body>
<p>Some text <q lang="no">A quote in a paragraph</q> Some text.</p>
</body>
</html>
```

CSS

- http://www.w3schools.com/css/tryit.asp?filename=trycss_link2
- http://www.w3schools.com/css/tryit.asp?filename=trycss_link_focus
- http://www.w3schools.com/css/exercise.asp?filename=exercise_pseudo_classes1

CSS

- También existen los **pseudo-elementos**.
- Permite definir estilos de partes específicas de un elemento.
 - Primera letra de una línea o elemento.
 - Insertar contenido antes o después del contenido de un elemento.

```
selector::pseudo-element {  
    property:value;  
}
```

CSS

- El **pseudo-elemento :: first-line** permite definir un estilo específico para la primera línea de un texto.

```
p::first-line {  
    color: #ff0000;  
    font-variant: small-caps;  
}
```

- Solo aplicables a elementos de tipo bloque.
 - Permite modificar las propiedades relacionadas con el texto no el resto.

CSS

- El **pseudo-elemento :: first-letter** permite definir un estilo específico para la primera letra de un texto.

```
p::first-letter {  
    color: #ff0000;  
    font-size: xx-large;  
}
```

- Solo aplicables a elementos de tipo bloque.
 - Permite modificar las propiedades relacionadas con el texto no el resto.

CSS

- Los **pseudo-elementos** se pueden combinar con clases CSS o entre si.

```
p.intro::first-letter {  
    color: #ff0000;  
    font-size: 200%;  
}
```

```
p::first-letter {  
    color: #ff0000;  
    font-size: xx-large;  
}  
  
p::first-line {  
    color: #0000ff;  
    font-variant: small-caps;  
}
```

CSS

- El **pseudo-elemento :: before** permite insertar contenidos antes del contenido de un elemento.

```
h1::before {  
    content: url(smiley.gif);  
}
```

- El **pseudo-elemento :: after** permite insertar contenidos después del contenido de un elemento.

```
h1::after {  
    content: url(smiley.gif);  
}
```


CSS

- El **pseudo-elemento :: selection** permite modificar el estilo de la porción de un elemento que es seleccionada por un usuario.
 - Permite cambiar *color, background, cursor y outline*.

```
::selection {  
    color: red;  
    background: yellow;  
}
```

- http://www.w3schools.com/css/tryit.asp?filename=try_css3_selection
- http://www.w3schools.com/css/exercise.asp?filename=exercise_pseudo_elements1

CSS

– La propiedad **opacity** permite especificar la opacidad/transparencia de un elemento.

- Valores de 0 a 1 (1 por defecto).

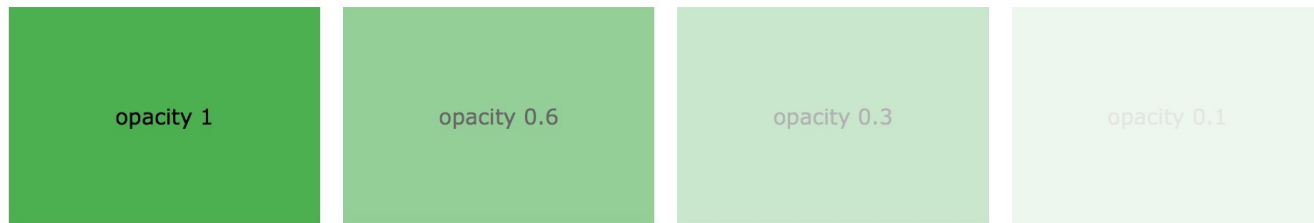
```
img {  
    opacity: 0.5;  
    filter: alpha(opacity=50); /* For IE8 and earlier */  
}
```

- Se usa con frecuencia en combinación con **hover**.

```
img {  
    opacity: 0.5;  
    filter: alpha(opacity=50); /* For IE8 and earlier */  
}  
  
img:hover {  
    opacity: 1.0;  
    filter: alpha(opacity=100); /* For IE8 and earlier */  
}
```

CSS

- Cuando se usa **opacity** en un elemento contenedor afecta a todos sus hijos.



- Para evitarlo se puede aplicar la opacidad con el color RGBA.

```
div {  
  background: rgba(76, 175, 80, 0.3) /* Green background with 30% opacity */  
}
```



CSS

- Es posible construir selectores para manipular elementos con determinados atributos o valores de los atributos.

```
a[target] {  
    background-color: yellow;  
}
```

```
a[target="_blank"] {  
    background-color: yellow;  
}
```

CSS

- También podemos seleccionar elementos cuyos atributos contengan una palabra determinada.

```
[title~="flower"] {  
    border: 5px solid yellow;  
}
```

- O que comiencen por una palabra.

```
[class="top"] {  
    background: yellow;  
}
```

CSS

- O por una cadena.

```
[class^="top"] {  
    background: yellow;  
}
```

- O que terminen por una cadena.

```
[class$="test"] {  
    background: yellow;  
}
```

- O que contengan una cadena.

```
[class*="te"] {  
    background: yellow;  
}
```

CSS

– Ejemplos interesantes:

- Barras de navegación:
 - http://www.w3schools.com/css/css_navbar.asp
- Desplegables:
 - http://www.w3schools.com/css/css_dropdowns.asp
- Tooltips:
 - http://www.w3schools.com/css/css_tooltip.asp
- Galerías de imágenes:
 - http://www.w3schools.com/css/css_image_gallery.asp

CSS

– Ejemplos interesantes:

- Formularios:

- http://www.w3schools.com/css/css_form.asp

- Sistema solar:

- <http://codepen.io/juliangarnier/pen/idhuG>

- Page flip:

- <http://codepen.io/fbrz/pen/whxbF>