

TEMA1. INTRODUCCIÓN.

1. FUNCIONAMIENTO BÁSICO DE CSS.....	2
2. INTEGRACIÓN DE ESTILOS CSS.....	3
2.1. CSS DENTRO DEL DOCUMENTO HTML.....	3
2.2. CSS EN UN DOCUMENTO EXTERNO.....	3
2.3. CSS EN LÍNEA.....	3
3. SINTAXIS REGLAS CSS.....	5
4. MEDIOS CSS.....	5
4.1. MEDIOS DEFINIDOS CON LAS REGLAS DE TIPO @MEDIA.....	5
4.2. MEDIOS DEFINIDOS CON LAS REGLAS DE TIPO @IMPORT.....	6
4.3. MEDIOS DEFINIDOS CON LA ETIQUETA <link>.....	6

1. Funcionamiento básico de CSS

Una forma de asignar estilos a nuestro documento es utilizar estilos HTML, esta forma de asignar estilos está obsoleta y no se debe utilizar nunca.

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Documento prueba </title>
</head>
<body>
  <!--este Código está obsoleto, no utilizarlo -->
  <h1>
    <font color="red" face="Arial" size="5">Titular de la
    página</font>
  </h1>
  <p>
    <font color="gray" face="Verdana" size="2">
      Un párrafo de texto no muy largo.
    </font>
  </p>
</body>
</html>
```

En el ejemplo podemos ver como se utiliza la etiqueta `` con sus atributos `color`, `face` y `size` para definir el color, el tipo y el tamaño de letra de cada elemento de la página.

El problema al utilizar estilos HTML es que cada vez que queramos aplicar el estilo tenemos que repetirlo y esto hace que nuestro código pueda crecer innecesariamente debido a la repetición de código, por otra forma el mantenimiento es mucho más costoso, ya que la modificación o eliminación de un valor repetido por toda nuestra página puede ser una labor muy pesada.

La solución que propone CSS es mejor, como se puede ver en el siguiente ejemplo:

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>Ejemplo de estilos con CSS</title>
  <style type="text/css">
    h1 { color: red; font-family: Arial; font-size: large; }
    p { color: gray; font-family: Verdana; font-size: medium; }
  </style>
</head>
<body>
  <h1>Titular de la página</h1>
  <p>Un párrafo de texto no muy largo.</p>
</body>
</html>
```

- ✓ CSS nos permite diferenciar entre los estilos (el formato) de la página y la estructura de la página (el código HTML) sin mezclar unos con otros.
- ✓ Utilizando CSS, se pueden establecer los mismos estilos con menos esfuerzo y sin ensuciar el código HTML de los contenidos con etiquetas ``.

En el ejemplo anterior, dentro de la etiqueta `style` se indica que todas las etiquetas `<h1>` de la página se deben ver de color rojo, con un tipo de letra Arial y con un tamaño de letra grande. Además, las etiquetas `<p>` de la página se deben ver de color gris, con un tipo de letra Verdana y con un tamaño de letra medio.

Definir los estilos de esta forma:

- ✓ Facilita la escritura de los estilos.
- ✓ Evita repeticiones.
- ✓ Nos permite diferenciar el estilo de la estructura de la página
- ✓ Mejora el mantenimiento.

2. Integración de estilos CSS

Tenemos tres formas de integrar el código CSS en nuestro código HTML:

- ✓ En línea.
- ✓ Dentro del documento HTML.
- ✓ Documento externo.

2.1. CSS en línea.

Este método añade el código CSS en la etiqueta HTML, y como podemos imaginarnos, tiene los mismos problemas que nos planteaba la utilización de los atributos para estilos de HTML. Este método no es recomendable utilizarlo.

```
<!DOCTYPE html>
<head>
<meta charset="UTF-8">
<title>CSS en línea</title>
</head>
<body>
<p style="color: black; font-family: Verdana;">Un párrafo de
texto.</p>
</body>
</html>
```

2.2. CSS dentro del documento HTML.

Utilizamos la etiqueta <style> dentro del <head> de nuestro documento HTML. De esta forma los estilos se aplican a todo el documento en el que se encuentran.

```
<!DOCTYPE html>
<head>
<meta charset="UTF-8">
<title>CSS en el propio documento</title>
<style>
  p { color: black; font-family: Verdana; }
</style>
</head>

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

Este método se emplea cuando se define un número pequeño de estilos o cuando se quieren incluir estilos específicos en una determinada página HTML que completen los estilos que se incluyen por defecto en todas las páginas del sitio web.

El principal inconveniente es que, si se quiere hacer una modificación en los estilos definidos, es necesario modificar todas las páginas que incluyen el estilo que se va a modificar.

2.3. CSS en un documento externo.

En este caso, todos los estilos CSS se incluyen en un archivo externo .css, que las páginas HTML enlazan mediante la etiqueta <link>. Podemos crear todos los archivos CSS que sean necesarios y cada página HTML puede enlazar tantos archivos CSS como necesite.

Para crear poder utilizar los estilos en un documento .css externo, vamos a tener que:

1. Crear un fichero .css con los estilos que necesitamos:

```
p { color: black; font-family: Verdana; }
```
2. Guardaremos el fichero .css. Normalmente vamos a crear una carpeta (que se suele llamar styles)
3. Y finalmente enlazaremos nuestro documento .css dentro de nuestro documento HTML con la etiqueta <link>, que añadiremos dentro del <head> y en el que utilizando el atributo href indicaremos la ruta donde se encuentra el fichero .css. De esta forma podremos acceder a todos los estilos que tengamos dentro del fichero .css

```
<!DOCTYPE html>
<head>
<meta charset= "UTF-8">
<title>CSS en un archivo externo</title>
<link href="css/estilos.css">
</head>
<body>
<p>Un párrafo de texto.</p>
</body>
</html>
```

De todas las formas de incluir CSS en las páginas HTML, esta es la más utilizada con mucha diferencia. La principal ventaja es que se puede incluir un mismo archivo CSS en multitud de páginas HTML, por lo que se garantiza la aplicación homogénea de los mismos estilos a todas las páginas que forman un sitio web.

Este método es el más utilizado, ya que nos permite reutilizar el código CSS en todos los documentos HTML que necesitemos. Además, el mantenimiento es mucho más sencillo, ya que un cambio en un archivo CSS permite actualizar los estilos de los documentos HTML que lo contengan.

Aunque generalmente se emplea la etiqueta <link> para enlazar los archivos CSS externos, también se puede utilizar la etiqueta <style>. La forma alternativa de incluir un archivo CSS externo se muestra a continuación:

```
<!DOCTYPE html>
<head>
<meta charset="UTF-8">
<title>CSStitle>
<style type="text/css" media="screen">
  @import '/css/estilos.css';
</style>
</head>
<body>
<p>Un párrafo de texto.</p>
</body>
</html>
```

En este caso, para incluir en la página HTML los estilos definidos en archivos CSS externos se utiliza una regla especial de tipo @import.

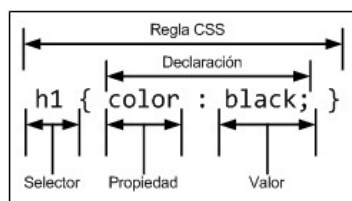
La URL del archivo CSS externo se indica mediante una cadena de texto encerrada con comillas simples o dobles o mediante la palabra reservada `url()`. De esta forma, las siguientes reglas `@import` son equivalentes:

```
@import "/css/estilos.css";  
@import url("/css/estilos.css");
```

El problema que tiene usar @import, es que, si no se puede cargar el fichero css, se para la carga de toda la página.

3. Sintaxis reglas CSS

CSS utiliza una sintaxis para crear reglas de estilo, el siguiente esquema muestra las partes que forman un estilo CSS muy básico:



Componentes de un estilo CSS básico

Los diferentes términos se definen a continuación:

- ✓ Regla: cada uno de los estilos que componen una hoja de estilos CSS. Cada regla está compuesta de una parte de "selectores", un símbolo de "llave de apertura" ({), otra parte denominada "declaración" y, por último, un símbolo de "llave de cierre" (}).
- ✓ Selector: indica el elemento o elementos HTML a los que se aplica la regla CSS.
- ✓ Propiedad: característica que se modifica en el elemento seleccionado, como por ejemplo su tamaño de letra, su color de fondo, etc.
- ✓ Valor: establece el nuevo valor de la característica modificada en el elemento.

Un archivo CSS puede contener un número ilimitado de reglas CSS, cada regla se puede aplicar a varios selectores diferentes y cada declaración puede incluir tantos pares propiedad/valor como se desee.

3.1. Medios CSS

Una de las características más importantes de las hojas de estilos CSS es que permiten definir diferentes estilos para diferentes medios o dispositivos: pantallas, impresoras, móviles, proyectores, etc.

Además, CSS define algunas propiedades específicamente para determinados medios, como por ejemplo la paginación y los saltos de página para los medios impresos o el volumen y tipo de voz para los medios de audio.

Los medios más utilizados actualmente son:

- ✓ screen (para definir el aspecto de la página en pantalla)
- ✓ print (para definir el aspecto de la página cuando se imprime)

La gran ventaja de CSS es que permite modificar los estilos de una página en función del medio en el que se visualiza:

3.2. Medios definidos con las reglas de tipo @media

Las reglas `@media` son un tipo especial de regla CSS que permiten indicar de forma directa el medio o medios en los que se aplicarán los estilos incluidos en la regla. Para especificar el medio en el que se aplican los estilos, se incluye su nombre después de `@media`. Si los estilos se aplican a varios medios, se incluyen los nombres de todos los medios separados por comas.

A continuación, se muestra un ejemplo sencillo:

```
@media print {  
  body { font-size: 10pt }  
}  
@media screen {  
  body { font-size: 13px }  
}  
@media screen, print {  
  body { line-height: 1.2 }  
}
```

El ejemplo anterior establece que el tamaño de letra de la página cuando se visualiza en una pantalla debe ser 13 píxel. Sin embargo, cuando se imprimen los contenidos de la página, su tamaño de letra debe ser de 10 puntos. Por último, tanto cuando la página se visualiza en una pantalla como cuando se imprimen sus contenidos, el interlineado del texto debe ser de 1.2 veces el tamaño de letra del texto.

3.3. Medios definidos con las reglas de tipo `@import`

Cuando se utilizan reglas de tipo `@import` para enlazar archivos CSS externos, se puede especificar el medio en el que se aplican los estilos indicando el nombre del medio después de la URL del archivo CSS:

```
@import url("estilos_basicos.css") screen;  
@import url("estilos_impresora.css") print;
```

Las reglas del ejemplo anterior establecen que cuando la página se visualiza por pantalla, se cargan los estilos definidos en el primer archivo CSS. Por otra parte, cuando la página se imprime, se tienen en cuenta los estilos que define el segundo archivo CSS.

Si los estilos del archivo CSS externo deben aplicarse en varios medios, se indican los nombres de todos los medios separados por comas. Si no se indica el medio en una regla de tipo `@import`, el navegador sobreentiende que el medio es `all`, es decir, que los estilos se aplican en todos los medios.

3.4. Medios definidos con la etiqueta `<link>`

Si se utiliza la etiqueta `<link>` para enlazar los archivos CSS externos, se puede utilizar el atributo `media` para indicar el medio o medios en los que se aplican los estilos de cada archivo:

```
<link media="screen" href="basico.css" >  
<link media="print, handheld" href="especial.css" >
```

En este ejemplo, el primer archivo CSS se tiene en cuenta cuando la página se visualiza en la pantalla (`media="screen"`). Los estilos indicados en el segundo archivo CSS, se aplican al imprimir la página (`media="print"`) o al visualizarla en un dispositivo móvil (`media="handheld"`), como por ejemplo en un iPhone.

Si la etiqueta `<link>` no indica el medio CSS, se sobreentiende que los estilos se deben aplicar a todos los medios, por lo que es equivalente a indicar `media="all"`.