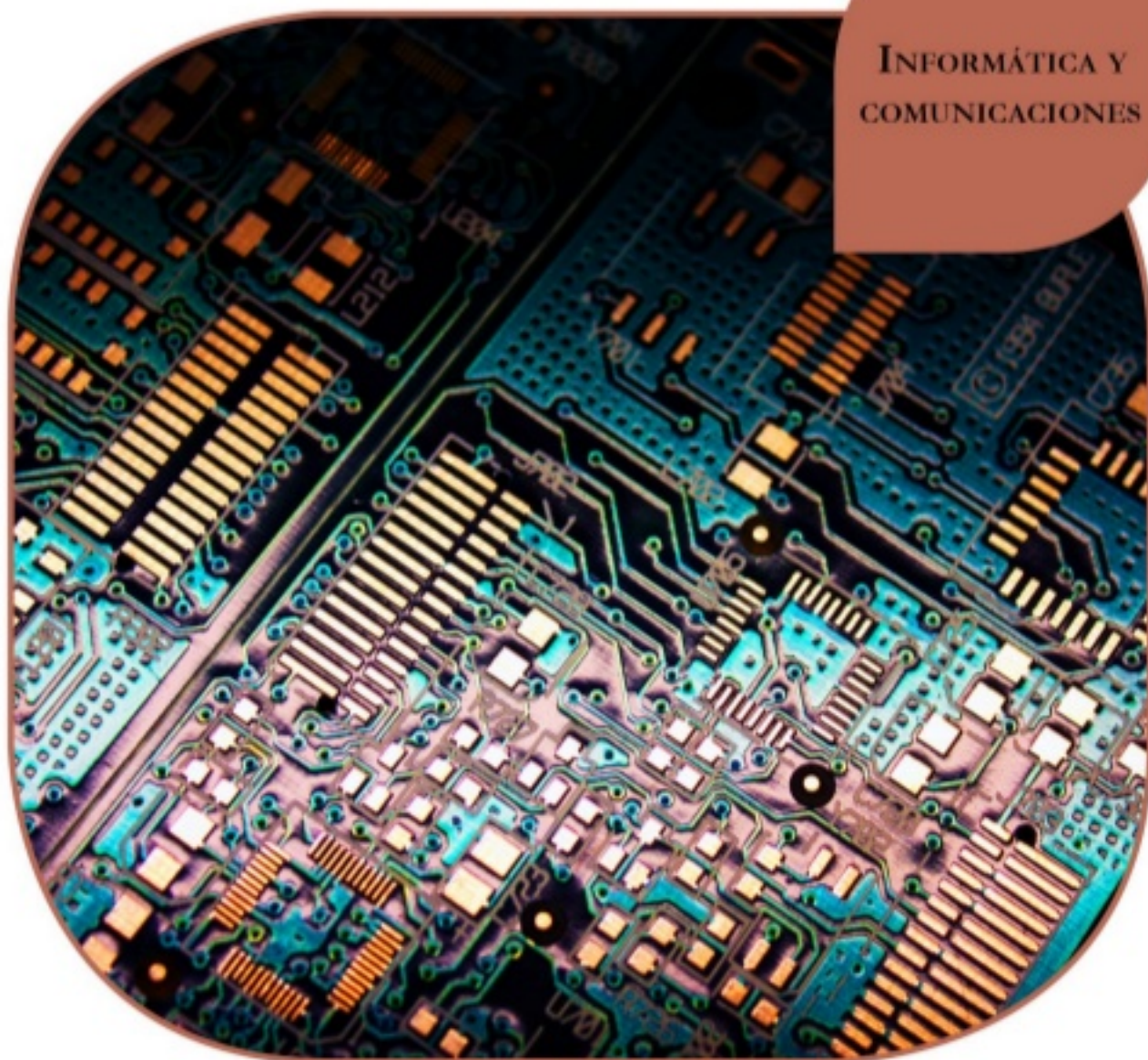


INFORMÁTICA Y
COMUNICACIONES



UF1303. ELABORACIÓN DE HOJAS DE ESTILO

Contenidos basados en los Certificados de profesionalidad



Certia
editorial

UF1303. ELABORACIÓN DE HOJAS DE ESTILO



Certia
editorial

DATOS DEL AUTOR

Enrique García Casal es ingeniero técnico en Informática de Gestión por la Universidad de Vigo.

Ha ejercido como profesional de la informática desde 2002 desarrollando software para grandes multinacionales de seguros y banca durante su estancia en Madrid. Es aquí donde descubre las posibilidades de HTML y comienza su aprendizaje y formación autodidacta en esta materia.

En 2005 se traslada a Galicia, donde sigue desarrollando software como analista programador en una conocida consultora.

En 2006 ficha por una empresa de comercio electrónico dedicada al turismo y comienza a experimentar con otras tecnologías de cliente como Flex, que le permiten tener una visión más profunda de lo que se puede conseguir desde el lado del cliente de una aplicación web.

En la actualidad ejerce como freelance a la vez que sigue formándose en nuevas tecnologías como AngularJS o NodeJS, y lo combina con el desarrollo de manuales formativos relacionados con certificados de profesionalidad.

FICHA

Elaboración de hojas de estilo.
Informática y comunicaciones

1ª Edición
Certia Editorial, Pontevedra, 2016

Autor: Enrique García Casal

Formato: 170 x 240 mm • 131 páginas.

ELABORACIÓN DE HOJAS DE ESTILO. INFORMÁTICA Y COMUNICACIONES.

No está permitida la reproducción total o parcial de este libro, ni su tratamiento informático, ni la transmisión de ninguna forma o por cualquier medio, ya sea electrónico, mecánico, por fotocopia, por registro u otros métodos, sin el permiso previo y por escrito de los titulares del Copyright.

Derechos reservados 2016, respecto a la primera edición en español, por Certia Editorial.

ISBN: 978-84-16481-09-5

Depósito legal: PO 207-2016

Impreso en España - Printed in Spain

Certia Editorial ha incorporado en la elaboración de este material didáctico citas y referencias de obras divulgadas y ha cumplido todos los requisitos establecidos por la Ley de Propiedad Intelectual. Por los posibles errores y omisiones, se excusa previamente y está dispuesta a introducir las correcciones pertinentes en próximas ediciones y reimpresiones.

Fuente fotografía portada: MorgueFile, autoriza a copiar, distribuir, comunicar públicamente la obra y adaptar el trabajo.

Familia profesional: INFORMÁTICA Y COMUNICACIONES

Área profesional: Desarrollo

FICHA DE CERTIFICADO DE PROFESIONALIDAD (IFCD0110) CONFECCIÓN Y PUBLICACIÓN DE
PÁGINAS WEB (RD 1531/2011, de 31 de octubre modificado por el RD 628/2013, de 2 de agosto)

Correspondencia con el Catálogo Modular de Formación Profesional				
H. Q	Módulos certificado	H.CP	Unidades formativas	Horas
210	MF0950_2: Construcción de páginas web	210	UF1302: Creación de páginas web con el lenguaje de marcas	80
			UF1303: Elaboración de hojas de estilo	70
			UF1304: Elaboración de plantillas y formularios	60
210	MF0951_2: Integración de componentes soft	180	UF1305: Programación con lenguajes de guión en páginas web	90
90	MF0952_2: Publicación de páginas web	90	UF1306: Pruebas de funcionalidades y optimización de páginas web	90
	MP0278: Módulo de prácticas profesionales no laborales	80		90
510	Duración horas totales certificado de profesionalidad	560	Duración horas módulos formativos	480

ÍNDICE

• INTRODUCCIÓN.....	11
• UNIDAD DIDÁCTICA 1. Hojas de estilo en la construcción de páginas web.....	15
1.1. Funciones y características.....	17
1.1.1. Descripción de estilos	17
1.1.2. Utilización de estilos.....	18
1.1.3. Los estilos en el lenguaje de marcas.....	19
1.1.4. Los estilos con herramientas de edición web	20
1.2. Hojas de estilo y accesibilidad.....	24
1.2.1. Adecuación de las hojas de estilos.....	24
1.3. Tipos de estilo: incrustados, enlazados, importados, en línea.....	29
1.3.1. Descripción de los tipos de estilo.....	29
1.3.2. Enlazar una hoja de estilo externa a un documento HTML...30	
1.3.3. Incrustar un estilo dentro de un documento HTML.....	32
1.3.4. Importar una hoja de estilo desde un documento HTML	34
1.3.5. Importar una hoja de estilo a través de un archivo con estilos	35
1.3.6. Utilización y optimización de los tipos de estilos	36
1.4. Selectores y reglas de estilo.....	39
1.4.1. Estructura de los estilos.....	43

1.4.2. Sintaxis básica de estilos.....	44
1.4.3. Utilización de elementos y pseudoelementos.....	45
1.4.4. Utilización de clases y pseudoclases.....	46
1.5. Atributos de estilo para fuentes, color y fondo, texto y bloques (párrafos).....	48
1.5.1. Descripción de los atributos de estilo.....	48
1.5.2. Utilización de los atributos de estilo.....	51
1.6. Creación de ficheros de estilo.....	64
1.6.1. Definición de los ficheros de estilo.....	64
1.6.2. Creación de ficheros de estilo genéricos.....	66
1.6.3. Adaptación de los ficheros de estilo para distintos dispositivos.....	68
RESUMEN.....	73
EVALUACIÓN.....	75
 • UNIDAD DIDÁCTICA 2. Diseño, ubicación y optimización de los contenidos de una página web.....	 77
2.1. Creación de un documento funcional.....	79
2.1.1. Descripción de los objetivos de la página web.....	79
2.1.2. Definición de los elementos funcionales de la página web.....	80
2.1.3. Descripción de cada elemento.....	82
2.2. Diseño de los contenidos.....	82
2.3. Identificación de la información a ubicar en la página web.....	84

2.4. Selección de contenidos para cada elemento de la página.....	85
2.5. Utilización del documento funcional para las especificaciones del diseño.....	86
2.6. Tipos de página para la ubicación de contenidos.....	87
2.7. Definición de los tipos de página en base a los contenidos y funcionalidades.....	92
2.8. Selección de los tipos de página para la página web.....	93
2.9. Utilización del documento funcional para las especificaciones del tipo de página.....	93
2.10. Especificaciones de navegación.....	94
2.11. Creación de un mapa de navegación de páginas.....	95
2.12. Utilización del documento funcional para integrar el mapa de navegación.....	97
2.13. Elementos utilizados para la navegación.....	99
2.13.1. Definición de los elementos utilizados para navegar.....	99
2.13.2. Utilización del documento funcional para especificar los elementos de navegación.....	100
2.14. Elaboración de una guía de usuario.....	100
RESUMEN.....	103
EVALUACIÓN.....	105
 • RESUMEN FINAL.....	 109
 • EJERCICIOS Y ACTIVIDADES.....	 111

• EVALUACIÓN FINAL	117
• SOLUCIONES	123
• BIBLIOGRAFÍA/WEBGRAFÍA	127

INTRODUCCIÓN

El desarrollo de una página web implica el conocimiento de varias tecnologías que nos llevarán a visualizar en un navegador todo el trabajo realizado durante dicho desarrollo. HTML (*HiperText Markup Language*) nos ayuda a crear el contenido, y JavaScript y los lenguajes de *script* en general (*ECMAScript*) nos ayudan a crear interacciones con el usuario, validaciones de formularios, etc. Para modificar la apariencia estándar de una página web se crearon las *hojas de estilo en cascada* o *CSS* (siglas en inglés de *Cascading Style Sheets*).

CSS se define como «un lenguaje usado para definir y crear la presentación de un documento escrito en HTML»¹.



Hasta el momento existen tres versiones de CSS. La entidad encargada de formular las especificaciones de cada nivel o versión es el *World Wide Web Consortium* (W3C), cuyas guías deben seguir los navegadores para dar soporte y los desarrolladores para crear las páginas web. Cada versión está construida sobre la anterior y le añade nuevas funcionalidades. Brevemente, cada una de esas tres versiones ofrece las funcionalidades que comentamos a continuación:

- CSS1 (iniciada en 1996 y abandonada en 2008):
 - Propiedades para modificar los estilos de las fuentes de texto: tipo, tamaño, énfasis, etc.
 - Propiedades para cambiar el color en texto, fondos, bordes y otros elementos.

¹ Cita obtenida de WIKIPEDIA, Hoja de Estilos en Cascada [Internet] Disponible en: http://es.wikipedia.org/wiki/Hoja_de_estilos_en_cascada [Fecha de acceso: 13/04/2015].

- Propiedades de texto como pueden ser espaciado entre palabras, letras, líneas, etc.
 - Propiedades para la alineación de las imágenes textos, tablas, etc.
 - Propiedades para los elementos de caja (por ejemplo el elemento HTML <DIV>) como pueden ser borde, margen, relleno o espaciado.
 - Propiedades de identificación y presentación del elemento HTML lista (, , <DL>).
- CSS2 (iniciada en 1998 y abandonada en 2008):
 - Posicionamiento relativo/absoluto/fijo de las capas (<DIV>), niveles de cada capa (z-index), etc.
 - Se definen los «media types», que indican en qué dispositivo será mostrado el documento vinculado.
 - Se da soporte a las hojas de estilo auditivas.
 - Propiedades para mostrar el texto bidireccional, mostrar sombras, etc.
 - CSS2.1: es una revisión de CSS2 cuyo objetivo es corregirla y mejorarla. Después de varias revisiones y rechazos, fue finalmente aprobada como recomendación oficial en junio de 2011.
 - CSS3: es la última especificación publicada, pero tiene la particularidad de que no es una única especificación que engloba varias funcionalidades, sino que está dividida en módulos, cada uno con sus funcionalidades definidas y con su estado de desarrollo y publicación. A día de hoy hay una gran cantidad de módulos publicados como recomendaciones oficiales, y la gran mayoría de ellos están implementados en los navegadores más importantes en sus últimas versiones. Algunas de las funcionalidades añadidas por CSS3 son:
 - Posibilidad de aplicar gradientes en bordes y fondos de los elementos.
 - Posibilidad de aplicar bordes redondeados en los elementos.

- Definición de colores en formato RGBA.
- Animaciones.
- Transiciones.



CSS es, pues, una especificación que está en constante evolución, añadiendo capacidades y revisando las ya existentes, teniendo siempre como objetivo conseguir una web más atractiva para el usuario y más sencilla de implementar para el desarrollador.

HOJAS DE ESTILO EN LA CONSTRUCCIÓN DE PÁGINAS WEB

1

- **CONTENIDO**

- 1.1. Funciones y características.
- 1.2. Hojas de estilo y accesibilidad
- 1.3. Tipos de estilo: incrustados, enlazados, importados, en línea.
- 1.4. Selectores y reglas de estilo
- 1.5. Atributos de estilo para fuentes, color y fondo, texto y bloques (párrafos)
- 1.6. Creación de ficheros de estilo

- **RESUMEN**

- **EVALUACIÓN**

Las hojas de estilo CSS juegan un papel fundamental en la construcción de páginas web. La separación entre el contenido y la apariencia de una página es fundamental para conseguir un diseño atractivo, para realizar un buen desarrollo del proyecto y para su posterior mantenimiento.

1.1. Funciones y características

Las hojas de estilo CSS se usan para modificar la apariencia de un documento web. Asociando estilos a un elemento es posible cambiar la apariencia que por defecto le aplica el navegador web cuando es renderizado para su visualización.

1.1.1. Descripción de estilos

Un estilo CSS se define de una forma muy sencilla usando palabras claves definidas por la W3C. Los estilos se definen dentro de *reglas* que, a su vez, están compuestas por uno o más *selectores* y un *bloque de declaración*. Este bloque de declaración es el «contenedor» donde se definen los estilos que se van a aplicar a los elementos del documento que cumplan con el selector. Estará compuesto de pares *«propiedad: valor;»*.

El *selector* define qué elementos del HTML se verán afectados por las declaraciones hechas dentro de su bloque de declaración. Los tipos de selectores pueden ser por nombre (usando el nombre del elemento), por ID (usando el atributo «ID» del elemento), por clase (usando el atributo *class* del elemento), por posición (usando las llamadas pseudoclases), etc.

Un ejemplo genérico de sintaxis de una regla CSS es el siguiente:

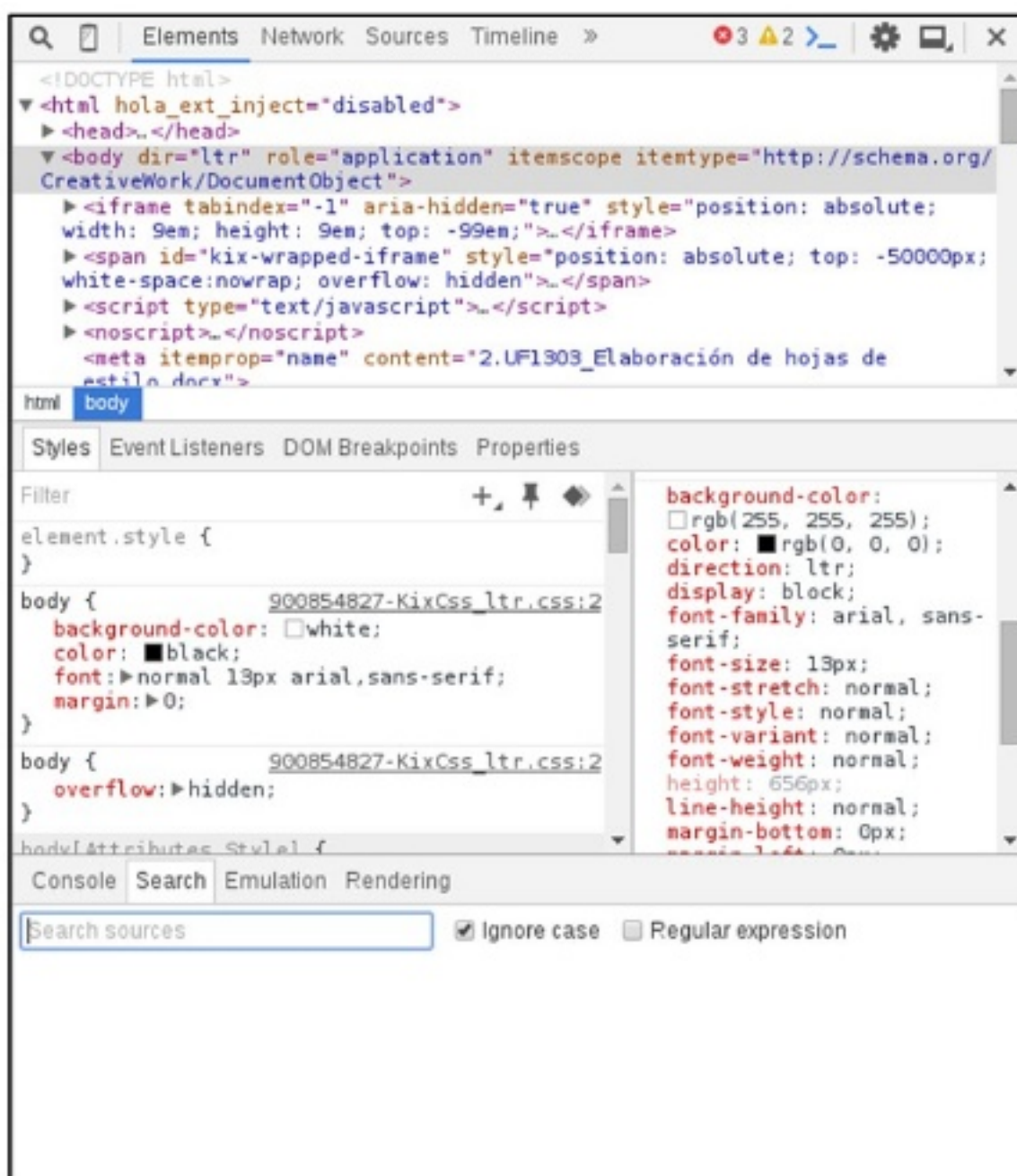


1.1.2. Utilización de estilos

Los estilos CSS pueden usarse de tres formas distintas a la hora de dar formato a un documento HTML:

- El autor del documento introduce los estilos CSS en sus documentos, usando estilos en línea, hojas de estilos internas u hojas de estilos externas (como veremos posteriormente).
- Es posible usar estilos CSS introducidos por el usuario que accede al documento usando un archivo CSS configurado en el navegador, y que sobrescribirá los estilos definidos por el autor de la/s página/s web.
- Los navegadores tienen también una serie de estilos marcados por defecto para los diferentes elementos HTML (*body*, enlaces, tablas, etc.).

Por supuesto, el modo más habitual de usar los estilos CSS es en el momento del desarrollo de la página web, por parte del autor, introduciendo los estilos en hojas externas. Para emplearlas otras dos formas de trabajar con estilos, los navegadores actuales aportan las llamadas «herramientas para diseñadores» (habitualmente accesibles con la tecla F12), que permiten modificar los estilos de una página ya cargada en el navegador. La imagen que vemos a continuación muestra las «herramientas para diseñadores» del navegador Google Chrome:



Con esta herramienta es posible realizar muchas más acciones, por lo que es recomendable para cualquier desarrollador aprender a manejarla.

1.1.3. Los estilos en el lenguaje de marcas

En versiones anteriores a HTML5, la W3C definió los elementos HTML con un conjunto de propiedades que ayudaban a formatearlos y posicionarlos. Tal es el caso de atributos como «*bgscolorm*» en el elemento <BODY>, o el atributo «*border*» en el elemento <TABLE>. En estos casos, la especificación HTML5 opta por eliminar esas propiedades para que la aplicación de estos estilos se haga a través de CSS. En otros casos, la especificación HTML5 directamente elimina

elementos usados para cambiar el aspecto de lo que encierra, como puede ser la etiqueta `` para los textos.

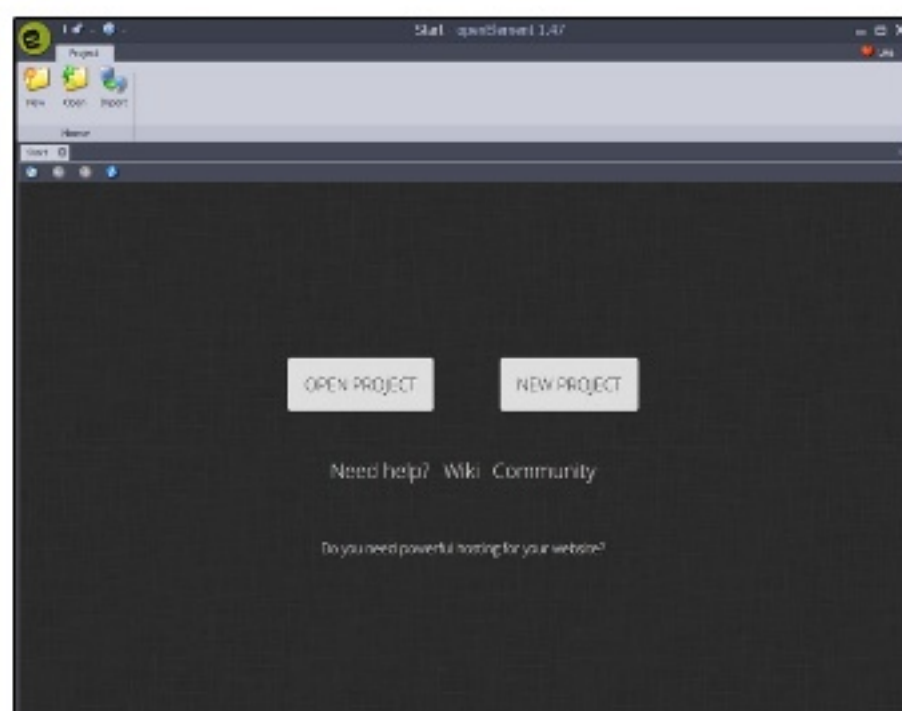
Todas estas decisiones tomadas por la W3C van encaminadas a intentar que en los desarrollos se lleve a cabo lo ya comentado aquí: la separación entre contenido y apariencia. Algunas de las ventajas conseguidas con esta separación son:

- Con el cambio del valor de una regla, como puede ser el tipo de letra, se puede modificar toda la apariencia del sitio web.
- A la hora de desarrollar un sitio web, el desarrollador, centrado en el contenido y la estructura del sitio web, puede realizar su labor independientemente del diseñador, más centrado en la apariencia visual.
- Al poder reutilizar estilos se ahorra en ancho de banda consumido.
- Se mejora la accesibilidad, ya que cada elemento HTML se usará para lo que fue concebido y no, por ejemplo, para posicionar otros elementos como pasaba con el elemento `<TABLE>`.

Todas estas ventajas aportadas por CSS hacen imprescindible para el desarrollador conocer sus especificaciones y usarlas en el desarrollo de sitios web.

1.1.4. Los estilos con herramientas de edición web

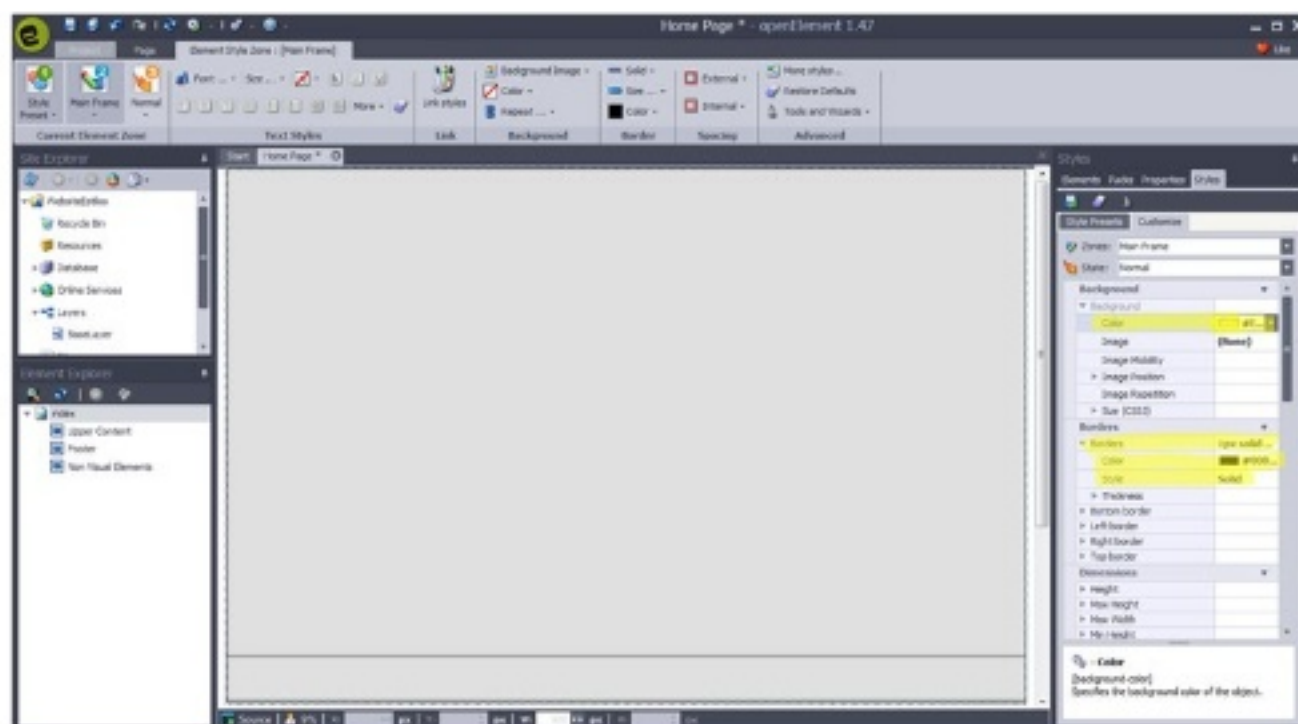
Al igual que el desarrollo de documentos HTML, desarrollar hojas de estilo CSS no requiere más herramientas que un editor de textos como pueden ser el bloc de notas de Windows o cualquier editor de otra plataforma. Sin embargo existen herramientas en el mercado con licenciamiento de pago, gratuitas, *software* libre, etc. que ayudan en la edición de documentos HTML y en la aplicación de estilos CSS a los elementos HTML que forman dicho documento sin necesidad de tener un gran conocimiento sobre CSS y HTML. Estas herramientas tienen la característica de ser WYSIWYG (acrónimo de *What You See Is What You Get*, en español ‘Lo que ves es lo que obtienes’), por lo que permiten ver el resultado final al mismo tiempo que se desarrolla. La herramienta más conocida es, sin duda, Adobe Dreamweaver, pero se trata de una herramienta de pago. Para ilustrar el uso de este tipo de herramientas se utilizará otra herramienta gratuita llamada OpenElement (www.openelement.com).



Una vez instalado y creado el proyecto que se va a desarrollar, es muy sencillo aplicar estilos a los elementos HTML usando la herramienta «*style*»:



El siguiente ejemplo muestra cómo se aplicó un color gris al fondo y un borde de color negro de 1pt de grosor y con un tipo de línea sólida de un elemento <DIV> seleccionado con el ratón.



Estos estilos se guardan en la regla de estilo definida por el selector ID del elemento <DIV> seleccionado, que en este caso se llama «XBody (id="XBody")», y se almacenan en una hoja de estilos externa llamada «index.css» vinculada en la sección <HEAD> con la etiqueta:



Esta hoja de estilos externa se localiza en la raíz de la carpeta donde OpenElement generó el proyecto. Si se edita esa hoja de estilos se puede encontrar el siguiente código CSS, que es precisamente la regla que se ha creado para el elemento con ID XBody.



El código HTML generado por OpenElement se puede consultar pinchando en el botón «Source», y será el mostrado en la siguiente imagen:

```

1 Add new
2 <!DOCTYPE HTML>
3 <html xmlns="http://www.w3.org/1999/xhtml" dir="ltr" lang="es">
4 <head>
5 <meta http-equiv="content-type" content="text/html; charset=UTF-8" />
6 <meta name="generator" content="openElement (1.47)" />
7 <link id="openElement" rel="stylesheet" type="text/css" href="WEFiles/Css/v02/openElement.css" />
8 <link id="OETemplate1" rel="stylesheet" type="text/css" href="Templates/BaseLayer.css?v=63562" />
9 <link id="OEBase" rel="stylesheet" type="text/css" href="index.css?v=635623020214289527" />
10 <!--[if lte IE 7]>
11 <link rel="stylesheet" type="text/css" href="WEFiles/Css/ie7.css?v=635083121531874595" />
12 <![endif]>
13 <script type="text/javascript">
14 var WEInfoPage = {"PHPVersion":"phpOK","OEVersion":"1-47-0","PagePath":"index","Culture":"DEF"
15 </script>
16 <script type="text/javascript" src="WEFiles/Client/jquery/1.10.2.js?v=635144069497419815"></script>
17 <script type="text/javascript" src="WEFiles/Client/jquery/migrate.js?v=635562228397726464"></script>
18 <script type="text/javascript" src="WEFiles/Client/Common/oe.min.js?v=635582321480031784"></script>
19 Add new
20
21 </head>
22 <body class="">
23 <form id="XForm" method="post" action="#"></form>
24 Add new
25 <div
26 id="XBody" class="BaseDiv RBoth OEPageXbody OESK XBody_Default" style="z-index:1000">
27 <div class="OES2 OES2_DivContent OES2G XBody W">
28 <div class="OES2 OES2_XBodyContent OES2G XBody OECT OECT_Content OECTAbs">
29 </div>
30 <div class="OES2 OES2_XBodyFooter OES2G XBody OECT OECT_Footer OECTAbs">
31

```

En internet se pueden encontrar multitud de recursos que sirven de ayuda en la creación de distintos elementos CSS. Algunos de ellos se nombran a continuación

Uso	Recurso
Diseño de rejillas ²	Grid Designer 2 Grid Generator Gridnator Grid System Generator Variable Grid System
Herramientas para menús CSS	CSS Menu Builder CSS Menu Maker CSS Menu Generator
Herramientas para botones CSS	Navigation Tab Menu Generator CSS Button and Field Tool
Generador de sprite CSS ³	CSS Sprite Generator
Herramienta para formularios CSS	Quick Form Builder

² «Rejilla» o «grid» es un conjunto de filas y columnas formadas por DIV que se usan para estructurar la página web.

³ Se denomina «sprite» a un conjunto de imágenes pequeñas agrupadas en una sola imagen. Es una técnica usada para ahorrar ancho de banda.

1.2. Hojas de estilo y accesibilidad

Se define la accesibilidad como la capacidad que tiene un sitio web de ser consultado por un usuario independientemente de su discapacidad, de acceder a sus contenidos y de navegar a través de él.

Un desarrollador de páginas web que adopte el uso de hojas de estilos CSS tendrá un control efectivo sobre la apariencia de las páginas y podrá eliminar código superfluo innecesario para la visualización. De esta forma se conseguirá que las páginas sean más accesibles para el usuario común y, además, esto repercutirá positivamente en la accesibilidad de todo tipo de usuarios a la página, ya que se conseguirá disminuir el tiempo de carga.

La primera medida que se debería tomar para hacer accesible la página es prescindir de las tablas (elemento `<TABLE>` de HTML) para su posicionamiento y estructura. Las tablas deben usarse única y exclusivamente para mostrar datos que requieran una estructura de tabla para ser mostrados. Esto repercutirá en una mejor lectura por parte de los *screen readers*⁴, lo que mejorará la accesibilidad de la página.

Otras medidas habituales son usar CSS siempre que se pueda evitando el uso de técnicas propias de HTML para aplicar estilos a elementos como ``, ya comentado, etc.

1.2.1. Adecuación de las hojas de estilos

Para adecuar las hojas de estilo a la accesibilidad se recogen numerosas técnicas recomendadas por la W3C que comentamos a continuación:

- Usar un número mínimo de hojas de estilo.
- Usar hojas de estilo externas en lugar de estilos en línea u hojas de estilos internos.
- En el caso de tener más de una hoja, usar el mismo nombre de clase (atributo *class*) para el mismo concepto en todas las hojas de estilos.

⁴ «Screen reader» o «lector de pantalla» es una aplicación software que trata de identificar e interpretar aquello que se muestra en pantalla representado mediante sintetizadores de texto a voz, iconos sonoros o salida en braille.

- Usar la unidad «*em*» para fijar el tamaño de la letra, en lugar de «*pt*».
- Usar unidades de medida relativas y porcentajes. Usar unidades absolutas de medida solo para imágenes de mapa de bits.
- Proporcionar un equivalente textual para cualquier imagen o texto importante generado por la hoja de estilos (es decir, cuando se use *background-image*, *list-style*, *content*, etc). CSS2 permite a los usuarios acceder a representaciones alternativas de los contenidos especificadas en valores de los atributos cuando se emplean juntos:
 - Selectores de atributos.
 - La función «*attr()*» y la propiedad «*content*».
 - Los pseudoelementos «*:before*» (antes) y «*:after*» (después).
- Todo contenido importante debe aparecer dentro del objeto del documento, ya que el texto generado por las hojas de estilo no forma parte del código fuente del documento y no estará disponible para las ayudas técnicas que acceden al HTML.
- Especificar siempre un tipo de letra genérico por defecto.
- Usar las siguientes propiedades CSS2 para controlar la información de la fuente: «*font*», «*font-family*», «*font-size*», «*font-size-adjust*», «*font-stretch*», «*font-style*», «*font-variant*» y «*font-weight*», en lugar de los siguientes elementos y atributos de tipo de letra desaconsejados en HTML: , <BASEFONT>, «*face*» y «*size*».
- Si es necesario usar los elementos HTML para controlar la información sobre el tamaño de letra, utilizar <BIG> y <SMALL>, que no están desaconsejados.
- Para dar estilo al texto se puede utilizar:
 - Mayúsculas/minúsculas: «*text-transform*» (para mayúsculas, minúsculas y primera letra mayúscula).
 - Efectos de sombra: «*text-shadow*».
 - Subrayado: «*text-decoration*».

- No usar los elementos HTML `<BLINK>` y `<MARQUEE>`, ya que no pertenecen a ninguna especificación HTML. Si se desea que un texto parpadee se puede utilizar *«text-decoration: blink»*.
- Usar hojas de estilo para dar estilos al texto mejor que utilizar imágenes para representar el texto.
- Emplear las siguientes propiedades CSS definidas en la versión 2 para formatear el texto y su posición en la página:
 - *«text-indent»*: usado para sangrías. No se deben utilizar `<BLOCKQUOTE>` o espacios para hacer las sangrías.
 - *«letter-spacing»*, *«word-spacing»*: usado para controlar el espacio entre letras o entre palabras.
 - *«white-space»*: esta propiedad controla el espacio en blanco del contenido de un elemento.
 - *«direction»*, *«unicode-bidi»*: propiedades usadas para controlar la dirección del texto.
 - *«:first-letter»* y *«:first-line»*: pseudoelementos usados para hacer referencia a la primera letra o primera línea de un párrafo.
- Para la aplicación de colores se recomienda usar números y no nombres. Las propiedades CSS para especificar colores son las siguientes:
 - *«color»*: usado para el color de primer plano del texto.
 - *«background-color»*: usado para especificar el color de fondo.
 - *«border-color»*, *«outline-color»*: usados para especificar colores de bordes.
 - *«:link»*, *«:visited»* y *«:active»*: pseudoclases usadas para especificar colores de los enlaces.
- Cerciorarse de que los colores de primer plano y de fondo tienen buen contraste. Si se especifica el color de primer plano, siempre se debe especificar también el color de fondo (y viceversa).
- Asegurarse de que la información no se transmite solo a través del color.

- Emplear para listas sin ordenar y para las ordenadas (por ejemplo, utilizar marcadores de forma apropiada) conjuntamente con CSS para proporcionar indicaciones contextuales.
- Evitar el uso de tablas para la maquetación, ubicación o alineación de los contenidos. Para ello deben usarse las siguientes propiedades CSS:
 - «*text-indent*», «*text-align*», «*word-spacing*» y «*font-stretch*»: permiten controlar el espaciado sin añadir espacios adicionales.
 - «*text-align:center*»: permite centrar el texto.
 - «*margin*», «*margin-top*», «*margin-right*», «*margin-bottom*» y «*margin-left*»: permiten crear espacios en los cuatro lados del contenido de un elemento.
 - «*float*», «*position*», «*top*», «*right*», «*bottom*» y «*left*»: permiten posicionar cualquier elemento independientemente del lugar que le correspondería.
 - «*empty-cells*»: permite dejar vacías celdas de tablas. Es utilizada para evitar el uso de un espacio en blanco o un espacio *non-breaking* para lograr el efecto visual.
- Proporcionar un texto equivalente para cualquier imagen usada en la página web se debe, incluso para las imágenes invisibles o transparentes. Para ello es necesario usar la propiedad «*alt*» del elemento .
- Utilizar las propiedades CSS para especificar los estilos de los bordes:
 - «*border*», «*border-width*», «*border-style*», «*border-color*».
 - «*border-spacing*» y «*border-collapse*» para los bordes de tablas.
 - «*outline*», «*outline-color*», «*outline-style*» y «*outline-width*» para contornos dinámicos.
- Aplicar propiedades auditivas que ayuden a usuarios con deficiencias visuales. CSS define las siguientes:
 - «*volume*»: Usada para controlar el volumen del texto hablado.

- «*speak*»: usado para especificar si el contenido se pronunciará y, en caso afirmativo, si se debe deletrear o leer como palabras.
 - «*pause*», «*pause-before*» y «*pause-after*»: usado para controlar las pausas antes y después de anunciar el contenido.
 - «*cue*», «*cue-before*» y «*cue-after*»: usado para especificar un sonido que se reproducirá antes y después del contenido.
 - «*play-during*»: usado para controlar los sonidos de fondo durante la presentación de un elemento.
 - «*azimuth*» y «*elevation*»: usados para especificar una dimensión al sonido usado, por ejemplo para distinguir las voces.
 - «*speech-rate*», «*voice-family*», «*pitch*», «*pitch-range*», «*stress*» y «*richness*»: usados para controlar las cualidades de los contenidos hablados.
 - «*speak-punctuation*» y «*speak-numeral*»: especifican la forma de decir los números y la puntuación.
 - «*speak-header*»: especifica cómo se debe decir la información sobre los encabezados antes de una celda de tabla.
- Se recomienda crear distintas hojas de estilo para adaptar la página web a los distintos dispositivos de salida para los que es diseñada (Braille, sintetizadores de voz o dispositivos TTY, pantalla, móvil, etc.). Para ello se usan los «tipos de medios» definidos en la versión 2 de CSS (empleados con las reglas *@media*). Las reglas *@media* también pueden reducir los tiempos de descarga porque permiten a las aplicaciones de usuario ignorar reglas inapropiadas.

Siguiendo estas técnicas se conseguirá que el sitio web desarrollado sea con total seguridad un sitio web accesible. Aunque no es obligatorio seguirlas, sí es recomendable, aunque en la práctica solo se aplicarán aquellas propiedades CSS que afecten al usuario para el que está desarrollada la página. No es habitual usar las propiedades que ayudan con el sonido si el usuario potencial del sitio web no tiene deficiencias visuales, por ejemplo.

1.3. Tipos de estilo: incrustados, enlazados, importados, en línea

Un desarrollador puede aplicar estilos en una página web de distintas formas. El formato de las reglas CSS no cambia, pero sí la forma de vincularlo al documento HTML y al elemento al que se quiere dar estilo.

1.3.1. Descripción de los tipos de estilo

Los estilos CSS pueden ser de tres tipos distintos en función de la forma en la que se usen:

- Estilos en línea (*inline*): con este método se inserta el estilo directamente dentro de una etiqueta HTML, usando para ello su atributo «*style*». No es la forma recomendada de hacerlo por su complejidad a la hora de localizar el elemento, pero sí es recomendable hacerlo de forma ocasional como solución a algún problema. Sin embargo, esta forma sí es utilizada muy frecuentemente en la maquetación de correos electrónicos, ya que los clientes de correo no soportan las hojas de estilos externas.
- Hojas de estilo internas: los estilos se incrustan dentro de la sección <HEAD> de una página web, entre la etiqueta <STYLE> de apertura y la correspondiente etiqueta </STYLE> de cierre. De esta forma ya se está separando el contenido (HTML) de la apariencia (CSS). Se suele usar esta solución cuando se trata de una página pequeña o se necesita tener todo el código en el mismo fichero. Se usa la propiedad «*class*» del elemento y el *nombre* de la etiqueta o el atributo «*ID*» para seleccionar los elementos afectados por la regla
- Hojas de estilo externas: en este caso las definiciones de las reglas están almacenadas en un archivo externo. Esta es la forma recomendada para desarrollar páginas web, ya que la separación entre el contenido de la página web y el aspecto visual favorece el desarrollo, su mantenimiento posterior, la reutilización del código, etc. Al igual que con las hojas de estilo internas, se pueden utilizar todos los tipos de selectores posibles.
- Hojas de estilo importadas: al igual que en el caso anterior, las hojas de estilos estarán almacenadas en un archivo externo pero se utiliza

la directiva «@import» para importarlas al documento HTML. Esta directiva se puede usar tanto dentro de un bloque <STYLE> como dentro de una hoja de estilos externa. Si se hace de esta última forma, lo ideal es añadirla al principio del archivo.

En todos ellos, la definición del par «*propiedad: valor;*» no difiere. Lo que sí es diferente es la forma de definir la regla o clase, ya que por ejemplo en los *estilos inline* no se definen reglas, sino que simplemente se definen las propiedades que se van a aplicar.

1.3.2. Enlazar una hoja de estilo externa a un documento HTML

En el punto anterior comentamos la posibilidad de añadir estilos almacenados en una hoja de estilos externa al propio documento HTML. Para que esto sea factible es necesario «enlazar» dicha hoja de estilos al documento.

Se usa la etiqueta <LINK> dentro de la sección «HEAD» para enlazar una hoja de estilos externa. Esta hoja de estilos CSS externa es un archivo de texto plano con la definición de todas las reglas de estilos con los selectores que se desean aplicar al documento. Este archivo de texto debe tener la extensión «.css».

A continuación podemos ver los atributos de la etiqueta <LINK>:

Atributo	Valor	Descripción
rel	<i>stylesheet</i>	Especifica la relación entre el recurso enlazado (la hoja de estilos CSS) y la página HTML.
type	<i>text/css</i>	Especifica el tipo de recurso enlazado
href	<i>url</i>	Especifica la URL del archivo CSS, ya sea relativa o absoluta, y puede ser externo o interno al servidor
media	<i>all, screen, print, braille, embossed, handheld, projection, speech, tty, tv</i>	Especifica el tipo de medio en el que se van a aplicar los estilos del archivo CSS. Los más usados son « <i>screen</i> », para indicar que los estilos se aplicarán en la pantalla del ordenador, y « <i>print</i> », para indicar que los estilos se aplicarán cuando la página se imprima. Otro usado frecuentemente es « <i>handheld</i> », que define el aspecto de la página cuando se visualiza en un dispositivo móvil

El siguiente ejemplo muestra cómo conseguir esta página usando una hoja de estilos externa.



Para ello se crea un archivo «*estilos.css*» situado en la misma carpeta que el archivo HTML:

```
@import url(estilo_normal.css) screen and (min-width: 600px);
```

La hoja de estilos «*estilos.css*» se vinculará usando la etiqueta <LINK> y aplicando una URL relativa, tal y como sigue:

```
#menu {float: right;}  
#menu ul{list-style: none;}
```

Así, el documento HTML contendrá el siguiente código:

```

{
    width: 200px;
}

<meta charset="utf-8">
<title>Prueba Estilos en Línea</title>
<link rel="stylesheet" type="text/css" href="estilos.css"
media="screen">
</head>
<body>
    <h1>Prueba estilos CSS en línea </h1>
    <p class="estilo-parrafo">
        Este texto está centrado y de color rojo
    </p>
</body>
</html>

```

1.3.3. Incrustar un estilo dentro de un documento HTML

Como se ha visto anteriormente, para incrustar estilos CSS dentro del propio documento HTML es necesario utilizar el elemento `<STYLE>`. Para que estos estilos estén disponibles en toda la página, los elementos `<STYLE>` deben estar dentro de la sección «HEAD».

A la etiqueta `<STYLE>` se le pueden añadir atributos al igual que a la etiqueta `<LINK>`. A continuación vemos algunos de ellos.

Atributo	Valor	Descripción
<i>media</i>	<i>all, screen, print, braille, embosed, handbeld, projection, speech, tty, tv</i>	Especifica el tipo de medio en el que se van a aplicar los estilos del archivo CSS. Los más usados son « <i>screen</i> », para indicar que los estilos se aplicarán en la pantalla del ordenador y « <i>print</i> » para indicar que los estilos se aplicarán cuando la página se imprima. Otro usado frecuentemente es « <i>handbeld</i> », que define el aspecto de la página cuando se visualiza en un dispositivo móvil
<i>scoped</i>	<i>scoped</i>	Especifica que los estilos se aplican solamente al elemento padre de este elemento y a elementos secundarios de ese elemento

<i>type</i>	<i>text/css</i>	Especifica el tipo de recurso enlazado
-------------	-----------------	--

Para conseguir la página anteriormente mostrada usando estilos incrustados será necesario definir las reglas de estilos en el propio documento HTML. Esta vez se opta por un *selector por nombre*, que aplicaría el estilo definido dentro a todos los elementos <P> que haya en el documento:

```
<style type="text/css" media="screen">
  p{
    color:red;
    text-align: center;
  }
</style>
```

El documento HTML definitivo tendrá el siguiente contenido:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Prueba Estilos en Línea</title>
  <style type="text/css" media="screen">
    p{
      color:red;
      text-align: center;
    }
  </style>
</head>
<body>
  <h1>Prueba estilos CSS en línea </h1>
  <p>
    Este texto está centrado y de color rojo
  </p>
</body>
</html>
```

1.3.4. Importar una hoja de estilo desde un documento HTML

Otra forma de trabajar con hojas de estilos externas es a través de una regla especial de tipo «@import». Este tipo de reglas tienen la peculiaridad de que deben preceder siempre a cualquier otro tipo de regla, con excepción de la regla «@charset», la cual se utiliza en una hoja de estilos para indicar la codificación de caracteres de esta.

La regla «@import» va seguida por una URL, que puede ser absoluta o relativa, y apuntar a una hoja de estilos interna o externa. Las siguientes declaraciones son equivalentes:

- «@import 'estilos.css';».
- «@import "estilos.css";».
- «@import url('estilos.css');».
- «@import url("estilos.css");».

Siguiendo con el ejemplo, el documento HTML importando la hoja de estilos «estilos.css» contendrá el siguiente código:

```
@import url(estilo_normal.css) screen and (min-width: 600px);

#menu {float: right;}
#menu ul{list-style: none;}

@media screen and (min-width: 450px) and (orientation: portrait)
{
    #menu li
    {
        float: right;
        margin: 0 0 0 .5em;
        border: 1px solid #00AAFF;
    }
}

@media screen and (min-width: 850px)
{
```


1.3.5. Importar una hoja de estilo, a través de un archivo con estilos

De la misma forma que se importa una hoja de estilos usando la regla «@import» dentro de un bloque <STYLE>, es posible importar otra hoja de estilos dentro de la propia hoja de estilos vinculada en el documento HTML. Esto suele ser muy útil en proyectos grandes divididos en módulos en los que cada uno de estos módulos tiene sus propios estilos definidos.

Así, por ejemplo, el archivo «*estilos.css*» podría contener la siguiente declaración «@import», que importará otra hoja de estilos llamada «*estilos_2.css*» usando una url relativa.

```
@import url('estilos_2.css') screen;
```

El archivo «*estilos_2.css*» contendría la definición de las reglas de estilo:

```
{  
    width: 200px;  
    text-align: center;  
}
```

Y el documento HTML contendrá el siguiente código:

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="utf-8">  
    <title>Prueba Estilos en Línea</title>  
    <link rel="stylesheet" type="text/css" href="estilos_2.css"  
media="screen">  
  </head>  
  <body>  
    <h1>Prueba estilos CSS en línea </h1>  
    <p class="estilo-parrafo">  
      Este texto está centrado y de color rojo  
    </p>  
  </body>  
</html>
```