

Tema 5

Hojas de estilo CSS (I)

Objetivos

- Identificar las ventajas que aporta la utilización de las hojas de estilo para especificar la apariencia de la presentación de los documentos HTML.
- Reconocer las ventajas que aporta la separación entre la descripción de contenidos y la especificación de la apariencia de su presentación.
- Conocer las diversas formas existentes para aplicar estilo en documentos HTML, así como la prioridad existente entre ellas.
- Aplicar las características de las hojas de estilo en los documentos HTML.

Introducción

En los temas anteriores se estudiaron los elementos más utilizados a la hora de estructurar los contenidos en un documento HTML, así como sus etiquetas y atributos. En este tema se aborda el estudio de las hojas de estilo en cascada que permiten controlar la apariencia de la presentación de los distintos elementos de un documento HTML. El estudio de las hojas de estilo en cascada se ha dividido en varios temas debido a su amplitud, facilitándose de este modo su aprendizaje de una forma más detallada y ordenada. Estos temas se han planificado de forma consecutiva dentro de la periodificación de contenidos del módulo profesional.

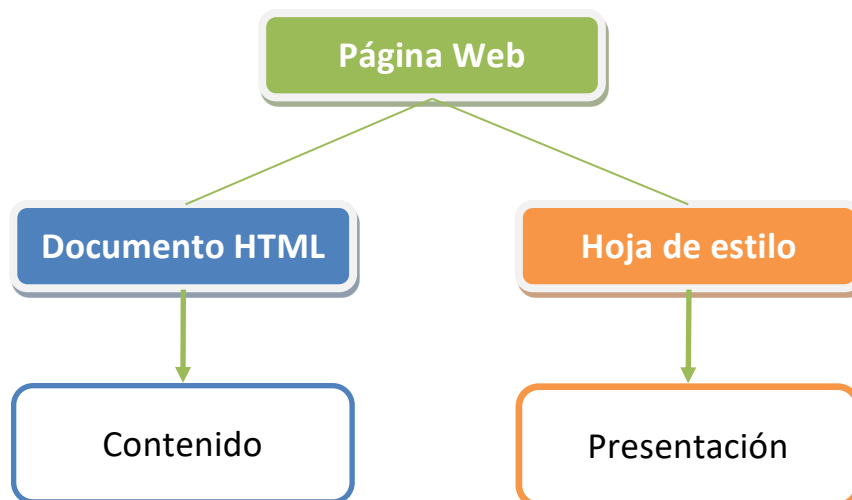
Índice

5.1 ¿QUÉ SON LAS HOJAS DE ESTILO?	2
5.2 FORMAS DE INTEGRAR ESTILO EN UN DOCUMENTO HTML	5
5.3 REGLAS DE ESTILO	10
5.4 SELECTORES	14
5.5 HERENCIA Y PRECEDENCIA	22
5.6 EJEMPLOS DE APLICACIÓN DE LAS HOJAS DE ESTILO	23
Glosario de términos	24

5.1 ¿QUÉ SON LAS HOJAS DE ESTILO?

El uso de las hojas de estilo es un mecanismo que permite tener un control exhaustivo sobre la presentación de un documento HTML. Se utilizan para especificar la apariencia visual que tendrán los distintos elementos que conforman una página Web. Así, mediante las hojas de estilo es posible definir todos los aspectos que intervienen en la apariencia visual de la presentación, como pueden ser: los colores del texto, el estilo y tamaño de las fuentes del texto, el posicionamiento de los elementos, la distancia entre elementos, el interlineado, etc. El aspecto visual de la presentación de las páginas Web tiene su importancia, puesto que el servicio de información Web es, por definición, un medio de información multimedia. En este sentido, debe tenerse en cuenta que la preocupación por obtener una composición y apariencia visual adecuada, intuitiva y agradable de la presentación de contenidos en las páginas Web, es un aspecto que puede facilitar la consulta e interacción por parte de los potenciales usuarios.

La aplicación de las hojas de estilo facilita realizar una separación exhaustiva entre la descripción de los contenidos y su presentación en las páginas Web. Tal como se ha estudiado en los temas anteriores, las etiquetas y atributos del lenguaje HTML permiten estructurar los contenidos en las páginas Web. Mientras que las hojas de estilo permiten aplicar las reglas de estilo a estos elementos para controlar la apariencia visual de su presentación.



En realidad, existen diversos lenguajes de hojas de estilo, es decir, diversas formas de codificar las propiedades o características de estilo aplicable a las etiquetas de HTML. El lenguaje HTML no está asociado a ningún lenguaje de hojas de estilo determinado. El lenguaje de hojas de estilo más extendido es el lenguaje de hojas de estilo denominado CSS. El término CSS es el acrónimo de *Cascading Style Sheet* en inglés, que se traduce como Hojas de Estilo en Cascada. El lenguaje de hojas de estilo CSS es adecuado para separar los contenidos y su presentación y, además, es la forma más extendida de proporcionar diseño visual a las páginas y sitios Web. En la actualidad, conocer profundamente el lenguaje CSS es imprescindible para abordar con garantías la construcción de páginas y sitio Web agradables visualmente.

La denominación “en cascada” del término CSS hace referencia a que, a la hora de crear un estilo visual, las especificaciones realizadas a un determinado elemento son aplicables también a todos los elementos que se encuentren incluidos en él, atendiendo a un criterio de herencia. Por ejemplo, si se especifica un tipo de letra para el elemento <body>, entonces todos los elementos de la página heredarán esta característica de estilo, de manera que se presentarán en la ventana de visualización del navegador con el mismo tipo de fuente. Así, un elemento <p>, que estará incluido el cuerpo del documento, heredará el tipo de letra especificado, salvo que se especifique uno específico para ese párrafo en concreto.

A continuación, en el siguiente ejemplo, puede apreciarse el aspecto del código de una página Web que utiliza las hojas de estilo en cascada para especificar las características de apariencia visual de la presentación. Puede apreciarse que se utiliza el lenguaje de marcas HTML para estructurar el contenido de la página y el lenguaje de hojas de estilo CSS para especificar la apariencia visual de la presentación de cada uno de los elementos de la página. Las características de estilo de la presentación visual de los distintos elementos de la página Web se han especificado, en este caso, dentro de la etiqueta <style> en la cabecera.

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Ejemplo 1</title>
  <meta charset="utf-8" />
  <style>
    body
    {
      background-color: #d0e4fe;
      font-family: Arial, Helvetica, sans-serif;
    }
    h1
    {
      color: orange;
      text-align: center;
    }
    p
    {
      font-family: "Times New Roman";
      font-size: 20px;
    }
  </style>
</head>
<body>
  <h1>Primer ejemplo CSS</h1>
  <p>
    Este es el primer ejemplo de CSS. Las hojas de estilo en cascada se utilizan para
    especificar las características de la presentación. Puede apreciarse que se utiliza
    el lenguaje HTML para estructurar el contenido del documento y el lenguaje CSS para
    especificar la apariencia visual de la presentación de cada uno de los elementos
    del documento.
  </p>
</body>
</html>
```

El ejemplo anterior, produce el siguiente resultado en la ventana de visualización del navegador:

Ejemplo CSS

Este es el primer ejemplo de CSS. Las hojas de estilo en cascada se utilizan para especificar las características de la presentación. Puede apreciarse que se utiliza el lenguaje HTML para estructurar el contenido del documento y el lenguaje CSS para especificar la apariencia visual de la presentación de cada uno de los elementos del documento.

Ventajas del uso de hojas de estilo

La separación entre los contenidos y sus características de presentación presenta numerosas ventajas para la construcción de páginas Web. Esta característica esencial de las hojas de estilo en cascada es el origen de las principales ventajas de la utilización de las hojas de estilo en cascada, entre las que se encuentran las siguientes:

- Proporciona una separación exhaustiva entre la descripción de los contenidos y su presentación visual en las páginas y sitios Web.
- Facilita la escritura del código y su mantenimiento.
- Mejora la legibilidad del código.
- Facilita la modificación de la apariencia visual de las páginas y sitios Web.
- Mejora la Accesibilidad Web.
- Obliga a crear documentos HTML/XHTML bien definidos y estructurados.
- Facilita el desarrollo del diseño Web adaptativo (*responsive*) que permite presentar el mismo documento en diversos tipos de dispositivos diferentes.

Cuando se utilizan las hojas de estilo pueden realizarse modificaciones en las propiedades visuales de los distintos elementos y, de esta manera, se puede cambiar completamente la apariencia visual de la página Web, o de los sitios Web. Un ejemplo de las posibilidades que ofrece su uso se puede encontrar visitando el sitio Web Zengarden (<http://www.csszengarden.com/>), donde se puede ver el efecto que produce utilizar hojas de estilo distintas sobre una misma página Web.

Evolución histórica y versiones de CSS

Las hojas de estilo aparecieron poco después que el lenguaje de etiquetas SGML, debido a la necesidad de definir un mecanismo que permitiera aplicar estilo a los documentos electrónicos. El gran impulso de los lenguajes de hojas de estilo se produjo con la expansión Internet y el crecimiento del lenguaje HTML para la construcción de páginas Web. El organismo W3C (*World Wide Web Consortium*), que se encarga de definir los estándares Web, propuso el lenguaje de hojas de estilo en cascada CSS (*Cascading Style Sheets*) a finales de 1994. A finales de 1996, el organismo W3C publicó la primera recomendación oficial, conocida como CSS de nivel 1 o CSS1. En 1998, se publica su segunda recomendación oficial, conocida como **CSS de nivel 2 o CSS2**. Posteriormente, se ha actualizado esta recomendación, dando lugar a la revisión CSS 2.1. que es la versión de CSS que se suele utilizar. Actualmente, además, continua en fase de desarrollo la recomendación conocida como **CSS de nivel 3 o CSS3** que es compatible con las versiones anteriores de CSS. A pesar de estar en desarrollo, las versiones actuales de los navegadores soportan la mayoría de sus características.

5.2 FORMAS DE INTEGRAR ESTILO EN UN DOCUMENTO HTML

las hojas de estilo permiten especificar la apariencia de la presentación visual de los elementos que componen un documento HTML. Por lo tanto, de alguna manera, debe existir una relación entre cada uno de los elementos definidos en el documento HTML y su correspondiente especificación de estilo. Existen tres formas distintas de asociar la especificación de estilo a un documento HTML:

1. **Estilo en línea.** Consiste en incluir el código CSS en la misma línea de código HTML que define un elemento de la página, utilizando para ello el atributo style. De manera que el estilo CSS se aplica directamente a cada elemento HTML. El atributo style puede aplicarse a cualquier etiqueta de HTML. Por ejemplo:

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Hojas de estilo en línea</title>
  <meta charset="utf-8" />
</head>
<body style="font-family: Arial, Helvetica, sans-serif;">
  <h1 style="background-color: rgb(0,0,0); color: rgb(192,192,192);
    text-align:center;">Ramón y Cajal</h1>
  <h2 style="color:#0000ff; text-align:center; ">(1852-1934)</h2>
  <p style="text-align:justify">
    Santiago Ramón y Cajal fue un médico español, especializado en histología y
    anatomía patológica. Compartió el premio Nobel de Medicina en 1906 con Camillo
    Golgi en reconocimiento de su trabajo sobre la estructura del sistema nervioso por
    sus investigaciones sobre los mecanismos que gobiernan la morfología y los procesos
    conectivos de las células nerviosas, una nueva y revolucionaria teoría que empezó a
    ser llamada la "Doctrina de la neurona", basada en que el tejido cerebral está
    compuesto por células individuales. Humanista, además de científico, está
    considerado como cabeza de la llamada Generación de Sabios.
  </p>
</body>
</html>
```

2. **Hojas de estilo internas.** Consiste en incluir el código CSS en la cabecera del documento HTML, utilizando para ello la etiqueta de cabecera <style>. De manera que el estilo CSS se aplica al documento HTML en el cual se integra. Por ejemplo, para aplicar estilo CSS al cuerpo del documento y todos los párrafos de esta página, se haría:

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Hojas de estilo internas</title>
  <meta charset="utf-8" />
  <style>
    body {
      font-family: Arial, Helvetica, sans-serif;
      margin: 50px;
    }
    h1 {
      background-color: rgb(0,0,0);
      color: rgb(192,192,192);
      text-align: center;
    }
    h2{
      color: #0000ff;
      text-align: center;
    }
    p {
      text-align: justify;
    }
  </style>
</head>
<body>
  <h1>Leonardo da Vinci</h1>
  <h2>(1452-1519)</h2>
  <p>
    Fue un polímata florentino del Renacimiento italiano, a la vez pintor, anatomista,
    arquitecto, paleontólogo, artista, botánico, científico, escritor, escultor,
    filósofo, ingeniero, inventor, músico, poeta y urbanista. Leonardo estudió con el
    célebre pintor florentino Andrea de Verrocchio. Sus primeros trabajos de
    importancia fueron creados en Milán al servicio del duque Ludovico Sforza. Trabajó,
    a continuación, en Roma, Bolonia y Venecia, y pasó los últimos años de su vida en
    Francia, por invitación del rey Francisco I.
  </p>
  <p>
    Es, frecuentemente, descrito como un arquetipo y símbolo del hombre del
    Renacimiento, genio universal, además de filósofo humanista cuya curiosidad
    infinita solo puede ser equiparable a su capacidad inventiva. Leonardo da Vinci es
    considerado como uno de los más grandes pintores de todos los tiempos y,
    probablemente, es la persona con el mayor número de talentos en múltiples
    disciplinas que jamás ha existido. Como ingeniero e inventor, Leonardo desarrolló
    ideas muy adelantadas a su tiempo, tales como el helicóptero, el submarino y el
    automóvil. Muy pocos de sus proyectos llegaron a construirse, puesto que la mayoría
    no eran realizables durante esa época. Como científico, Leonardo da Vinci hizo
    progresar mucho el conocimiento en las áreas de anatomía, la ingeniería civil, la
    óptica y la hidrodinámica.
  </p>
</body>
</html>
```

3. **Hojas de estilo externas.** Consiste en crear un archivo externo que incluye el código CSS y vincularlo con uno o varios documentos HTML, utilizando para ello la etiqueta de cabecera `<link>`. De manera que un mismo archivo de estilo CSS puede ser vinculado con varios archivos HTML y, por tanto, el estilo se aplicaría, directamente, a estos documentos HTML, reutilizando, por tanto, el código. Cualquier cambio que se realice sobre en las especificaciones de estilo del archivo externo afectaría inmediatamente a todos los documentos HTML con los que el archivo de estilo externo está vinculado. Se utiliza para especificar el estilo que afecta a todo un sitio Web. El archivo externo que especifica el estilo es un archivo de texto plano que suele tener la extensión `.css`.

En el siguiente ejemplo, se utiliza una hoja de estilo externa CSS para especificar el estilo de un documento HTML. En el ejemplo, puede apreciarse el uso que se hace de la etiqueta de cabecera `<link>` para enlazar la especificación de estilo, que se realiza en el archivo *Estilo.css*, con el documento HTML, que se almacena en el archivo *Ejemplo.html*.

Nombre del archivo: *Ejemplo.html*

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Hojas de estilo externas</title>
  <meta charset="utf-8" />
  <link rel="stylesheet" type="text/css" href="./Estilo.css">
</head>
<body>
  <h1>Juan de la Cierva</h1>
  <h2>(1895-1936)</h2>
  <p>
    Juan de la Cierva y Codorníu fue un inventor y científico aeronáutico español,
    ingeniero de caminos, canales y puertos y aviador. Inventó el autogiro, aparato
    precursor del actual helicóptero. Desde su infancia destacó su interés por el mundo
    de la aviación, y junto a su amigo Tomás de Martín-Barbadillo construyó pequeños
    modelos capaces de volar.
  </p>
  <p>
    Juan de la Cierva construyó en Madrid en 1920 su primer autogiro, el Cierva C.1,
    utilizando fuselaje, ruedas y estabilizador vertical. La planta motriz era un motor
    Le Rhône de 60 HP. El aparato no llegó a volar, pues el rotor inferior giraba a
    menos velocidad de la prevista, y el efecto giroscópico y la asimetría de la
    sustentación hicieron volcar el aparato. A este primer autogiro siguieron dos
    construcciones también fallidas, el C.2 y el C.3, en las que el inventor intentó,
    infructuosamente, resolver el problema de la diferencia de sustentación entre la
    pala que avanza y la que retrocede. Sin embargo, en las pruebas se consiguieron
    algunos saltos de unos dos metros, lo que apuntaba a la viabilidad del invento. La
    asimetría de la sustentación del rotor no se resolvería plenamente hasta el
    prototipo C.4, en el que la Cierva incluyó su revolucionaria idea de articular las
    palas del rotor en su raíz.
  </p>
</body>
</html>
```

Nombre del archivo: *Estilo.css*

```
body {  
    font-family: Arial, Helvetica, sans-serif;  
    margin: 50px;  
    background-color: aliceblue ;  
}  
  
h1 {  
    background-color: rgb(0,0,0);  
    color: rgb(192,192,192);  
    text-align: center;  
}  
  
h2 {  
    color: #0000ff;  
    text-align: center;  
}  
  
p {  
    text-align: justify;  
}
```

Orden de prioridad

Como se acaba de ver, hay varias formas de asociar la especificación de estilo. Así, la especificación de estilo que afecta a los elementos de un documento puede estar distribuida en una hoja de estilo externa, o en una hoja de estilo interna, o en línea mediante el atributo style. Pero entonces, ¿Qué estilo se usará cuando se ha especificado más de un estilo para un mismo elemento de HTML? ¿Qué ocurre si además los estilos a aplicar son contradictorios? Las respuestas a estas cuestiones hay que buscarlas, precisamente, en el comportamiento en cascada de las hojas de estilo. Este comportamiento establece un criterio de prioridad al aplicar los estilos en caso de colisión. Los estilos se aplican en cascada según el siguiente orden de prioridad, de mayor a menor:

1. Estilo en línea.
2. Hoja de estilos interna.
3. Hoja de estilos externa.

En general, siempre tiene mayor prioridad en la aplicación del estilo, aquella especificación de estilo que se encuentre definida de manera más cercana al elemento HTML correspondiente. Si el link de una hoja de estilo externa se especifica después de la hoja de estilo interna en la cabecera, la hoja de estilo externa anulará la interna. Por ejemplo:

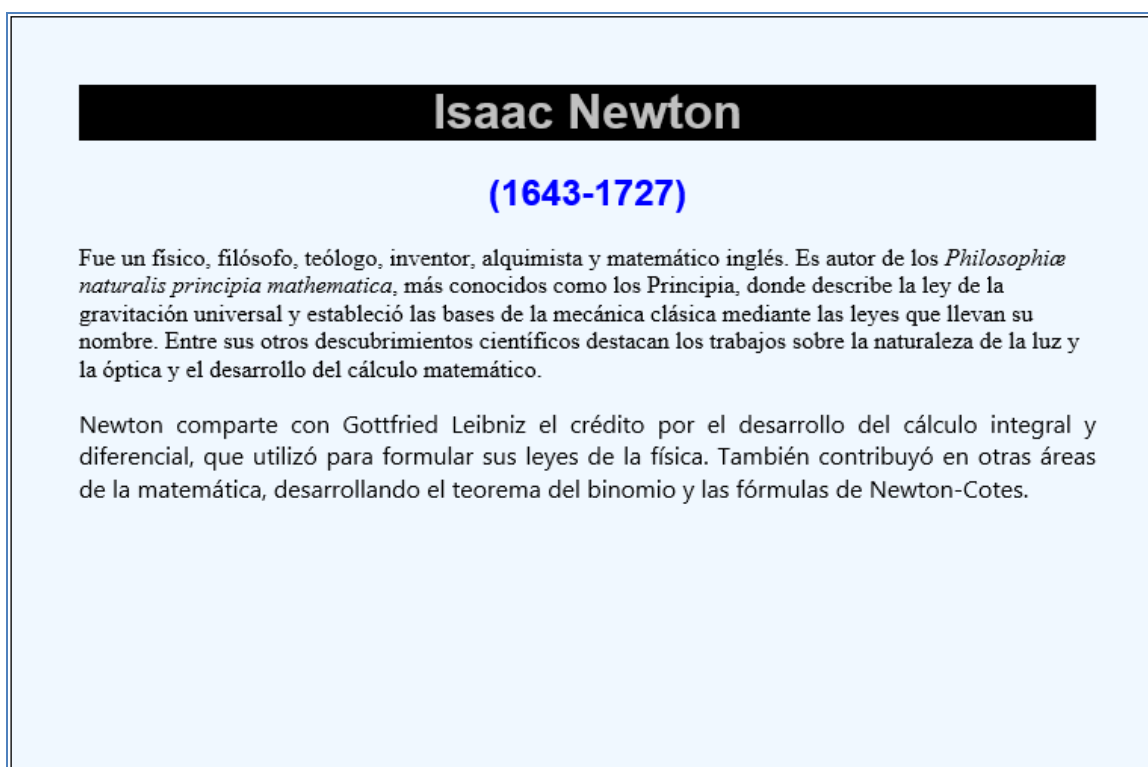
```
<!DOCTYPE html>  
<html>  
<head>  
    <title>CSS - Hojas de estilo externas</title>  
    <meta charset="utf-8" />  
    <link rel="stylesheet" type="text/css" href="../Estilo.css">  
    <style>  
        p {  
            font-family: 'Times New Roman';  
        }  
    </style>  
</head>  
</html>
```



```
</style>
</head>
<body>
  <h1>Isaac Newton</h1>
  <h2>(1643-1727)</h2>
  <p style="text-align: left">
    Fue un físico, filósofo, teólogo, inventor, alquimista y matemático inglés. Es
    autor de los <i>Philosophiæ naturalis principia mathematica</i>, más conocidos
    como los Principia, donde describe la ley de la gravitación universal y estableció
    las bases de la mecánica clásica mediante las leyes que llevan su nombre. Entre sus
    otros descubrimientos científicos destacan los trabajos sobre la naturaleza de la
    luz y la óptica y el desarrollo del cálculo matemático.
  </p>
  <p style="font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif">
    Newton comparte con Gottfried Leibniz el crédito por el desarrollo del cálculo
    integral y diferencial, que utilizó para formular sus leyes de la física. También
    contribuyó en otras áreas de la matemática, desarrollando el teorema del binomio y
    las fórmulas de Newton-Cotes.
  </p>
</body>
</html>
```

El ejemplo anterior, está vinculado con el archivo de hoja de estilo externa *Estilo.css* utilizado anteriormente. Puede apreciarse que el texto del primer párrafo se alineará a la izquierda. Además, el primer párrafo se escribirá con una fuente de tipo *Times New Roman* y el segundo párrafo con *Segoe UI*. Todo ello es debido al orden de prioridad en cascada existente en CSS.

El ejemplo anterior, produce el siguiente resultado en la ventana de visualización del navegador:



Hojas de estilo importadas

Otra forma de usar las hojas de estilo externas es emplear el comando `@import`. Este comando es una propiedad de estilo de CSS2, mientras que la etiqueta `<link>` lo es de HTML. La ventaja consiste en que podrá usarse no sólo para invocar una hoja de estilo externa en un documento HTML, sino también para importar otra hoja de estilo en una hoja de estilo externa. En el siguiente ejemplo, se utiliza una hoja de estilo externa CSS para especificar el estilo de un documento HTML. En el ejemplo, puede apreciarse el uso que se hace del comando de CSS `@import` para enlazar la especificación de estilo externa, que se realiza en el archivo *Estilo1.css*, con el documento HTML, que se almacena en el archivo *Ejemplo1.html*.

Nombre del archivo: *Estilo1.css*

```
h1 {  
    text-decoration: underline;  
}
```

Nombre del archivo: *Ejemplo1.html*

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>CSS - Hojas de estilo externas mediante @import</title>  
    <meta charset="utf-8" />  
    <style type="text/css">  
        @import url(Estilo1.css);  
    </style>  
</head>  
<body>  
    <h1>Título de nivel 1</h1>  
</body>  
</html>
```

5.3 REGLAS DE ESTILO

Tal y como se ha estudiado en el apartado anterior, existen tres formas distintas de utilizar hojas de estilo en cascada CSS. La primera de ellas, estilo en línea, se utiliza para aplicar estilo particular a unos elementos concretos de la página, por lo que suele utilizarse en casos puntuales. Por ello, el estudio de las hojas de estilo se va a centrar en la aplicación de las hojas de estilo internas y externas, porque son las formas de aplicar estilo más generales y utilizadas y, también, porque permiten expresar toda su funcionalidad. La especificación de las hojas de estilo internas y externas se realiza de la misma manera, la única diferencia reside en el lugar donde se realiza la especificación: en la cabecera del mismo documento HTML, en el caso de las hojas de estilo internas; y, en un archivo aparte, en el caso de las hojas de estilo externas.

Sintaxis de las reglas de estilo CSS

La especificación del estilo de un documento HTML consiste en definir un conjunto de reglas de estilo CSS que se aplicarán a cada uno de los elementos que forman el documento HTML. Como se

puede observar en la siguiente ilustración, las reglas de estilo CSS están formadas por dos componentes: un **selector**, que indica cuáles serán los elementos a los que se aplica el estilo, y una **declaración** entre llaves, que establece las propiedades y valores que se aplicarán. A su vez, cada declaración se compone de **propiedad** y **valor**. La estructura genérica de una regla de estilo es:



Así, en el ejemplo del recuadro anterior, el selector es h1 y tiene especificadas dos declaraciones de estilo encerradas entre llaves y separadas por un carácter de punto y coma. La primera declaración especifica la propiedad color para asignarle el valor blue. Y, la segunda declaración, especifica que el valor de la propiedad font-size es 12px. De esta manera se ha definido que todos los elementos <h1> que aparezcan en la página se presentarán visualmente sobre la ventana de visualización del navegador con un color del texto en azul y con un tamaño de letra de 12 píxeles.

En el ejemplo que se presenta a continuación, se han definido tres reglas de estilo: la primera se aplica sobre el elemento body, que es la etiqueta que define el cuerpo de una página; la segunda se aplica sobre el elemento p; y, la tercera se aplica sobre los elementos h1 y h2. La primera regla tiene definidas cinco declaraciones de formato mediante pares del tipo *propiedad: valor*, la segunda tiene una declaración de estilo y la tercera define, de forma agrupada, las propiedades de estilo que tendrán los dos elementos que aparecen en el selector.

```
body {  
    background: white;  
    font-size: 11pt;  
    font-family: Arial,Helvetica,sans-serif;  
    color: black;  
    text-align:justify;  
}  
p {  
    text-indent: 2em;  
}  
h1, h2 {  
    font-family: Verdana;  
    color: #ff0000;  
}
```

La sintaxis de la construcción de las reglas de estilo CSS es, por tanto:

Selector { propiedad1: valor1; propiedad2: valor2; . . . ; propiedadN: valorN }

El selector puede ser cualquier elemento HTML. Las propiedades y los valores asignados a cada una de ellas determinan la apariencia de la presentación visual del elemento HTML seleccionado, a través del selector al que afecta.

Propiedades y valores

Las propiedades son las características de estilo que se aplican a los selectores para modificar la apariencia de su presentación en la página. Estas propiedades de estilo forman parte de la especificación CSS. Es importante tener presente que los valores permitidos dependen de cada propiedad en concreto. Las propiedades que se pueden aplicar a los elementos que forman un documento HTML, así como los tipos de valores que pueden tomar estas propiedades pueden consultarse en la Guía rápida CSS 2.1 del organismo W3C:

- **W3C España.** <http://www.w3c.es/Divulgacion/GuiasReferencia/CSS21/>

Además, pueden consultarse multitud de direcciones Web en las que se presentan las propiedades y los valores disponibles para especificar las reglas de estilo CSS, como pueden ser:

- **LibrosWeb.** <http://librosweb.es/referencia/css/>
- **w3Schools.** <http://www.w3schools.com/cssref/default.asp>
- **Sidar.** <http://www.sidar.org/recur/desdi/traduc/es/css/propidx.html>
- **HtmlHelp.** <http://htmlhelp.com/es/reference/css/properties.html>
- **DesarrolloWeb.** <http://www.desarrolloweb.com/articulos/186.php>

Los valores que se pueden asignar a las propiedades de estilo dependen de las características concretas del elemento al que se aplican y pueden ser de distintos tipos: valores enteros y reales, valores de cadena de tipo URI, valores numéricos de tipo color RGB en hexadecimal, valores de tipo cadena que se expresan mediante términos reservados, etc. En las guías rápidas que se han indicado anteriormente pueden apreciarse los tipos de valores permitidos para cada propiedad.

Algunos valores que se aplican a ciertas propiedades son unidades de medida, también denominadas **unidades CSS**. Por ejemplo, el valor del tamaño del ancho de un elemento, la longitud de un margen, el tamaño de la fuente, etc. Las unidades de medida CSS se pueden clasificar en dos grupos: las unidades relativas y las unidades absolutas.

- **Relativas.** Se llaman así porque son unidades que se especifican de forma relativa al medio o soporte sobre el que se presenta la página Web. Actualmente, los usuarios pueden utilizar diversos dispositivos para acceder a la Web: ordenadores, tablets, smartphones, etc. Se recomienda el uso de unidades relativas, porque se ajustarán mejor al medio con el que el usuario está accediendo a la página. Las unidades relativas son las siguientes:
 - **Fuente actual: em.** La unidad em es relativa a la fuente actual con la que se está trabajando por defecto en el sistema del usuario. Por ejemplo, si un usuario tiene configurada el tamaño de la fuente por defecto en 12 puntos, entonces el valor *1em* será igual a 12 puntos y el valor *2em* será igual a 24 puntos.

- Altura de la letra x: ex. Esta unidad se refiere a la altura de la letra x. Así, el valor *1ex* será igual a la altura de la letra x, según la fuente actual del usuario.
- Píxeles: px. Un píxel es un punto en la pantalla del dispositivo. Dependiendo de la resolución de la pantalla, un píxel puede ser de mayor o menor tamaño.

```
p { font-size: 12px; }
```

- Porcentaje. El valor en porcentaje es siempre relativo a otro. El valor al que se refiere el porcentaje depende de cada propiedad en concreto y puede ser: el valor de otra propiedad para el mismo elemento, el valor de una propiedad del elemento padre, un valor dependiente del contexto, etc. Por ejemplo, si el valor del tamaño de la fuente es de 12pt y se especifica la propiedad correspondiente con un valor de 150%, entonces el tamaño de la fuente sería igual al 150% de los 12pt actuales, que equivale a 18pt. El uso de unidades en porcentaje está ampliamente aceptado y es ampliamente utilizado.

```
h2 { width: 80%; text-align: center; }
```

- Absolutas. Las unidades absolutas son medidas fijas que deberían verse igual en todos los dispositivos, como, por ejemplo, los centímetros que son una convención de medida internacional. Aunque, en principio, pueden parecer más útiles, puesto que se verían en todos los dispositivos igual, tienen el problema de adaptarse peor a las particularidades de cada dispositivo. Puede que 1 centímetro (cm) sea una medida razonable en una pantalla de un ordenador de sobremesa, sin embargo, podría ser una longitud exageradamente grande para la pantalla de un smartphone. Se recomienda utilizar, por tanto, siempre unidades de medida relativas. Las unidades absolutas son:
 - Puntos: pt. Un punto es 1/72 pulgadas.
 - Pulgadas: in.
 - Centímetros: cm.
 - Milímetros: mm.
 - Picas: pc. Una pica son 12 puntos.

```
h1 { margin: 0.5in; word-spacing: 4mm; }
```

Otro tipo de valores que se asignan a las propiedades de estilo son los **valores de color** que se expresan con valores RGB, empleando para ello seis o tres números hexadecimales para especificar un determinado color. Por ejemplo, el valor #fff equivaldría a #ffffff, y el valor #123 equivaldría a #112233. Además, los colores se pueden especificar también mediante valores RGB decimales con la notación *rgb(r,g,b)*, siendo los valores de r, g, b números decimales entre 0 y 255. Otra notación posible es *rgb(r%,g%,b%)*, siendo cada uno de los valores r%, g%, b% un valor entre 0 y 100.

```
h1 { background-color: #c3def5; color: rgb(192,192,192); }
```

También pueden utilizarse **valores de tipo URI** en las hojas de estilo en cascada. Estos valores sirven para especificar rutas hacia elementos externos o recursos de la página como pueden ser imágenes. Los valores de tipo URI en CSS se especifican con la notación `url(valor)`, siendo *valor* la dirección URL deseada, que puede ser absoluta o relativa. Si es relativa, se interpreta considerando el archivo donde se especifica la propiedad correspondiente.

```
body { background:url("../images/otraimagen.jpg"); }
```

Se han estudiado algunos de los tipos de valores más comunes que suelen formar parte de las declaraciones de estilo. Sin embargo, pueden utilizarse otros tipos de valores. Puede obtenerse una referencia más completa, consultando la especificación de cada propiedad de estilo.

Comentarios en las hojas de estilo CSS

Una hoja de estilo externa puede contener comentarios que pueden extenderse varias líneas. Los delimitadores de comentario son los mismos que se utilizan con lenguaje C, es decir: `/* ... */`.

```
/*  
  Autor: Vicente  
  Última fecha de modificación: 3 de septiembre de 2016  
*/  
h1 {  
  color : white ;           /* texto blanco */  
  background-color : black ; /* sobre fondo negro */  
}
```

5.4 SELECTORES

Tal como se ha estudiado en el punto anterior, el selector de una regla de estilo indica cuáles serán los elementos a los que se aplicará el estilo especificado en la declaración de estilo. De modo que, las propiedades de estilo especificadas, y sus correspondientes valores, se refieren a los elementos del documento HTML a través del selector. Por ejemplo, obsérvese la siguiente regla de estilo:

```
p {  
  color: red;  
}
```

En el ejemplo, se aplicaría el color de texto rojo a los elementos indicados por el selector. En este caso, el selector hace referencia a todos los párrafos del documento HTML. Sin embargo, ¿Sería posible actuar solamente sobre unos párrafos determinados y dejar el resto con otro color del texto? ¿Sería posible aplicar una misma declaración de estilo a distintos tipos de elementos? La respuesta es, obviamente, sí. Para ello, existen distintas formas de especificar el selector de una regla de estilo para poder referirse a unos elementos concretos del documento HTML, según el criterio de selección que se desea aplicar. A continuación, se estudian las distintas posibilidades existentes a la hora de definir el selector de una regla de estilo:

Selectores de etiqueta

La sintaxis de un selector de etiqueta consiste en el nombre de la etiqueta seguido de la declaración de estilo entre llaves. En estos casos, la declaración de estilo se aplica a todos los elementos del documento HTML que se construyen con la etiqueta especificada. Por ejemplo:

```
div { background-color: red; }
```

En este caso, se aplica fondo de color rojo a todas las divisiones del documento. Observe que se indica únicamente el nombre de la etiqueta, sin los signos menor que (<) y mayor que (>).

También es posible aplicar el mismo efecto de estilo a distintos selectores de etiqueta. En este caso, cada una de las etiquetas implicadas se escribe separada con una coma. Por ejemplo:

```
h1, h2, h3 { color: #ff6a00; }
```

Lo que significa que todos los elementos <h1>, <h2> y <h3> de la página tendrán el mismo color del texto, que será el especificado en la declaración de la regla de estilo, es decir, #ff6a00.

Selectores de clase

Los selectores de etiqueta permiten asociar estilo a una etiqueta HTML. Para poder aplicar un mismo estilo a varios elementos del documento HTML, pueden definirse clases de estilo. De este modo, el desarrollador podrá definir sus propios selectores. La definición de una clase de estilo para un tipo de elemento concreto, que esta especificado mediante su etiqueta, se realiza de la forma siguiente:

```
etiqueta.nombre_de_la_clase { declaración(es) de estilo }
```

Como puede apreciarse se incluye un punto entre la etiqueta y el nombre que queremos atribuir a la clase y, a continuación, se añade la declaración de estilo escrita entre llaves. Por ejemplo:

```
p.rojo { color: red; }
```

La definición de clase de estilo anterior puede aplicarse a cualquier párrafo del documento HTML, a través del atributo class de HTML:

```
<p class="rojo">Este es un párrafo cuyo texto se escribe con color rojo</p>
```

Todos los párrafos del documento al que se le aplique la clase de estilo rojo se presentarán en la ventana de visualización del navegador según la declaración especificada en esa clase. No obstante, la clase así definida no tiene ningún efecto si no se invoca en el documento HTML mediante el atributo class que está definido en todas las etiquetas HTML.

También pueden definirse las denominadas clases universales que aportan mayor funcionalidad. Una **clase de estilo universal** puede ser aplicada a cualquier elemento de la página. Por ejemplo:

```
*.grande { Font-size: 50px }
```


Esta clase puede aplicarse a diversos tipos de elementos en el documento HTML, por ejemplo:

```
<p class="grande">Texto 1</p>
<div class="grande">Texto 2</div>
<h1 class="grande">Texto 3</h1>
```

Un nombre de clase de estilo puede estar formado por letras, cifras, el guion y el carácter de subrayado. El primer carácter no puede ser un número, un guion o un carácter de subrayado. Hay que evitar los espacios en blanco, así como las palabras reservadas.

Habitualmente, se suele omitir el carácter de selector universal (*) al definir una clase universal, de manera que la anterior definición de clase de estilo universal se escribiría de la siguiente forma:

```
.grande { font-size: 50px }
```

En el siguiente ejemplo, puede apreciarse el uso que se hace de las clases de estilo, juntamente con las etiquetas <div> y . Las clases de estilo definidas se aplican a los elementos <div> y del documento HTML para formatear la apariencia visual de la presentación.

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Clases</title>
  <meta charset="utf-8" />
  <style>
    span.rojo {
      color: red;
    }
    .tipoletra {
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    }
    .negrita {
      font-weight: bold;
    }
    .cursiva {
      font-style: italic;
    }
    p.centrado {
      text-align: center;
    }
  </style>
</head>
<body class="tipoletra">
  <h1>Ejemplo de uso de clases</h1>
  <p class="centrado">
    Quien pronto se acuesta y pronto se levanta,
    es persona <span class="negrita">saludable</span>,
    <span class="cursiva">rica</span>
    y <span class="rojo">sabia</span>.
  </p>
</body>
</html>
```

El ejemplo anterior, produce el siguiente resultado en la ventana de visualización del navegador:

Ejemplo de uso de clases

Quien pronto se acuesta y pronto se levanta, es persona **saludable**, *rica* y **sabia**.

Como se ha estudiado anteriormente, una clase de estilo hace referencia a un conjunto de elementos. Pero, además, un elemento puede pertenecer a varias clases. En el ejemplo siguiente se puede apreciar cómo se define un elemento al que se le aplican varias clases de estilo. Así, por ejemplo, considerando la siguiente declaración de estilo:

```
.muebles {  
  border-color: red;  
}  
.habitacion {  
  border-width: 2px;  
  border-style: dashed;  
}
```

Para hacer que un elemento pertenezca a varias clases, éstas se separan mediante espacios:

```

```

De esta manera, se crearía un borde rojo en todos los elementos que pertenecieran a la clase muebles y se crearía un borde de línea discontinua de 2 píxeles en todos los elementos de la clase habitacion. La imagen cama.jpg, al pertenecer a las dos clases, tendría un borde discontinuo rojo de 2 píxeles de grosor.

Selectores de identificador

Todas las etiquetas de HTML pueden incluir un atributo llamado identificador: el **atributo id**. Este atributo tiene como fin asignar un nombre único a un elemento dentro de la página Web. De esta manera, un elemento queda identificado y, por tanto, diferenciado de todos los demás, ya que dentro de una página no puede haber más de un elemento con el mismo identificador. Además, un elemento sólo puede tener un único identificador. El selector id permite, por tanto, identificar un elemento único en la página. En el ejemplo siguiente se puede observar cómo se identifica un párrafo concreto de la página mediante el uso de un identificador:

```
<p id="parrafo1">
```

Pues bien, una vez que se ha asignado un identificador a un elemento en un documento HTML, se puede aplicar estilo a ese elemento a través de su identificador. En este caso, puesto que se trata de un párrafo, la regla de estilo quedaría de la siguiente manera:

```
p#parrafo1 { font-size: 14pt; }
```

Aunque, lo más habitual es omitir el nombre de la etiqueta, por lo que quedaría escrito mejor así:

```
#parrafo1 { font-size: 14pt; }
```

Como puede observarse, se utiliza el símbolo almohadilla (#) y el nombre del identificador que haya sido asignado al elemento concreto al que se le desea aplicar estilo, seguido de la declaración de estilo entre llaves.

El selector id permite aplicar una hoja de estilo de forma similar a como lo hace el selector class, aunque con la diferencia de que no puede emplearse más de una vez en un documento. Es posible mezclar declaraciones de estilo class e id.

El nombre de un selector identificador puede contener letras, números, el guion y el carácter de subrayado. El primer carácter no puede ser un número, un guion o un carácter de subrayado. Es conveniente evitar los espacios en blanco, así como las palabras reservadas.

Pseudo-clases

Las pseudo-clases se utilizan para seleccionar un elemento en función, no del código como hasta ahora, sino en función del estado de este. El estado de un elemento puede depender del comportamiento del usuario, es decir, de los eventos que se produzcan. Así, por ejemplo, el estado de un enlace puede depender de los enlaces las visitas que ha realizado ya el usuario, del movimiento del ratón, etc. Normalmente las pseudo-clases se utilizan con los enlaces, aunque puede aplicarse a otros elementos HTML, y definen el estilo según los eventos que hayan ocurrido: que el enlace esté sin visitar, que haya sido visitado, que el cursor del ratón se encuentre por encima, etc.

Algunas pseudo-clases se definen a continuación:

- `:link`. Se aplica a enlaces nunca visitados.
- `:visited`. Se aplica a enlaces visitados al menos una vez.
- `:hover`. Se aplica a un elemento seleccionado por el usuario pero que no ha sido activado, como por ejemplo al situar el ratón encima sin hacer clic.
- `:active`. Se aplica a un elemento que está siendo activado, mientras el usuario hace clic con el botón izquierdo del ratón.
- `:focus`. Se aplica a un elemento que tiene el foco, por ejemplo, un cuadro de texto en un formulario.

La pseudo-clase `a:hover` debe especificarse después de `a:link` y de `a:visited`. Y la pseudo-clase `a:active` tiene que especificarse después de `a:hover`. En el ejemplo siguiente, puede observarse la definición de pseudo-clases para aplicar estilo sobre los elementos enlace de un documento HTML.

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS - Pseudo-clases</title>
  <meta charset="utf-8" />
  <style>
    p:first-child {                               /* Uso de la pseudo-clase first-child */
      color:blue;
    }
    a:link { color: black; }                      /*Enlace no visitado*/
    a:visited { color: blue; }                   /*Enlace ya visitado*/
    a:hover { color: green; }                   /*Cursor encima del Enlace*/
    a:active { color: red; }                    /*Al seleccionar el Enlace*/
    a:focus { background-color: yellow; }       /*Enlaces seleccionados con el teclado*/
  </style>
</head>
<body>
  <p>Esto es el primer párrafo</p>
  <p>Esto es un párrafo</p>
  <a href="css_syntaxis.html">Sintáxis de CSS</a>
</body>
</html>
```

Además, en el ejemplo anterior, puede apreciarse también el uso de la pseudo-clase `:first-child`. Esta pseudo-clase se refiere al elemento que es el primer hijo de otro elemento. En el ejemplo anterior, este selector se refiere a cualquier elemento párrafo que sea el primer elemento hijo de cualquier otro elemento. Por lo que solo el primer párrafo del documento HTML se presentará en azul, por ser el primer elemento hijo del elemento `<body>`.

Pseudo-elementos

Los pseudo-elementos hacen referencia a algunas partes de los elementos de una página a los que HTML no otorga identidad propia. Algunos de los pseudo-elementos más utilizados son:

- `:first-letter`. Se aplica al primer carácter del contenido del elemento.
- `:first-line`. Se aplica a la primera línea del elemento.
- `:before`. Se aplica al elemento anterior.
- `:after`. Se aplica al elemento posterior.

El siguiente ejemplo pone la primera letra de cada párrafo con un tamaño 16 y, además, la primera línea en color rojo y con mayúsculas de un tamaño menor.

```
p:first-letter {
  font-size: 16pt;
}
p:first-line
{
  color:#ff0000;
  font-variant:small-caps;
}
```

El siguiente ejemplo inserta la imagen smiley.gif delante de todos los <h1> y detrás de los <h2>:

```
h1:before {  
  content:url(smiley.gif);  
}  
h2:after {  
  content:url(smiley.gif);  
}
```

Selectores jerárquicos

Hasta este momento, se han estudiado los selectores básicos que, aunque con sus limitaciones, permiten seleccionar elementos para asignar de estilos perfectamente. Las especificaciones CSS2 y CSS3 han agregado multitud de selectores. Los selectores jerárquicos son selectores avanzados que permiten definir criterios de selección complejos. Los selectores jerárquicos son los siguientes:

Sintaxis	Selector	Descripción
A B	Descendiente	Selecciona cualquier elemento B que es descendiente del elemento A.
A > B	Hijo	Selecciona cualquier elemento B que es hijo directo del elemento A.
A + B	Hermano adyacente	Selecciona cualquier elemento B que está situado de forma inmediata detrás del elemento A. Los elementos A y B deben ser hermanos.

Los selectores jerárquicos, por tanto, pueden seleccionar los elementos en función de dónde se encuentran.

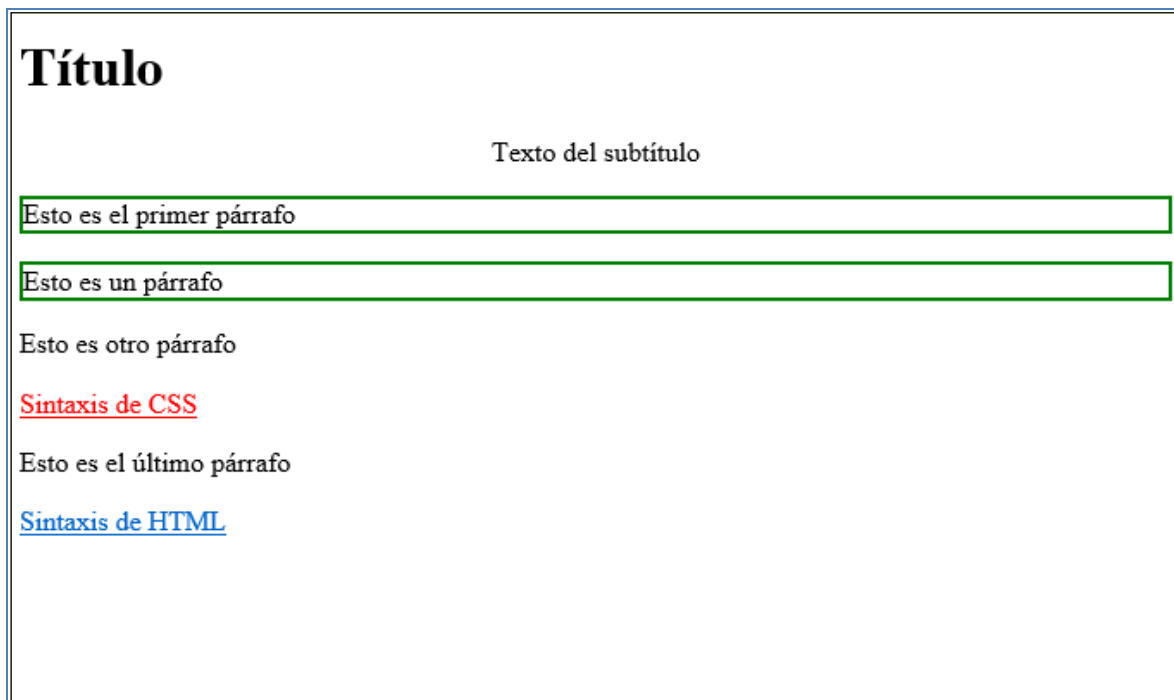
- **Selector descendiente.** Se usan dos selectores separados por un espacio, siendo el primero de ellos el ascendiente. En el siguiente ejemplo, se aplica la declaración de estilo a todos los párrafos incluidos en las divisiones pertenecientes a la clase *des*:

```
<!DOCTYPE html>  
<html>  
<head>  
  <title>CSS - Selectores jerárquicos</title>  
  <meta charset="utf-8" />  
  <style>  
    div.des p {  
      border-color: green;  
      border-width: 2px;  
      border-style: solid;  
    }  
  
    div > a {  
      color: red;  
    }  
  
    h1 + p {  
      text-align: center;  
    }  
  </style>  
</head>
```

```
<body>
  <h1>Título</h1>
  <p>Texto del subtítulo</p>
  <div class="des">
    <div>
      <p>Esto es el primer párrafo</p>
      <p>Esto es un párrafo</p>
    </div>
  </div>
  <div>
    <p>Esto es otro párrafo</p>
    <a href="css_syntaxis.html">Sintaxis de CSS</a>
  </div>
  <p>Esto es el último párrafo</p>
  <a href="html_syntaxis.html">Sintaxis de HTML</a>
</body>
</html>
```

- **Selector hijo.** Es similar al selector anterior solo que en este caso afecta solo a descendientes directos (hijos). Se usan dos selectores separados por el símbolo >, siendo el primero de ellos el padre y el segundo el hijo. En el ejemplo anterior, se aplica a los elementos enlace que se encuentren directamente dentro de una división, sin que haya ninguna etiqueta interpuesta.
- **Selector hermano adyacente.** Se emplea para seleccionar elementos que son hermanos, es decir, que tienen el mismo padre y que, además, son adyacentes o consecutivos en el código. En el ejemplo anterior, el texto del párrafo del subtítulo se presentará centrado.

El ejemplo anterior, produce el siguiente resultado en la ventana de visualización del navegador:



5.5 HERENCIA Y PRECEDENCIA

Las hojas de estilo en cascada, CSS, se caracterizan por incluir herencia. Así, las declaraciones de estilo afectan a los elementos definidos por el selector y también a sus descendientes. Por ejemplo, si se especifica la propiedad color para el elemento body, los elementos incluidos en la etiqueta body heredarán dicha propiedad. Sin embargo, también hay propiedades que no se heredan, como la propiedad border. Para conseguir que se herede dicha propiedad, se debe asignar el valor inherit a la propiedad que queremos que se herede del elemento padre, de la siguiente forma:

```
p {  
  border: inherit;  
}
```

Así, los párrafos heredarán las características de estilo del borde definido en un elemento padre.

En los apartados anteriores, se ha estudiado cómo es posible disponer de varias ubicaciones de las hojas de estilo: en línea, internas o externas. En caso de que exista una competencia entre varios elementos de estilo, interviene un orden de prioridad. En primer lugar, se toman en consideración las especificaciones de las hojas de estilo externas que se especifican en archivos .css y se incluyen en el documento HTML mediante la etiqueta de cabecera <link>, a continuación, las hojas de estilo internas que se especifican mediante la etiqueta de cabecera <style> y, finalmente, las hojas de estilo en línea que se especifican en la etiqueta del elemento mediante el atributo style.

En caso de conflicto o colisión entre una especificación de estilo definida al mismo tiempo en una hoja de estilo externa y en una hoja de estilo interna, la especificación que prevalece es la definida en la hoja de estilo interna. Del mismo modo, en caso de conflicto entre una hoja de estilo interna y una declaración en línea, esta última es la que prevalece. Los estilos declarados en línea tienen, por tanto, la prioridad más alta. La regla de prioridad consiste, por tanto, en considerar la hoja de estilo más próxima al elemento correspondiente.

En ocasiones, puede ocurrir que se definan dos reglas de estilo para un mismo elemento y, además, en la misma ubicación. Por ejemplo, en una hoja de estilo interna podría especificarse:

```
p#parrafo1 {  
  font-size: 14pt;  
}  
p {  
  font-size: 8pt;  
}
```

En este caso, se indica que todos los párrafos deben tener tamaño de letra 8, mientras que el párrafo identificado como párrafo1 debe tener tamaño 14. Entonces, ¿Qué tamaño será el que se aplicará, finalmente, a párrafo1? La respuesta es que se aplicará el tamaño 14, ya que esa regla es más concreta que la que se aplica a todos los párrafos en general. En conclusión, cuanto más concreto o específico sea el selector, más valor o precedencia tiene la regla. Así, el atributo id prevalecerá sobre el atributo class, y éste sobre cualquier selector simple. Además, los selectores simples se aplican por delante de las reglas heredadas.

5.6 EJEMPLOS DE APLICACIÓN DE LAS HOJAS DE ESTILO

En la siguiente dirección Web del sitio *w3schools* se accede a un gran número de ejemplos sencillos sobre Hojas de estilo CSS que ilustran el funcionamiento de las propiedades de estilo en un documento HTML.

http://www.w3schools.com/css/css_examples.asp

Los ejemplos están clasificados para hacer más sencillo su acceso, además puede consultarse el código al mismo tiempo que se visualiza el resultado de la presentación. Puede cambiarse el idioma a español. Se recomienda analizar el código de cada uno de los ejemplos para familiarizarse con el uso de las propiedades de estilo, así como con la construcción de las reglas de estilo. Igualmente, se recomienda realizar modificaciones sobre el código para indagar y experimentar sobre la base de los ejemplos propuestos.

Glosario de términos

Accesibilidad Web. Un sitio web accesible se caracteriza por ser transformable, comprensible y navegable. El término transformable se aplica a que la información y los servicios del sitio Web deben ser accesibles para todas las personas y deben poder ser utilizados con todos los dispositivos de navegación. El término comprensible se refiere a que el sitio Web debe incluir contenidos claros y simples. El término navegable significa que el sitio Web debe incluir mecanismos sencillos de navegación.

Diseño Web *responsive* o adaptativo. El diseño web adaptativo, o adaptable, conocido por las siglas RWD del inglés, *Responsive Web Design*, es una filosofía de diseño y desarrollo de sitios Web cuyo objetivo es adaptar la apariencia de las páginas Web al dispositivo que se esté utilizando para visualizarlas. De manera que la presentación de la información sea independiente del dispositivo y, por tanto, la visualización de la información sea adecuada. Hoy día las páginas Web se presentan en multitud de dispositivos como ordenadores, tabletas, teléfonos móviles, libros electrónicos, ordenadores portátiles, etc. Además, aún dentro de cada tipo, cada dispositivo tiene sus características concretas: tamaño de pantalla, resolución, potencia de CPU, sistema operativo o capacidad de memoria entre otras. Esta tecnología pretende que, un único diseño Web se obtenga una visualización adecuada en cualquier dispositivo. El diseñador y autor norteamericano Ethan Marcotte creó y difundió esta técnica a partir de una serie de artículos en *A List Apart*, que es una publicación en línea especializada en diseño y desarrollo Web. Esta idea inicial la extendió en su libro *Responsive Web Design*, posteriormente.