

## TEMA 5.- LENGUAJE DE HOJAS DE ESTILO CSS

### 4.1. Introducción a CSS

Cuando se trabaja con páginas web escritas en HTML y XHTML, es deseable que presenten un estilo concreto y un aspecto definido. 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 HTML, y la parte del proyecto donde se recogen los aspectos relativos a la apariencia. Aquí es donde aparece CSS (Cascading Style Sheets), un lenguaje de hojas de estilo que será el encargado de controlar el aspecto de los sitios web. El uso de hojas de estilo CSS aporta numerosas ventajas: mejora la accesibilidad del documento, reduce la complejidad de su mantenimiento, tanto si va a ser realizado por uno mismo como por terceros, y, además, facilita la visualización del sitio en dispositivos diferentes.

1. Creación de documentos estructurados en HTML/XHTML, donde se sitúan los contenidos, es decir, donde se crean los elementos de cada página (párrafo, titular, texto destacado, tabla, lista de elementos, etc.).
2. Creación de hojas de estilo en lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, etc.
3. La combinación de estos documentos tiene como resultado la vista final del proyecto web.

No todos los navegadores ofrecen el mismo soporte a las guías de estilo. Debe conocerse el motor de cada navegador, dado que es la parte que se encarga de interpretar el código HTML y CSS. En el cuadro 4.1 se muestran los soportes de las diferentes versiones de CSS, que se estudiarán más adelante, en los principales navegadores que se utilizan en la actualidad.

Navegador	CSS1	CSS2	CSS3	Motor
Firefox	Completo	Completo	Todos los selectores y pseudoclases, y ciertas propiedades	Gecko
Explorer	Completo	Completo	?	?
Chrome	Completo	Completo	Todos los selectores y pseudoclases, y muchas propiedades	Webkit
Safari	Completo	Completo	Todos los selectores y pseudoclases, y muchas propiedades	Webkit
Opera	Completo	Completo	Todos los selectores y pseudoclases, y muchas propiedades	Presto

Para añadir estilos a una página web bastaría con añadirlos en el código de la propia página web, en el sitio indicado para ello en la cabecera, utilizando las etiquetas <style>. Esto puede utilizarse para comenzar a familiarizarse con las hojas de estilo, pero se recomienda realizarlo en documentos aparte, como se explica en el siguiente apartado.

```
<html>

<head>

  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859" -l>

  <title> Ejemplo estilos CSS en el propio documento </title>

  <style type ="text/css">

    P {

      color: red; font-family: Verdana;

    }

  </style>

</head>

  <body>

    <p> párrafo de texto </p>

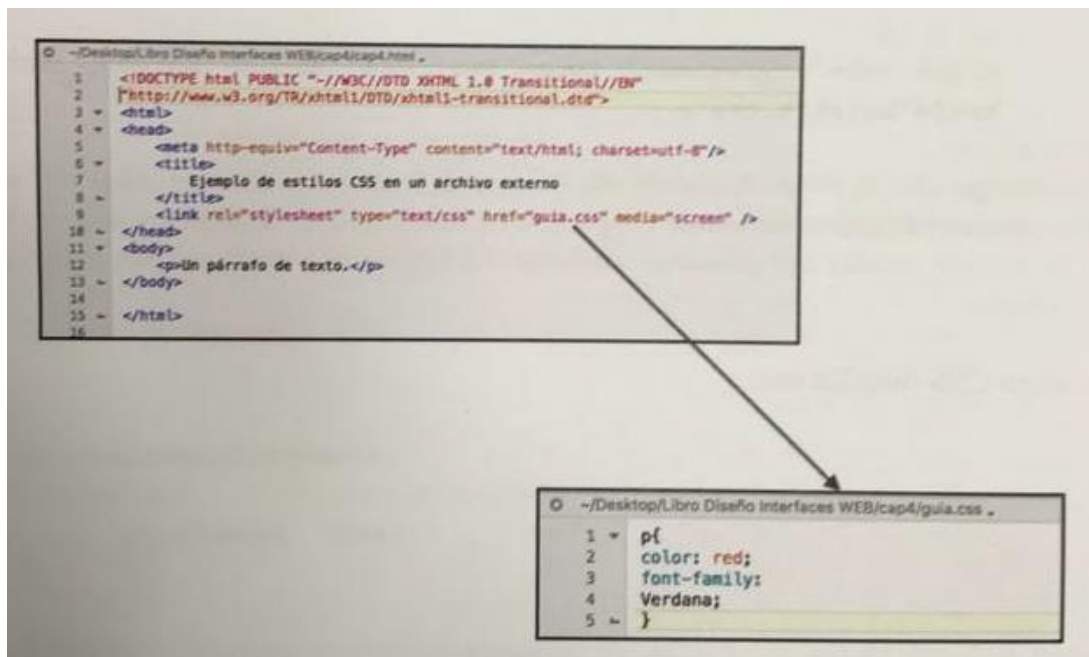
  </body>

</html>
```

El principal inconveniente de no utilizar los documentos CSS y hacerlo desde el propio código en HTML es que, si se quiere hacer una modificación en los estilos definidos, es necesario modificar todas las páginas que incluyen el estilo que se va a modificar.

## 4.2. Hojas de estilo externas

En la actualidad se utilizan hojas de estilo externas. En este caso, en lugar de tener el código encargado de modificar el sitio web, este código se encuentra en otro fichero. Desde el fichero principal de la página desarrollada en HTML se accede al fichero en código CSS, a través de la etiqueta la única particularidad que debe tenerse en cuenta es que el fichero donde se definen los estilos tendrá la extensión ".css". Se pueden enlazar tantos ficheros CSS como se deseen.



**<link rel="stylesheet" type="text/css" href="guia.css" media="screen"/>**

LA FUNCIÓN de cada uno de los atributos usados para enlazar el documento HTML con la hoja de estilos en CSS es:

- rel . Indica el tipo de relación que existe entre el recurso enlazado, el archivo CSS y página HTML. Para los archivos CSS, el valor es 'stylesheet'
- type,. Indica el tipo de recurso enlazado, en el caso de los archivos CSS su valor es 'text/ css'.
- href. Recoge 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 medio en el que se van a aplicar los estilos del archivo CSS. Más adelante se explican en detalle los medios CSS y su funcionamiento.

En la actualidad, el uso de CSS en HTML está muy extendido, puesto que a través de un solo archivo CSS es posible incluir una misma guía de estilo en todas las páginas HTML de un mismo sitio web; de esta forma se cumple uno de los principios más importantes perseguidos a la hora de realizar un correo diseño gráfico: que exista uniformidad de estilo entre todas las páginas que componen el sitio. Además, si se hace necesario realizar un cambio común en el estilo de todas las páginas, bastará con hacerlo en la hoja de estilo y se extenderá por todo el sitio web enlazado a este archivo CSS.

- Ejemplo:

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859" -l>

```

<title> Ejemplo estilos CSS en archivo externo </title>

<link rel="stylesheet" type="text/css" href="guia.css" media="screen"/>

</head>

    <body>

<p> párrafo de texto </p>

</body>

</html>

```

El código CSS (guia.css)

```

P {
    color: blue; font-family: Verdana;
}

```

### 4.3. Medios

Las hojas de estilo permiten especificar cómo se representa un documento, en función del tipo de medio de visualización. Se denominan medios a los canales a través de los cuales se muestra la información del sitio web: auditivos, visuales, impresos, etc. Lo habitual es que se utilice una propiedad definida para cada tipo de canal. En la actualidad, el medio más común es el utilizado para definir el aspecto del sitio web en pantalla, screen, y el que define la página para ser impresa. print. En el cuadro 4.2 se recogen los tipos más frecuentes.

Para especificar el medio de destino se utilizan varios mecanismos, desde la hoja de estilo en CSS o desde el propio fichero HTML del sitio web.

**CUADRO 4.2**

**Tipos de medios o canales a través de los cuales es posible visualizar el sitio web**

Medio	Descripción
<i>all</i>	Todos los medios definidos.
<i>print</i>	Impresoras.
<i>braille</i>	Dispositivos táctiles que emplean el sistema braille.
<i>embosed</i>	Impresoras braille.
<i>handheld</i>	Dispositivos de mano: móviles, PDA, etc.
<i>projection</i>	Proyectores y dispositivos para presentaciones.
<i>screen</i>	Pantallas de ordenador.
<i>speech</i>	Sintetizadores para navegadores de voz utilizados por personas discapacitadas.
<i>tv</i>	Televisores y dispositivos con resolución baja.

#### 4.3.1. @media

La regla '@media' especifica los tipos de medios de destino. A través de esta opción es posible que, en una misma hoja de estilo, aparezcan reglas para distintos medios. Si un mismo estilo se aplica a varios medios, basta con indicar sus nombres separados por comas.

- Sintaxis de @media

```
@media nombreTipoMedio {  
  
Selector { propiedad: valor;  
  
}
```

En el siguiente diagrama de código se muestra un ejemplo de uso de '@media' para indicar el tipo de medio donde se va a visualizar el sitio web. En este caso, se observan dos tipos de medio, impresión y visualización por pantalla. El tamaño de la letra se especifica para cada medio por separado, mientras que para el interlineado se va a utilizar el mismo en ambos casos; por lo tanto, se puede indicar de forma conjunta.

- Ejemplo:

```
@media print {  
  
h1{  
  
    font-size: 12pt  
    font-family: Arial;  
    }  
  
    @media screen {  
        h1 {  
            font-size: 50px  
            font-family: Yu Gothic;  
2eva  
        }  
    }  
  
    @media screen, print 1 {  
        h1 {  
            line-height: 1.2  
        }  
    }  
}
```

}

#### 4.3.2. @import

A través de @import es posible importar nuevas reglas desde otros ficheros CSS, e indicar el medio o medios sobre los que debe aplicarse el nuevo estilo. En este caso, a diferencia de lo que ocurría con '@media', en primer lugar, aparece la URL que apunta al fichero CSS que contiene la guía de estilo, y, a continuación, se indica el medio sobre el que se va a aplicar. Es importante tener en cuenta que estas reglas deben preceder a todas las demás reglas.

Si los datos de estilo del fichero se van a aplicar a varios medios al mismo tiempo, se indicará el nombre de estos separados por comas.

- Sintaxis de (import:

@import url(nombreFichero.css) nombreTipoMedio;

- Ejemplo de código:

```
@import "ficheroEstilo1.css" print;
```

```
@import "ficheroEstilo2.css. screen;
```

```
@import "ficheroEstilo3.css. print, screen;
```

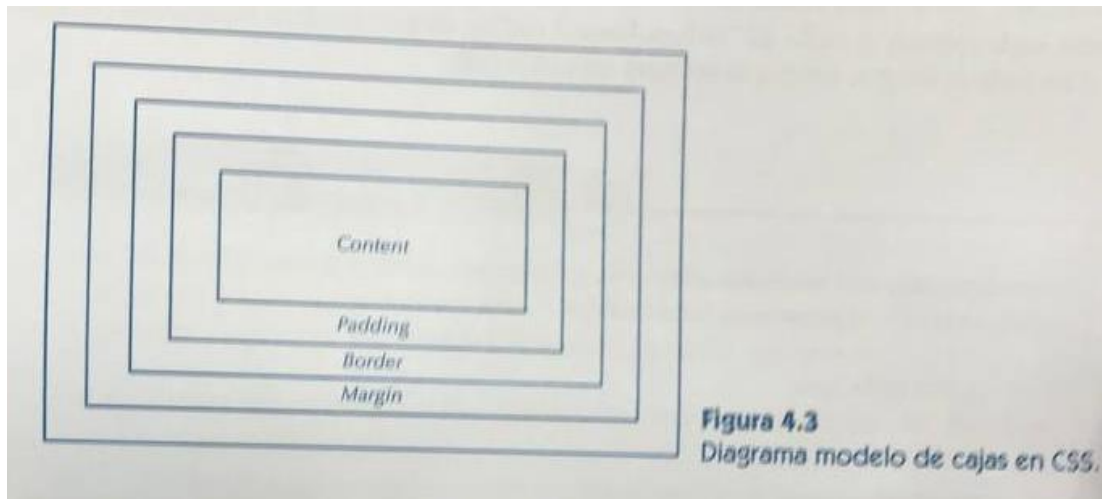
```
@import "ficheroEstilo4. css"; (Por defecto se aplica a todo)
```

En el ejemplo anterior, las dos primeras reglas aplicarán los estilos contenidos en los ficheros ficheroEstilo1.css" y "ficheroEstilo2.css", en medio impreso y por pantalla, respectivamente. La tercera regla aplicará el estilo de "ficheroEstilo3.css" en ambos medios. Finalmente, la última regla, al no indicar ningún medio, se aplicará en todos (all).

#### 4.4. El modelo de cajas

El modelo de cajas es una de las características más importantes de las hojas de estilo. Cuando se crea un página en HTML, el uso de cada etiqueta simula lo que podría considerarse una caja, dotando a cada una de ellas de un identificador, indicado por el nombre de la etiqueta. Este sistema de cajas permite, desde el fichero CSS, seleccionar cada caja y dotarle del estilo deseado.

El uso de las guías de estilo CSS permite modificar todas las características de cada una de las cajas creadas. A continuación, se analizan cada una de las partes a través de un ejemplo que irá incorporando nuevas propiedades.



Las propiedades principales con las que se trabaja en el modelo de cajas son `width` y `height` relativas a las dimensiones de la caja de contenido o `content`, se trata del ancho y el alto del área donde se muestra el contenido de la caja.

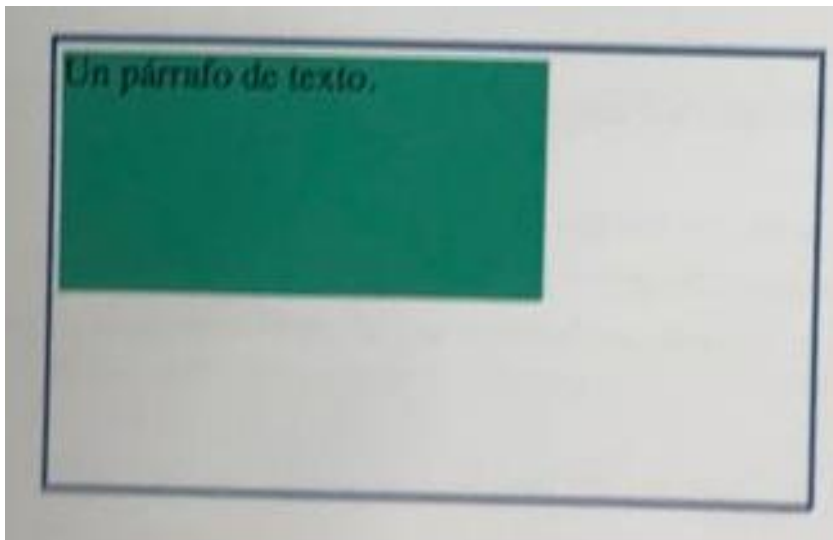
Otra de las propiedades más importantes del área de contenido es el fondo de la caja. Se diferencia entre `background image` (imagen de fondo), que corresponde a la imagen que se muestra por detrás del contenido y espacio de relleno, y `background color` (color de fondo), que es el color que se muestra por detrás del contenido y el espacio de relleno.

- Por ejemplo, si se introduce este código:

`article`

```
{  
    Background: #5DEAAB;  
    height: 100 px;  
    Width:200 px;  
}
```

- En pantalla, se verá:



La siguiente área que puede verse en el diagrama de cajas es la correspondiente al borde, llamado border. Esta zona es la que encierra el contenido y el relleno, content y padding. Es posible modificar su grosor, estilo y color; además, al igual que ocurre con padding, es posible dar formato a los cuatro extremos de la caja a la vez utilizando solo la palabra border o hacerlo de forma individual.

- Ejemplo:

```
article {  
border: 1px solid #000000;  
}
```

En aquellas ocasiones en las que se desee dar un formato diferente a cada uno de los extremos, se utilizarán algunas de las propiedades individuales destinadas a tal fin

Propiedades individuales	
<i>border-top</i> <i>border-right</i> <i>border-bottom</i> <i>border-left</i>	Modifican a la vez el grosor, el estilo y el color de cada lado del borde.
<i>border-width</i> <i>border-style</i> <i>border-color</i>	Modifican de forma individual el grosor, el estilo y el color de los cuatro extremos a la vez.
<i>border-top-width</i>	Modifica el grosor del borde superior.
<i>border-top-style</i>	Modifica el estilo del borde superior.
<i>border-top-color</i>	Modifica el color del borde superior



Finalmente, el margen que envuelve al resto de elementos CSS es el denominado **margin**, y sostiene a otras cajas del diseño. Las propiedades individuales son **margin-top**, **margin-right**, **margin-bottom** y **margin-left**. El funcionamiento es el mismo que el de la propiedad **padding**.

- Ejemplo:

```
article
{
margin-top: 10 px;
margin-right: 10 px;
margin-bottom: 10 px;
margin-left: 10 px;
}
```

## 4.5. Herencia

La herencia en CSS es el mecanismo mediante el cual determinadas propiedades de un elemento "padre" se transmiten a sus "hijos". No todas las propiedades CSS son heredables; los márgenes no se heredan porque es poco probable que un elemento "hijo" necesite los mismos márgenes que su "padre". Cuando se establece el valor de una propiedad CSS en un elemento sus elementos descendientes heredan de forma automática el valor de esa propiedad.

- Ejemplo:

```
<html>
<head>
    <meta http-equiv= "Content-Type" content="text/html; charset=iso-8859-1"/>
    <title>Ejemplo de herencia de estilos</title>
    <style type="text/css"
        body {color: blue; }
    </style>
</head>
<body>
    <h1>Titular de la página </h1>
    <p>Párrafo de texto</p>
</body>
</html>
```

En el ejemplo, el selector body solo establece el color de la letra para el elemento <body>. Esta propiedad se hereda de forma automática, ¿qué significa esto? Que los elementos descendientes muestran ese mismo color. Ahora bien, la herencia de estilos se aplica automáticamente, pero su efecto puede ser anulado estableciendo de forma explícita otro valor para la propiedad que se hereda.

En cambio, existen otras propiedades cuyo valor no se hereda, como es el caso de los bordes (border); es decir, si se define un estilo relativo al borde para el cuerpo, este no será aplicado al resto de elementos HTML.

### Resumen de las propiedades heredadas por defecto en HTML

color	text-align
font	visibility
letter-spacing	White-space
direction	Border-collapse
word-spacing	quotes
line-height	list-style-image

Ahora bien, como ya se ha dicho, existen ciertas propiedades que no son heredables "de padres a hijos", pero existe un mecanismo mediante el cual es posible indicar que un elemento, herede el valor de una propiedad. Para ello se utilizan los valores: inherit, initial y unset, que modifican el valor de la propiedad respecto a la herencia.

a) inherit. Aplica el valor de la propiedad del elemento "padre".

b) initial. Aplica el valor que presenta inicialmente la propiedad.

c) unset. Hereda el valor de la propiedad del elemento "padre"; si este no existe, aplica el valor inicial.

• Ejemplo:

```
body {  
border-width: 10px;  
border-color: blue;  
color:red;  
}  
h1{  
border:inherit;  
}
```

En este ejemplo, el elemento h1 hereda todas las propiedades establecidas en el body. El color se hace forma automática, puesto es una de las propiedades que se heredan, y al no establecerse otro valor para el color en h1 quedará de color rojo. Ahora bien, en este ejemplo

las propiedades de borde también serán heredadas puesto que así es indicado por el valor inherit.

#### Actividad propuesta

Reescribe el código anterior para que el elemento 'h1' sea de color verde. El resto de los elementos heredarán su valor de body.

## 4.6. CSS3

CSS3 (Cascading Style Sheets 3) es la tercera versión del lenguaje de hojas de estilo CSS. En este apartado se verán algunas de las novedades que presenta frente a su versión anterior, CSS2.1. Esta nueva actualización presenta todas las características de su predecesora y, además, aporta un mayor dinamismo en el diseño de la página web. En cuanto a los aspectos relativos a la programación, esta no varía; se sigue empleando la estructura de propiedades y selectores vista a lo largo del capítulo.

### A) Múltiples imágenes en el fondo

En esta nueva versión es posible poner varias imágenes de fondo en un mismo elemento. En el siguiente ejemplo, se puede ver cómo se utilizan varias imágenes para el fondo y, a continuación, de la URL de cada una de ellas indica las posiciones del background.

- Ejemplo:

```
#contenidoPrincipal {  
  width: 500px; height: 500px;  
  background:  
    url(imagen1.gif) bottom right no-repeat,  
    url(imagen2.jpg) center center no-repeat,  
    url(imagen3.jpg) repeat;  
}
```

### B) Nuevas propiedades para el fondo de los elementos

En CSS3 se incorporan nuevas propiedades que definen las características del fondo de los elementos. Estas son background-origin, background-clip y background-size.

- Background-origin especifica la posición de origen de un fondo o imagen.
- Background-clip permite definir el comportamiento de un fondo o imagen de fondo, con respecto al borde, si debe mostrarse por debajo de este o no. Esta propiedad se aplica a todas las cajas contenedoras conocidas: border-box, padding-box y content-box.

- Background-size permite definir el tamaño de las imágenes de fondo.

#### C) Esquinas redondeadas

Esta nueva propiedad permite redondear las esquinas de las áreas de la página, en concreto, a través del atributo 'border-radius'. Cabe tener en cuenta que, en función del navegador, el nombre del atributo varía: para Chrome y Safari se utiliza: '-webkit-border-radius', y para mozilla '-moz-border-radius'

Ejemplo:

```
div {  
border: 2px solid #ffffff;  
-moz-border-radius: 9px;  
-webkit-border-radius: 9px;  
}
```

#### D) Sombras

Esta versión incorpora una propiedad para crear sombras en los elementos, incluso en los textos. El atributo que se utiliza es box-shadow y está formado por cuatro valores: desplazamiento horizontal de la sombra, desplazamiento vertical de la sombra, difuminado y color de la sombra; estos valores aparecen siempre en el mismo orden.

- Ejemplo:

```
div {  
box-shadow: 5px -9px 3px .#000000;  
}
```

#### E) Transparencias de color

Otra de las nuevas incorporaciones en CSS3 es la posibilidad de añadir transparencia en el color; desde un color totalmente sólido hasta el transparente. La propiedad que define la transparencia se denomina 'opacity', y toma un valor decimal entre 0.0 y 1.0, donde 0.0 es la máxima transparencia y 1.0 la máxima opacidad.

- Ejemplo:

```
div {  
opacity: 0.5;  
}
```

#### F) Texto en varias columnas

En CSS3 se incorpora la opción de maquetar el texto en varias columnas a través de varios atributos: 'column-width' (define el ancho de las columnas), 'column-gap' (define el espacio en

blanco entre columnas) y `column-rule` (crea una línea entre las columnas). Aunque en la actualidad aún no son soportadas por ningún navegador, de manera experimental sí son implementadas por Safari, Google Chrome y Firefox. Al igual que en el caso de `border-radius`, se utiliza un prefijo para su uso en función del navegador en el que se usan: `-moz-` en Firefox Y `-webkit-` en Safari y Chrome.

- Ejemplo:

```
div{  
-moz-column-width: 20em;  
-moz-column-gap: 10px;  
-webkit-column-width: 20em;  
-webkit-column-gap: 10px;  
-webkit-column-rule: 2px solid #ffffff;  
-moz-column-rule: 2px solid #ffffff;  
}
```

## G) Animaciones

Una de las novedades más importantes es la creación de animaciones. Estas se crean directamente desde CSS3 sin la necesidad de utilizar otros lenguajes; además, con este tipo de implementación ya no será necesario tener instaladores plugins, que antes eran inevitables para la correcta visualización de las animaciones.

### Actividad propuesta

Diseña y escribe un programa en HTML y CSS3 que tenga tres columnas del mismo tamaño. Además, el color de fondo de la página debe ser azul con una transparencia del 30%.

## 4.7. Bootstrap

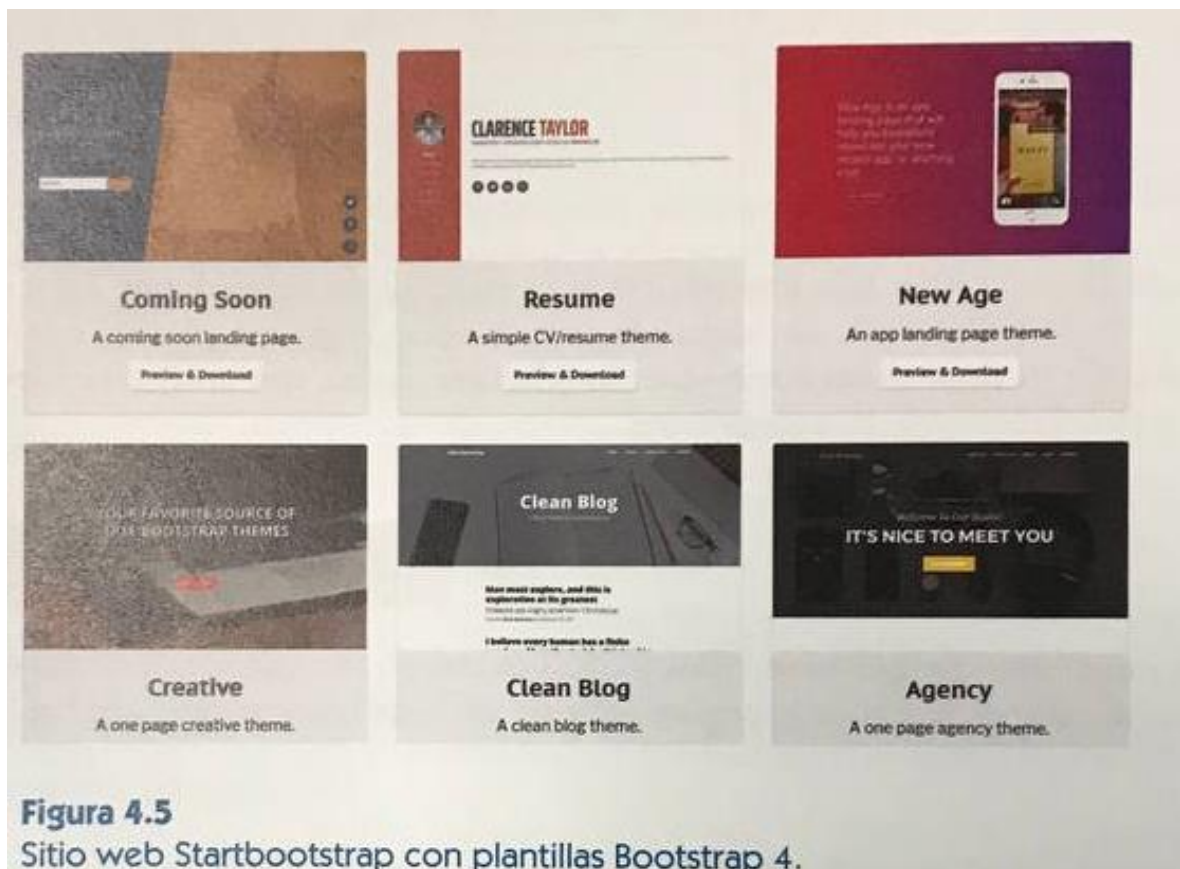
Bootstrap, creado por Twitter, constituye uno de los frameworks CSS más populares en la actualidad; esto es, plantillas CSS con gran capacidad de adaptación a distintos tipos de plataforma web mediante el uso de librerías CSS, que incluyen un amplio repositorio de tipográficas, botones y cuadros, entre otros. La versión actual es Bootstrap 4.x.

Bootstrap está en código abierto y se encuentra disponible en el primer enlace. Puedes acceder a las plantillas a través del segundo enlace.

1. <http://getbootstrap.com/getting-started/>

2. <https://startbootstrap.com/template-categories/a11/>

Esta herramienta permite crear un ilimitado conjunto de interfaces combinando todos los recursos que este framework pone al alcance de los usuarios. Bootstrap es compatible con la mayor parte de los navegadores más extendidos hoy en día: Google Chrome, Safari, Mozilla Firefox, Internet Explorer y Opera. Existen distintas plantillas sobre las que se pueden realizar los desarrollos y también se puede comenzar desde cero combinando todos los elementos que Bootstrap pone a disposición. junto al código disponible, pueden encontrarse diversas plantillas como las mostradas en la figura 4.5. Esta herramienta facilita el uso de este código, puesto que basta con tomar los elementos que se desean e incorporarlos después en la página web.



Por lo tanto, se puede resumir que Bootstrap es una hoja de estilo CSS, como las vistas hasta ahora, que puede aplicarse al documento de HTML. Para enlazar el código de un sitio web con el archivo contenedor de las reglas de estilo, 'bootstrap.min.css', se hace la misma forma que en el resto de los casos.

• Ejemplo:

```
<html>
```

```
<head>
```

```
<title>Prueba de Bootstrap</title>
```

```
<link href="css/bootstrap.min.css", rel="stylesheet,">
```

```
</head>

<body>

<div class="container" style="background-color: #cccccc"

<p>Esto es una prueba de bootstrap 4.</p>

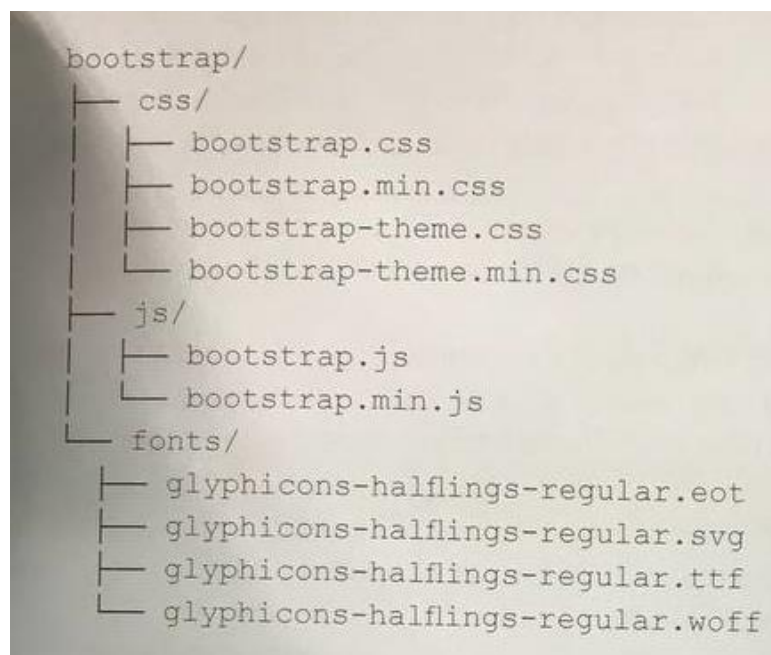
</div>

</body>

</html>
```

Utilizar Bootstrap en el diseño de un sitio web es muy sencillo. En primer lugar, desde su portal web se selecciona aquella plantilla que más adecúe el sitio que se está diseñando. Existen multitud de opciones de plantillas gratuitas; una vez que se ha encontrado la apropiada, se descarga en el equipo.

A continuación, se descarga el código CSS y JavaScript compilado del código fuente de la plantilla, que contiene todos los archivos Less, y JavaScript originales de Bootstrap. Tras descargar y descomprimir la estructura de ficheros es la siguiente:



Normalmente, para cada archivo se incluyen dos opciones: los archivos compilados o los archivos compilados y comprimidos. En las carpetas se diferencian los archivos Bootstrap.css, los cuales añaden estilos a formularios, tablas o imágenes; y archivos Bootstrap.js, que incluyen el código JavaScript para la creación de ventanas emergentes, alertas, efectos de carrusel, etc.

Tras enlazar el código del documento en HTML a la hoja de estilo CSS, como se ha explicado al inicio de este apartado, utilizando las etiquetas de cada uno de los elementos contenidos en la plantilla Bootstrap, será posible dar el formato deseado al sitio web, bien utilizando todos los elementos de la guía de estilo sin modificaciones, bien realizando una versión propia.

Una de las características principales que interesa conocer antes de comenzar a indagar entre el amplio abanico de posibilidades que ofrece esta herramienta, son los contenedores. Con-

viene reestructurar el fichero HTML en un conjunto de contenedores, denominados container, y otras variantes tales como container-fluid.

- Ejemplo: <div class="container">

.....

</div>