

CSS BÁSICO

LIMA UD2 – Tema 1

IES Plurilingüe Antón Losada Diéguez

Adrián Fernández González



Tabla de contenido

1. Introducción.....	2
1.1. Unidades básicas	2
1.1.1. Píxeles	2
1.1.2. Porcentual	2
1.2. Formatos de color	2
1.2.1. Textual	2
1.2.2. RGB	3
1.2.3. RGBA	3
1.2.4. Hexadecimal	3
1.3. Formatos específicos	3
1.4. Formatos abreviados	4
2. Color y fondo	4
2.1. Color	4
2.2. Background	5
2.2.1. Color.....	5
2.2.2. Imagen	5
2.3. Opacity o RGBA	6
3. Bordes	6
3.1. Bordes redondeados	7
4. Tamaño.....	7
5. Márgenes.....	8
5.1 Modelo de caja.....	8
6. Texto.....	8
6.1. Fuente.....	9
7. Listas.....	10

CSS Básico

1. Introducción

Existen infinidad de propiedades de CSS, por lo que se irán dividiendo en documentos en función de su nivel de conocimientos necesarios, utilidad o complejidad.

En este tema se verán los más simples y los conocimientos básicos para utilizarlos.

1.1. Unidades básicas

Hay varias unidades en las que estipular los valores de CSS en cuanto a colocación en pantalla y tamaños, pero las dos más básicas son píxeles y porcentual.

1.1.1. Píxeles

Los píxeles son la unidad básica de medida en CSS y representan los puntos del monitor. Están estandarizados y definen la unidad absoluta de medida.

1.1.2. Porcentual

Las unidades porcentuales son las que representan fracciones de la pantalla, es decir, porcentajes de tamaño frente a la ventana del navegador, la página, el elemento que lo contiene, etc.

Esta unidad es relativa, puesto que depende de los elementos que la contengan.

1.2. Formatos de color

Los colores en CSS se pueden estipular de múltiples formas, aunque las más habituales son textual, RGB, RGBA y Hexadecimal.

Estos se pueden usar indistintamente en aquellas propiedades que establezcan algún tipo de color.

1.2.1. Textual

El formato textual es el más básico, más fácil de utilizar pero el menos personalizable y que dificulta la modificación dinámica.

El estándar contiene 140 colores mediante nombres.



1.2.2. RGB

Este es el formato básico en la teoría de color, en la que cada color se representa mediante la combinación de rojo (Red) verde (Green) y azul (Blue).

Cada uno de estos colores toma un valor entre 0 y 255 y se estipula mediante `rgb(R, G, B)`;

```
p {  
    color: rgb(255, 0, 0);  
}
```

El 0 representa la ausencia de color y el 255 el color absoluto, siendo 255, 255, 255 el blanco y 0, 0, 0 el negro.

1.2.3. RGBA

Este formato es una extensión del anterior, añadiendo un cuarto valor que indica el alfa, la transparencia.

A diferencia de los otros tres, este valor va de 0, completamente transparente (invisible) a 1, completamente visible (como si no se pusiese alfa).

```
p {  
    color: rgba(0, 0, 255, 0.5);  
}
```

1.2.4. Hexadecimal

Este formato es el más sencillo y corto, por lo que es el más utilizado para colores **sin transparencia**.

El color se estipula en hexadecimal, con un par de dígitos por color y anteponiendo una almohadilla # antes.

```
p {  
    color: #00AACC;  
}
```

En este caso, el rojo valdrá 0, el verde AA y el azul CC.

Este método permite abreviarse, poniendo solo un dígito por color en los casos en los que ambos dígitos sean los mismos para los tres colores.

```
p {  
    color: #0AC;  
}
```

Este ejemplo es el mismo que el anterior.

1.3. Formatos específicos

Muchas propiedades tienen “subpropiedades” o “extensiones” que permiten establecer de forma independiente ciertas características más específicas.

Por ejemplo, la propiedad para establecer bordes permite seleccionar de forma independiente cada uno de los cuatro lados y aplicarles un valor diferente.

```
p {  
  border-top-style: dotted;  
  border-right-style: solid;  
  border-bottom-style: dotted;  
  border-left-style: solid;  
}
```

Hay que tener en cuenta que esta selección ha de estipularse justo después del nombre general y luego las cualidades específicas.

1.4. Formatos abreviados

Aquellas propiedades que permiten especificar diferentes cualidades, pueden aunarse en una, con el nombre general de la propiedad, estipulando los diferentes valores separados por espacios.

Por ejemplo, en los bordes, podemos estipular el ancho, el estilo y el color todo en una misma propiedad en vez de en tres distintas.

```
p {  
  border-width: 5px;  
  border-style: solid;  
  border-color: red;  
}  
  
p {  
  border: 5px solid red;  
}
```

Cada propiedad tiene su orden establecido y ha de respetarse o no funcionará.

Este formato abreviado puede combinarse con las propiedades específicas, de tal modo que podemos aplicar esas tres cualidades al borde superior, por ejemplo.

```
p {  
  border-top: 5px solid red;  
}
```

La mayoría de las propiedades que tienen guiones entre las palabras son susceptibles de agruparse.

2. Color y fondo

Estas propiedades estipulan el color del contenido y el fondo.

2.1. Color

Color del contenido del elemento, habitualmente el texto del mismo.

```
p {  
    color: #00AACC;  
}
```

2.2. Background

Un conjunto de propiedades que estipulan el fondo del elemento sobre el que irán el resto de cosas.

2.2.1. Color

Color del fondo del elemento.

```
body {  
    background-color: lightblue;  
}
```

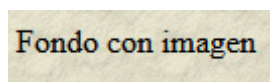


2.2.2. Imagen

Estas propiedades permiten establecer una imagen como fondo del elemento y establecer sus características.

La propiedad *background-image* permite especificar una imagen de fondo pasándole una URL.

```
body {  
    background-image: url("paper.gif");  
}
```



Esta propiedad puede combinarse con *background-color* de tal forma que si fallase la imagen, el color saldría como fondo y, si la imagen tiene partes transparentes, se vería a través.

Combinándola con *background-repeat* se puede estipular la forma de repetición entre:

- **repeat:** Valor por defecto, repite vertical y horizontalmente cubriendo el 100% del elemento.
- **repeat-x:** Repite solo horizontalmente.
- **repeat-y:** Repite solo verticalmente.
- **no-repeat:** No repite, simplemente pone la imagen una vez.

Esta, combinada con *background-position*, permite establecer el lugar en el que empieza la imagen.

La propiedad *background-attachment* permite establecer si la imagen se mueve o no al hacer scroll mediante *scroll* (por defecto) o *fixed*.

2.3. Opacity o RGBA

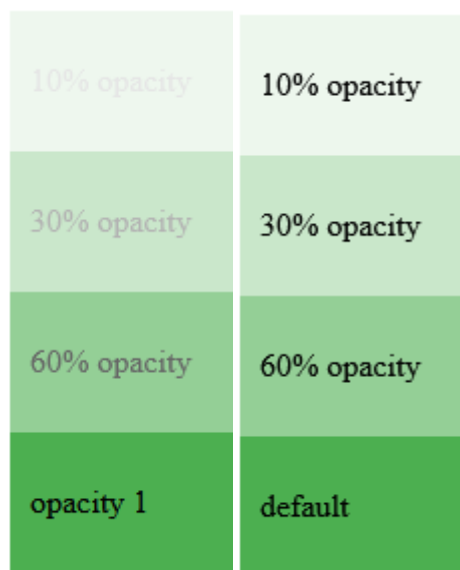
Existen dos formas de hacer algo transparente pero que actúan de formas distintas.

Por un lado, mediante la propiedad *opacity* el elemento entero junto a sus hijos se vuelve transparente. Esta se estipula mediante un valor entre 0 y 1, siendo 0 invisible y 1 totalmente visible.

```
div {  
  background-color: green;  
  opacity: 0.3;  
}
```

El problema de esta propiedad es que puede ser difícil de ver con unos niveles altos de transparencia y sobre todo si el elemento incluye texto.

Si lo que se quiere hacer es el fondo transparente, lo recomendable es utilizar RGBA como color, ya que solo su fondo se vuelve transparente.



Tanto uno como el otro, dejan ver a través, por lo que si hay algo detrás, como una imagen, se verá.

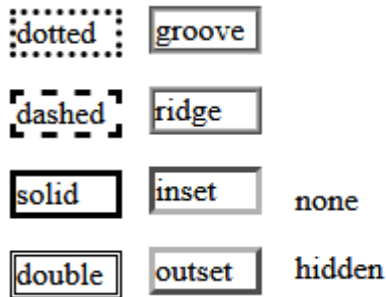
3. Bordes

Estas propiedades permiten establecer las características de los bordes, desde su estilo hasta su grosor.

Todas permiten aplicarse a todos los bordes por igual, o a uno en concreto.

```
table {
  border-style: solid;
  border-width: 1px;
  border-color: black;
}
```

La propiedad *border-style* es necesaria para que las otras funcionen e indica el tipo de borde.



Con la propiedad *border-width* se puede indicar el grosor del borde.

La propiedad *border-color* estipula el color del borde. Por defecto es negro.

Hay que tener en cuenta que el grosor del borde se añade al tamaño total del elemento, a mayores del ancho y alto.

3.1. Bordes redondeados

Con la propiedad *border-radius* se puede estipular la redondez de los bordes.

```
div {
  border: 2px solid red;
  border-radius: 8px;
}
```

Bordes
redondeados

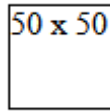
Esta propiedad puede establecerse sin un borde establecido, de tal forma que, si el elemento tiene un color de fondo, se vea redondeado.

Esta propiedad puede establecerse de forma independiente a cada esquina.

4. Tamaño

Con las propiedades *width* y *height* se puede estipular el ancho y el alto del elemento.

```
div {
  width: 50px;
  height: 50px;
}
```

Nota, estas propiedades no funcionan con los elementos inline.

5. Márgenes

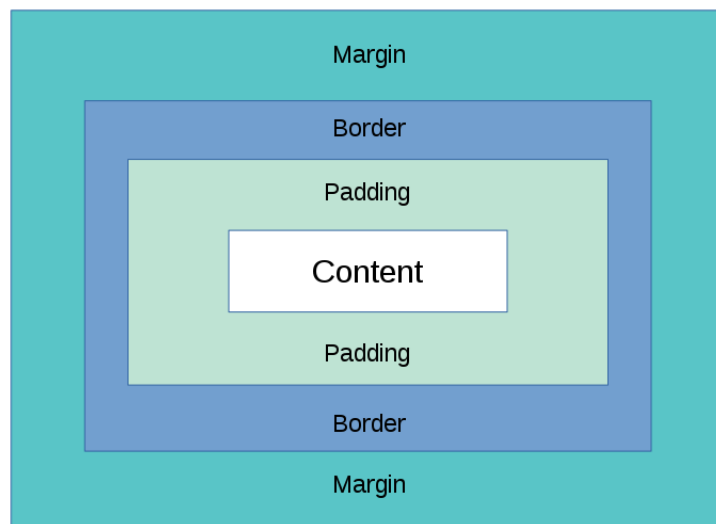
Las propiedades *margin* y *padding* establecen el margen externo e interno respectivamente.

```
div {  
  margin: 10px;  
  padding: 10px;  
}
```

Hay que tener en cuenta que el *padding* se añade al tamaño total del elemento, a mayores del ancho y alto.

5.1 Modelo de caja

El modelo de caja es la estructura de un elemento.



El contenido representa al elemento, su ancho y alto.

Luego, viene el *padding*, el margen interno que hay desde el borde al propio elemento.

A continuación, el borde y su grosor.

Por último, el *margin* o margen externo.

6. Texto

Además del color visto anteriormente, hay otras propiedades básicas.

Con *text-decoration* se puede añadir una línea sobre el texto, una que lo atraviese o una bajo él dependiendo del valor que tome.

```
span {  
  text-decoration: underline;  
}
```

Overline

~~Line-through~~

Underline

Con esta propiedad puede quitarse el subrayado a los enlaces, estableciéndola a *none* para ellos.

Otra propiedad básica es la alineación de texto para colocarlo a izquierda, centrado, derecha o justificado.

```
h1 {  
  text-align: center;  
}
```

6.1. Fuente

Estas propiedades permiten modificar el tipo de letra, tamaño, peso y estilo.

Con la propiedad *font-family* se puede estipular el tipo de letra a utilizar.

```
.mono {  
  font-family: "Lucida Console", "Courier New", monospace;  
}
```

Es recomendable estipular varias fuentes por si no fuesen compatibles con el navegador, haciendo que escoja la siguiente de la lista. También se puede estipular directamente el tipo:

- **Serif:** Fuentes con decoración en los bordes de las letras. Utilizados en imprenta por ser más formales y elegantes.
- **Sans-serif:** Fuentes con letras limpias sin decoración. Dan un aspecto minimalista, moderno y facilitan la lectura en los ordenadores.
- **Monospace:** Fuentes que establecen el mismo espacio para las letras. En las otras fuentes, letras como la i o la m tienen tamaños distintos. Tienen un aspecto mecánico.
- **Cursive:** Fuentes inclinadas y con un aspecto de escritura humana, aunque más difíciles de leer.
- **Fantasy:** Fuentes decorativas, con aspecto único, aunque difíciles de leer en algunos casos.



Hay que tener en cuenta que la fuente tiene que estar instalada en el sistema del cliente para que este pueda verla.

Con la propiedad *font-size* se puede estipular el tamaño de la letra. Por defecto son 16px.

```
h1 {  
    font-size: 40px;  
}
```

La propiedad *font-weight* permite poner el texto en negrita o en normal. Tiene un tercer valor, lighter, pero su efecto no es apreciable.

```
.negrita {  
    font-weight: bold;  
}
```

normal

lighter

bold

Con *font-style* se puede estipular si el texto es normal, cursivo u oblicuo.

```
p.italic {  
    font-style: italic;  
}
```

normal

italic

oblique

La diferencia en *italic* y *oblique* es que, la cursiva viene establecida en la fuente por el diseñador, con ciertas modificaciones en algunos caracteres, mientras que la oblicua se inclinando el texto normal.

7. Listas

Los iconos, denominados viñetas, de las listas, tanto ordenadas como no, pueden ser modificados.

Mediante la propiedad *list-style-type* se puede estipular el estilo del icono que aparece en cada elemento de la lista. Las listas ordenadas y las listas no ordenadas tienen sus propios valores.

```
ul {
  list-style-type: square;
}
```

- Café
- Té
- Coca Cola

```
ol {
  list-style-type: upper-roman;
}
```

- I. Café
- II. Té
- III. Coca Cola

A mayores, con *list-style-image* se puede estipular una imagen personalizada.

```
ul {
  list-style-image: url('sqpurple.gif');
}
```

- Café
- Té
- Coca Cola

Es una buena práctica estipular ambas propiedades, de tal forma que si la imagen falla, queda el icono de respaldo.

La propiedad *list-style-position* estipula si las viñetas están dentro del elemento lista o fuera. Por defecto, toma el valor *outside*.

```
ul {
  list-style-position: outside;
}
```

- Coffee - A t
- Tea - An arc
Camellia sir
- Coca Cola -
original ingi

```
ul {
  list-style-position: inside;
}
```

- Coffee - A
plant
- Tea - An ;
Camellia sir
- Coca Col;
its original i