



UT2. GUÍAS DE ESTILO

Parte 1

DISEÑO DE INTERFACES WEB

2º Desarrollo de Aplicaciones Web

Jose A. Sánchez

CIFP LA LABORAL

Tabla de contenido

1.	Objetivos.....	2
2.	Introducción	2
3.	Guías de Estilo	2
	Tipografías	3
	Colores.....	5
	Iconografía.....	5
4.	Lenguaje de Marcas HTML	6
	Tablas y Marcos.....	7
	A. Tablas	7
	B. Marcos.....	8
	HTML5	9
5.	Introducción al Lenguaje de Hojas de Estilo CSS	11
	Creando estilos.....	12
6.	Hojas de Estilo Externas.....	13

1. Objetivos

- a) Analizar y seleccionar colores y tipografías adecuados para la visualización en pantalla
- b) Utilizar marcos y tablas para presentar información de manera ordenada
- c) Identificar nuevos elementos y etiquetas en la versión HTML5
- d) Reconocer posibilidades de modificación de etiquetas HTML
- e) Valorar la utilidad de las hojas de estilo para conseguir diseños uniformes en todo el sitio web
- f) Analizar opciones para la presentación de información en documentos web
- g) Valorar la importancia de definir y aplicar guías de estilo en el desarrollo de aplicaciones web
- h) Crear y utilizar plantillas de diseño
- i) Definir estilos de forma directa
- j) Asociar estilos globales en hojas externas
- k) Identificar las propiedades de los elementos
- l) Crear clases de estilos
- m) Analizar y utilizar selectores básicos y avanzados
- n) Manejar y actualizar guías de estilo
- o) Analizar las propiedades de las hojas de estilo para texto y fuente
- p) Posicionar y flotar elementos en la página web utilizando herramientas oportunas
- q) Utilizar nuevas propiedades desarrolladas en versiones actuales del lenguaje para hojas de estilo CSS3

2. Introducción

Como podemos observar, los objetivos de esta UT son muchos, debido a que las guías de estilo, incluidas las hojas de estilo, así como el lenguaje de marcas que se emplea para usarlas son un tema muy amplio para el que se necesitaría aplicar casi un año entero para abordarlas de forma amplia.

Por todo ello, esta UT2, intenta mostrar una visión global de las guías de estilo, así como realizar un pequeño repaso sobre lenguaje de marcas (que ya se debió aprender en 1º, en concreto en el módulo profesional de Lenguaje de Marcas y Sistemas de Gestión de Información) y también sobre las hojas de estilo que se aplican hoy día en el desarrollo web. En concreto nos centraremos en HTML5 y CSS3 como tecnologías emergentes dentro del mundo del desarrollo web.

3. Guías de Estilo

Al igual que el mundo se rige por “ciertas normas”, como las normas de circulación, protocolos de actuación, etc. El desarrollo web y más concretamente, el diseño de interfaces también se rige por alguna de esas normas; lo que en la actualidad se llaman guías de estilo. La definición de estos criterios y normas, permiten que la página o sitio web tenga una apariencia mucho más uniforme y atractiva para el usuario que la visita.

Al igual que se crean guías de estilo para revistas y libros, también se realizan para el desarrollo de las interfaces web.

El objetivo claro de la creación de las guías de estilo no es otro que el de lograr un mensaje coherente, eficaz y correcto, ofreciendo al usuario una experiencia más personalizada durante el uso de la página o sitio web. Dicho esto, podemos identificar como **objetivo principal** de las guías de estilo como: establecer un estilo uniforme para todo el sitio web, consiguiendo un estilo homogéneo que favorece la navegación del usuario por el sitio. Desde el punto de vista del programador y diseñador, las guías de estilo favorecen el desarrollo del sitio web.

La definición de la guía de estilo siempre recae en las personas encargadas del diseño y programación de la interfaz web con la misión de lograr una mejor toma de decisiones tanto en el proceso de diseño como en la fase posterior de mantenimiento del sitio web.

La guía de estilo debe recoger aspectos como la introducción en la interfaz de fotografías e imágenes, la gama de colores, los tipos de letra, etc.

Desde ya hace algún tiempo, el gigante Google ha propuesto su propia guía de estilos para desarrolladores (<https://developers.google.com/style>) pudiendo acceder para ver cómo se empleará el diseño de la interfaz.

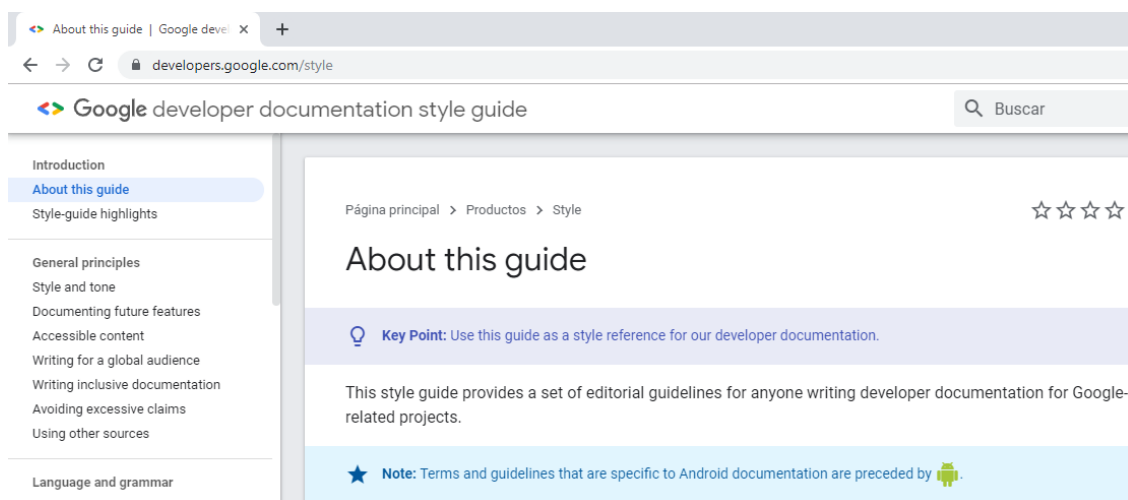


FIGURA 1. WEB GOOGLE PARA DESARROLLADORES (ESTILOS)

Tipografías

La tipografía, hace alusión al tipo de letra que se escoge para un determinado diseño. Normalmente, el uso de fuentes familiares mejora la calidad de la lectura. Es importante elegir un estilo funcional que permita la visualización en todos los navegadores, es decir, que esté soportado por ellos.

En casi cualquier editor de texto, se nos permite seleccionar la fuente que vamos a emplear. Al igual, que definimos el tipo de letra y la fuente a usar en la escritura de un documento, también lo tenemos que hacer en la definición de la interfaz de nuestra página web.

A la hora de elegir la tipografía hay que atender a diferentes aspectos:

- **Fuente** (por ejemplo, la “Arial” es soportada por todos los navegadores)
- **Estilo** (para no abusar de negritas, cursivas o subrayados, que incluso pueden confundir al usuario porque normalmente se usan para enlaces o hipervínculos)
- **Tamaño** (debiendo elegir un tamaño acorde al diseño de la página)
- **Color** (no usar colores arriesgados como por ejemplo una letra rosa en fondo rojo)



FIGURA 2. EJEMPLO PARA LEGIBILIDAD DE TEXTO

En los inicios del desarrollo web y diseño de la interfaz, se utilizaban lo que se llamaban **fuentes seguras**, que son aquellas que los usuarios tienen instaladas por defecto en sus dispositivos. Hoy día, y gracias a que la mayoría de los navegadores soportan la directiva *@font-face*, es posible utilizar casi cualquier tipografía a través de *Google Fonts* sin necesidad de que el usuario las tenga previamente instaladas. Así, diferentes fuentes como *TrueType* o *OpenType* pueden convertirse a los formatos convenientes para ser representadas.

Las recomendaciones más frecuentes a la hora de tener en cuenta para la elección de una tipografía son:

- No utilizar más de dos o tres tipografías en un mismo proyecto
- Usar fuentes claras
- Escoger buen contraste entre letras y fondo
- No utilizar líneas de texto demasiado largas
- Utilizar interlineado de 1,5 puntos adicional al tamaño de letra del cuerpo
- Evitar la justificación. Es recomendable alineación a la izquierda (discutible)
- No utilizar demasiados subrayados; los dejaremos para determinar hipervínculos

Colores

En los grandes sitios web, es habitual utilizar un color asociado a cada sección, de esta forma podemos identificar fácilmente en qué apartado se está. El uso de una escala de color coherente también facilita la navegación. En este punto, se debe ser coherente y usar siempre un color para lo mismo e intentar transmitir armonía y tener en cuenta la psicología del color, así como atender las necesidades del cliente que nos encarga el sitio web para atender a sus necesidades.

TABLA 1. COLORES Y SU SIGNIFICADO

Color	Significado
Azul	Comercio de confianza
Rojo	Amor – Pasión
Naranja	Diversión
Rosa	Moda
Verde	Naturaleza

Iconografía

Otro de los aspectos a destacar dentro del diseño de una interfaz web y de la concreción de una guía de estilo es la iconografía. En el desarrollo de una web usamos ciertos iconos que representan acciones asociadas a su función. En este punto, es aconsejable que se mantenga una apariencia común entre iconos.

Al utilizar iconos, evitamos el uso excesivo de textos. En ese principio se basa la elección de estos iconos, es decir, una buena elección es importante puesto que, si un usuario no es capaz de determinar su significado, no se habrá conseguido el propósito de ahorrarle tiempo en la visualización de la página y por tanto el icono pierde su funcionalidad.



FIGURA 3. EJEMPLO DE ICONOGRAFÍA DEL DISEÑO WEB

4. Lenguaje de Marcas HTML

Los sitios web están compuestos por páginas que están escritas en algún lenguaje. En la actualidad existe un creciente desarrollo del sector profesional relacionado con el diseño web, por lo que, comprender conocer y manejar correctamente un lenguaje de marcas asociado al desarrollo de páginas web es imprescindible.

HTML (*HyperText Markup Language*) es el lenguaje de marcado de hipertexto utilizado en las páginas web. Este tipo de texto presenta una forma estructurada y agradable, con hipervínculos que conducen a otros documentos y con posibilidad de insertar elementos multimedia (sonido, imágenes, videos, etc.) El fundamento de este lenguaje se basa en la especificación, en el texto, de la estructura lógica del contenido (título, párrafos, enumeraciones, definiciones, etc.). Sus características principales son:

- Sencillez
- No hay variables
- No se compila
- Se interpreta por el propio navegador
- A las instrucciones se les llama etiquetas
- Permite escribir hipertexto

La sintaxis de una etiqueta HTML es la siguiente:

```
<identificador_de_etiqueta> ... </identificador_de_etiqueta>
```

Definiendo una estructura de un documento HTML (página web) como la que sigue:

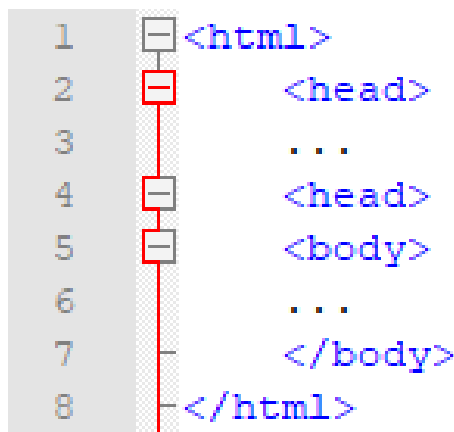


FIGURA 4. ESTRUCTURA HTML BÁSICA

Al no ser objetivo principal de este módulo presentar HTML como Lenguaje de Marcas (se debe conocer de módulos anteriores - LMSGI), sino usarlo para la definición de interfaces, se propone al alumnado que realice un repaso rápido del contenido de HTML desde la web <https://www.w3schools.com/html/>, donde se clarifica el uso de HTML en las páginas web, haciendo uso de ejemplos prácticos.

Se recomienda al alumnado que estudie lo referente a: Estructura de documentos HTML, Texto, párrafos, estilos y formatos, enlaces, inserción de imágenes y formularios.

Tablas y Marcos

Pese a que sea parte de HTML, conviene realizar un repaso a la definición de Tablas y Marcos dentro de este lenguaje de marcado.

A. Tablas

El uso de tablas nos permite mostrar la información de manera más organizada dentro de nuestro sitio web. Para la creación de una tabla habrá que definir las características de la misma y, a continuación, fila a fila, indicar el valor de cada uno de los elementos de cada columna, es decir, definir el contenido de cada una de las celdas que la componen.

La definición de un ejemplo de tabla se lleva a cabo de la siguiente forma, dando como resultado lo que se muestra a la derecha de la Figura 5 (sin borde):

```
<table>
  <tr>
    <td>1,1</td>
    <td>1,2</td>
    <td>1,3</td>
  </tr>
  <tr>
    <td>2,1</td>
    <td>2,2</td>
    <td>2,3</td>
  </tr>
</table>
```

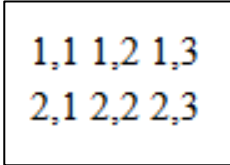


FIGURA 5. EJEMPLO DE ESTRUCTURA DE TABLA Y RESULTADO

En este ejemplo, no se ha añadido ninguna característica a la tabla por lo que no aparecen bordes ni ningún tipo de configuración en su apariencia. En la Tabla 2, se muestran los atributos principales de las tablas que podrán ser añadidos al ejemplo anterior para poder otorgar cierta apariencia a la misma.

TABLA 2. ATRIBUTOS PRINCIPALES DE TABLAS

Atributo	Descripción
Border	Indica grosor del borde de la tabla y las celdas. Por defecto es 0 (sin borde)
Width	Indica ancho de la tabla. Para designar este valor puede darse el tanto por ciento con respecto al tamaño de la página que tiene que ocupar, por ejemplo, el 100% si es el ancho completo de la ventana del navegador. También se puede dar en pixeles (px)
Align	Indica la alineación de la tabla: left, right o center

Ejercicio 1. Crea una página web HTML básica siguiendo la estructura estudiada anteriormente. Incluye en la página web una tabla añadiendo los atributos que consideres oportunos como la que se muestra a continuación:

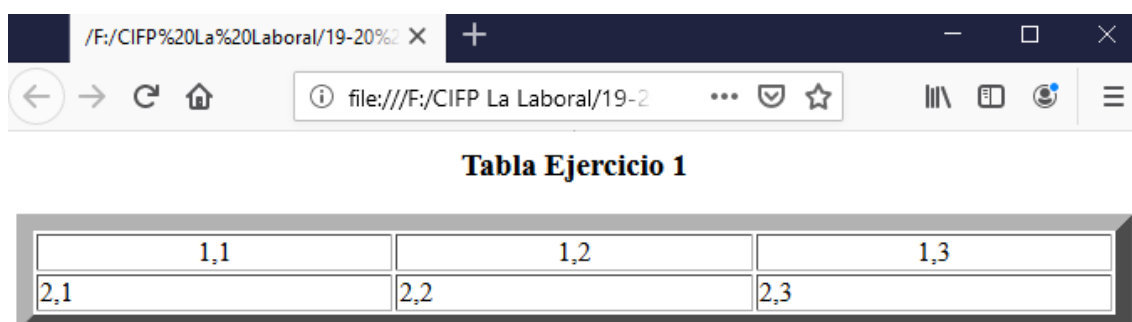


Tabla Ejercicio 1

1,1	1,2	1,3
2,1	2,2	2,3

En la imagen se muestra todo el navegador para que se vea la extensión de la tabla que se ha implementado debiendo parecerse lo máximo posible a la tabla que se muestra.

B. Marcos

Aunque en HTML5 el uso de Marcos está obsoleto, es interesante describir su funcionamiento puesto que, hoy día pueden encontrarse multitud de sitios web que se implementan haciendo uso de los mismos y los cuales puedan necesitar la definición de mejora.

Los marcos consisten en ventanas independientes que son incorporadas dentro de la página general, de tal forma que, cada página quedará dividida en varias subpáginas, lo que permite realizar diseños más organizados y limpios. En este tipo de definiciones, en lugar de tener las partes habituales (*head* y *body*), incorporamos un “**frameset**” en lugar del *body*.

Para llevar a cabo esto se utiliza la etiqueta `<frameset>` y a continuación la disposición del número de nuevas ventanas. En concreto se usa la forma:

```
<frameset cols=anchoColumna1, anchoColumna2, ..., anchoColumnaN>
    <frame ...>
    <frame ...>
</frameset>
```

De donde el primer elemento indica el número de columnas, así como su disposición en la página principal. La etiqueta `<frame>` se utilizará para definir las características que van a tomar cada uno de los marcos de forma individual.

De esta forma, quedan definidas las columnas, que darán lugar a los diferentes “frames” que componen la página web completa.

Además de la distribución por columnas, también se puede llevar a cabo la distribución por filas cambiando la palabra “*cols*” por “*rows*” o el uso de ambas dentro de una misma sentencia HTML para poder crear nuevas páginas web que contengan diferentes marcos de página.

En la Figura inferior (Figura 6) podemos observar una página web con dos marcos de página que definen la página web completa. En el interior, se puede observar la definición de marco y cómo observarlos en FrontPage (editor den páginas web del paquete de Microsoft Office)

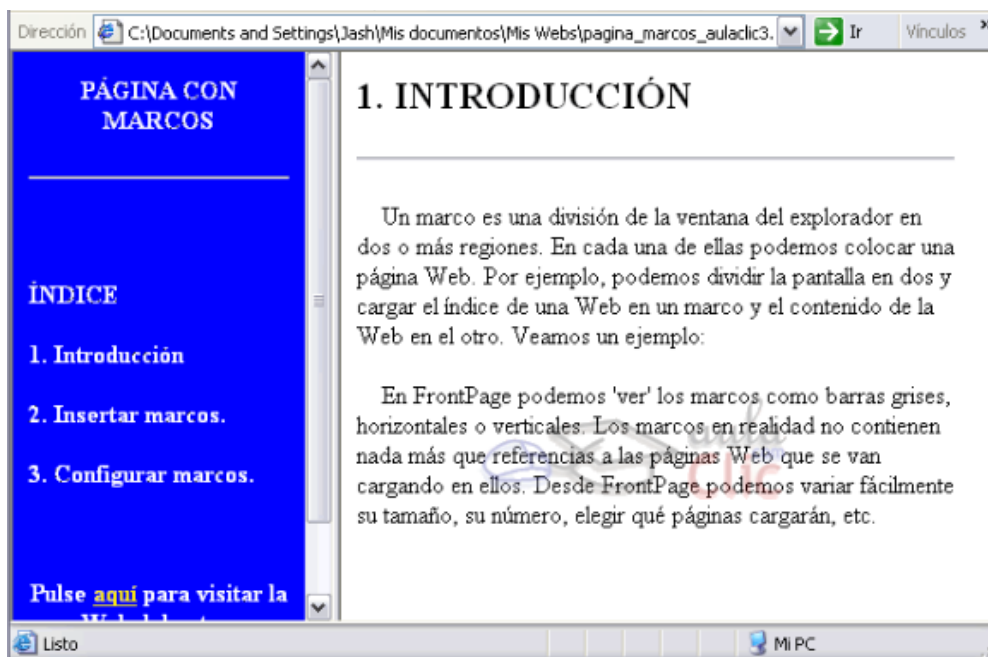


FIGURA 6. PÁGINA WEB CON MARCOS. FUENTE: AULACLIC

HTML5

HTML5 es la versión actual del lenguaje para la programación de páginas web HTML. Los sitios desarrollados en este lenguajes, únicamente se verán en navegadores actuales en los que esté soportado, puesto que las versiones anteriores no son capaces de interpretar algunas de las nuevas etiquetas que incorporan.

Una de las principales características de HTML5 son los elementos semánticos, también llamados marcados semánticos. Estos generan divisiones dentro del documento, pero, a diferencia de la habitual etiqueta de definición de divisiones `<div>`, si describen al navegador claramente su propósito.

En la siguiente figura (Figura 7) vemos algún ejemplo del cambio de HTML4 a HTML5 en la definición de las divisiones dentro de la página web.

Typical HTML4	Typical HTML5
<code><div id="header"></code>	<code><header></code>
<code><div id="menu"></code>	<code><nav></code>
<code><div id="content"></code>	<code><section></code>

FIGURA 7. CAMBIO HTML4 A HTML5 (ETIQUETA `<DIV>`)

A continuación, se definen los nuevos elementos semánticos en HTML5:

- `<article>` → Describe las unidades de contenido. Se pueden definir tantos elementos de este tipo con información vaya a contener la página web. Pueden estar subdivididas en más partes: cabecera (header), cuerpo (section) y pie de un artículo (footer).
- `<aside>` → Elemento que define la barra lateral de una página web. Puede contener cualquier tipo de contenido, normalmente utilizado para mostrar elementos que suponen un enlace hacia otros sitios web, por ejemplo, redes sociales, etc.
- `<footer>` → Equivale al pie de página. Se utiliza para colocar créditos del sitio web, datos de contacto, copyright, formularios de contacto, etc. Puede utilizarse de forma genérica para todo el sitio web o también para definir el pie de un elemento tipo `<article>`.
- `<header>` → Cabecera del sitio, define la parte superior de la página web. Suele mostrar algún elemento representativo de la web. En la mayor parte de los casos se sitúa el menú de navegación en ella (`<nav>`).
- `<main>` → Representa el contenido principal del cuerpo (body) de un documento. Consiste en el contenido directamente relacionado con el tema central del documento. Este contenido debe ser único para el documento. Únicamente podemos tener un elemento de tipo `<main>` en un documento y no debe descender de un elemento `<article>`, `<aside>`, `<footer>`, `<header>` o `<nav>`.
- `<nav>` → Crea un menú de navegación que enlaza con las diferentes partes del sitio web. Normalmente se sitúa justo debajo del encabezado o a la izquierda de la página.
- `<section>` → Etiqueta que sustituye a `<div>`. Se utiliza para crear diferentes secciones dentro de una página web. Estas secciones pueden contener a su vez varios elementos de tipo `<article>`.
- `<time>` → Representa la hora y la fecha en una página web.
- `<email>` → En formularios HTML5 incorpora nuevas etiquetas como `<email>` que permite crear un campo para introducir el correo electrónico y realiza una validación automática; de esta forma no es necesario implementar una función JavaScript del lado cliente para solventarlo.
- `<mark>` → Representa un texto resaltado, es decir, subrayado; de forma que queda marcada su relevancia en el contexto de la página web.
- `<figure>` → Representa contenido de carácter visual como: imágenes, ilustraciones, diagramas, fragmentos de código, etc. Este elemento se puede mover a otra página o a un apéndice sin que afecte al flujo principal.
- `<figcaption>` → Etiqueta que añade título a una figura. Se ubica anidado al elemento de la imagen (`<figure>`), al que va asociado.

- `<details>` → Mostrar información adicional de algún elemento de la página web. Su funcionamiento se basa en un “widget” que oculta y muestra información, similar al menú con efecto acordeón.
- `<summary>` → Esta nueva etiqueta incorpora un sumario de la etiqueta anterior `<details>`, añade información complementaria.

Ejercicio 2. Crea una página web HTML5 básica siguiendo una estructura adecuada y atendiendo a la inclusión de al menos 10 elementos semánticos que se hayan tratado anteriormente. No hace falta que la web contenga estilos, colores, etc. Bastará con realizar una prueba para comprobar el funcionamiento de HTML5 sobre el desarrollo y diseño web.

5. Introducción al Lenguaje de Hojas de Estilo CSS

Las hojas de estilo nos permiten centralizar y gestionar de manera bastante cómoda las instrucciones gráficas que tradicionalmente se insertaban en el fichero HTML. El lenguaje en el que se escriben es el CSS (*Cascading Style Sheet*) que controla todo el aspecto formal de una página o sitio web.

Cuando se trabaja con páginas web escritas en HTML y XHTML, es deseable que se presenten en un estilo concreto y con un estilo definido, para evitar elementos y figuras aleatorias dentro de la página web. La forma más eficiente para poder llevar esto a cabo es distinguir entre la parte en la que se sitúan los contenidos, los ficheros programados en lenguaje HTML y la parte del proyecto donde se recogen los aspectos relativos a la apariencia. Justo en este punto es donde aparece CSS, lenguaje de hojas d estilo que será el encargado de controlar el aspecto de los sitios web, como ya se comentó anteriormente.

El uso de hojas de estilo aporta ventajas como la mejora de la accesibilidad del documento, la reducción de la complejidad de su mantenimiento (tanto si se realiza de forma individual, como por terceros) y, además, facilita la visualización del sitio en dispositivos diferentes.

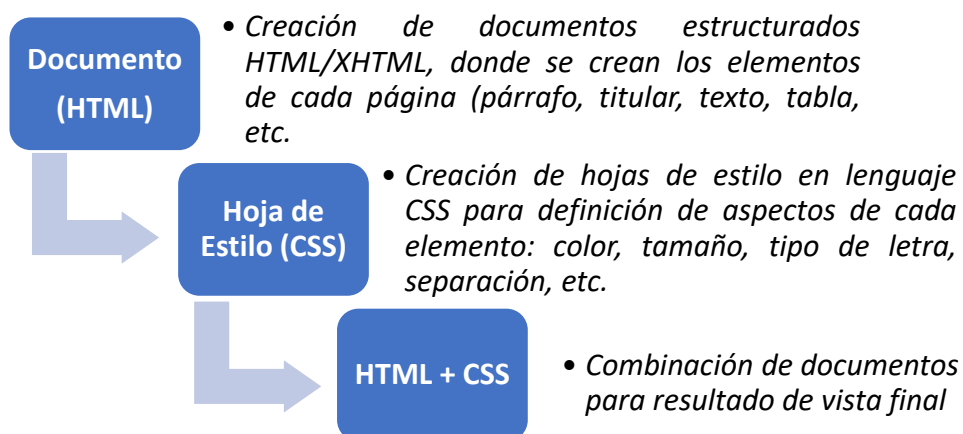


FIGURA 8. DIAGRAMA DE PROCESO PARA CONEXIÓN HTML Y CSS

Como vemos en la Figura 8, para la definición de la interfaz web final, será necesario el documento HTML (definición de página web) y la Hoja de Estilo CSS (definición del aspecto de la interfaz) para finalmente obtener una combinación que de como resultado la web final.

Como no todos los motores ofrecen el mismo soporte a las guías de estilo, es recomendable conocer el motor de cada navegador, dado que es la parte que se encarga de interpretar el código HTML y CSS. En la Tabla 3, se estudian los soportes de las diferentes versiones de CSS dentro de los principales navegadores que se utilizan en la actualidad.

TABLA 3. CARACTERÍSTICAS DE LOS MOTORES DE NAVEGACIÓN EN RELACIÓN A CSS

Navegador	CSS1	CSS2	CSS3	Motor
Firefox	Completo	Completo	Todos los selectores y pseudoclasas y ciertas propiedades	Gecko
Explorer	Completo	Completo	Soporta CSS3 y HTML5 a partir de su versión 10.0	Trident
Chrome	Completo	Completo	Todos los selectores y pseudoclasas y muchas propiedades	Webkit
Safari	Completo	Completo	Todos los selectores y pseudoclasas y muchas propiedades	Webkit
Opera	Completo	Completo	Todos los selectores y pseudoclasas y muchas propiedades	Presto

Ejercicio 3. Revisa los motores de cada uno de los navegadores e intenta observar las propiedades relacionadas con el manejo de hojas de estilo (CSS), así como otros aspectos relacionados con las versiones tanto del motor como del navegador al que se aplica, etc.

Creando estilos

Una vez visto qué es una hoja de estilo, sus usos y sus aplicaciones dentro de una página web, vamos a definir estilos dentro de un mismo documento HTML. En este punto, dejaremos para más adelante ([Punto 6](#) de este documento), la definición de hojas de estilo externas que nos servirán para dar aspecto a todo el diseño de nuestro sitio web.

Para añadir estilos dentro de una página web bastaría con utilizar la etiqueta `<style>` en el sitio indicado para ello en la cabecera de nuestro documento web. Esto puede servir como punto de partida para comenzar a familiarizarse con las hojas de estilo, pero se recomienda la creación de documentos diferenciados para la definición de la página o sitio web para facilitar el tratamiento de estas como ya se explicó anteriormente.

En la Figura inferior (Figuras 9) se puede observar, en primer término, el código de definición de un estilo simple para los párrafos y en segundo, la propia visualización de este contenido. Esto querrá decir que, todos los párrafos que establezcamos con la

etiqueta <p> dentro del documento HTML tendrán ese estilo (color rojo y fuente Verdana).

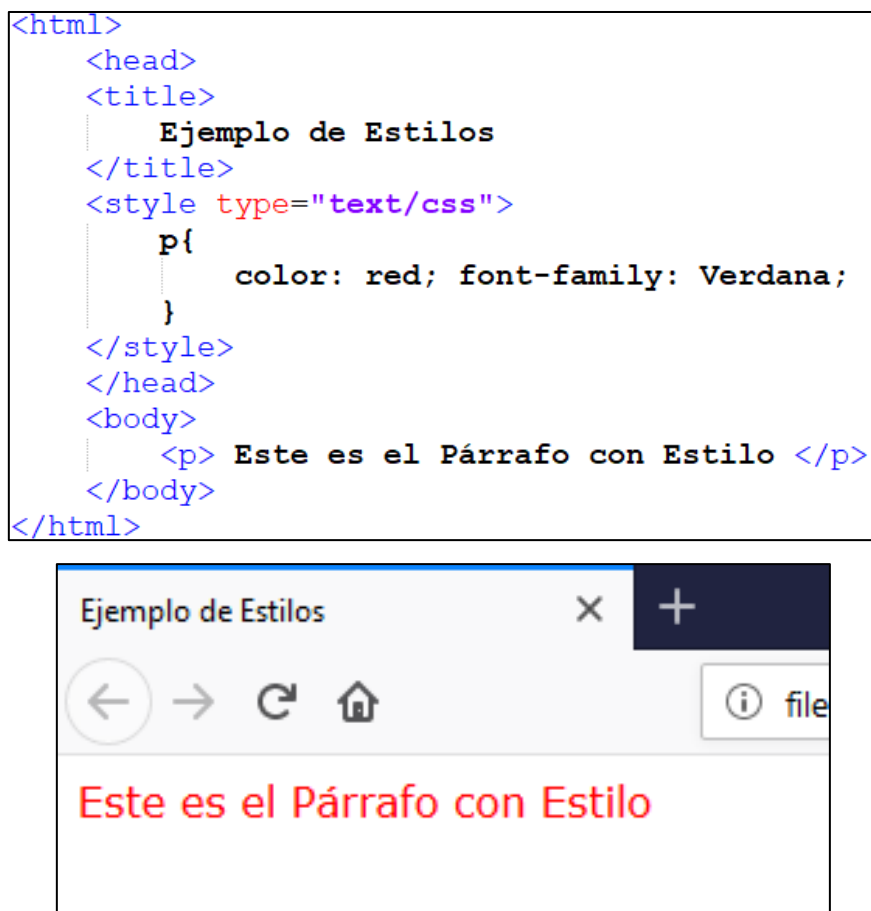


FIGURA 9. DEFINICIÓN DE ESTILO INCLUIDO EN PÁGINA HTML Y VISUALIZACIÓN DEL CÓDIGO

Ejercicio 4. Crea una nueva página web HTML a la que incorpores un estilo en el mismo documento (.html) para que las cabeceras de nivel 3 sean de color verde y tengan como fuente “Arial”

6. Hojas de Estilo Externas

Hoy en día, la mejor forma de crear hojas de estilo para las páginas o sitios web es tener hojas de estilo externas. Esto quiere decir que, en lugar de tener el código para modificar el estilo y aspecto del sitio web en el mismo documento que la definición de la página en HTML, lo tenemos en otro fichero (.css). Desde el fichero principal de la página desarrollada en HTML se accederá al fichero en código CSS, a través de la etiqueta <link>. Debemos tener en cuenta que el fichero donde se definen los estilos tendrá extensión .css y que se pueden establecer tantos ficheros CSS como se desee dentro de una web.

En la Figura 10, se pueden observar los dos ficheros creados para la definición de la página web creada en la Figura 9, atendiendo, en esta ocasión a los dos ficheros diferenciados (HTML y CSS).

Fichero: **Ejemplo.html**

```
<html>
  <head>
    <title>
      Ejemplo de Estilos con Ficheros Independientes
    </title>
    <link rel="stylesheet" type="text/css" href="guia.css" media="screen" />
  </head>
  <body>
    <p> Este es el Párrafo con Estilo </p>
  </body>
</html>
```

Fichero: **guía.css**

```
p{
    color: red; font-family: Verdana;
}
```

FIGURA 10. FICHEROS HTML Y CSS PARA DEFINICIÓN DE ESTILOS

En la Figura 11, podemos observar el resultado de aplicar este estilo, teniendo la misma visualización del contenido de la web que en la Figura 9 cuando lo definíamos en un mismo fichero.

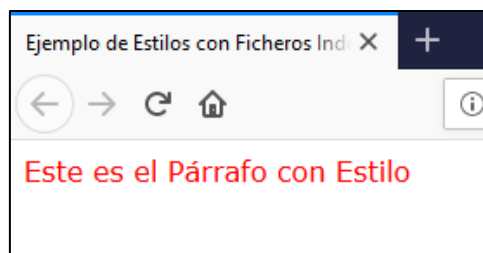


FIGURA 11. VISUALIZACIÓN DE PÁGINA FINAL

La función de cada uno de los atributos empleados para enlazar el documento HTML con la hoja de estilo en CSS es la siguiente:

- **rel** → Indica el tipo de relación que existe entre el recurso enlazado, el archivo CSS y la página HTML. Para archivos CSS el valor es *"stylesheet"*.
- **type** → Indica el tipo de recurso enlazado. En el caso de CSS su valor es *'text/css'*.
- **href** → Hace referencia a la URL del archivo CSS, esta puede ser relativa o absoluta y puede apuntar a un recurso interno o externo al sitio web.
- **media** → Indica el tipo de medio en el que se van a aplicar los estilos del archivo CSS. Más adelante (Parte 2 de la UT2) se explicarán en detalle los medios CSS.

Ejercicio 5. Replica el ejercicio 4, atendiendo a la creación de dos ficheros independientes; uno para la página web (HTML) y otro para la hoja de estilo (css)