DISEÑO WEB CON HTML5 Y CSS3

MANUEL TORRES REMON



EDITORIAL MACRO

Esta publicación incluye:
Material de descarga

DISEÑO WEB CON HTML5 Y CSS3

MANUEL TORRES REMON

EDITORIAL MACRO

Perú - México - Colombia - Chile - Ecuador - España - Bolivia - Uruguay - Guatemala - Costa Rica

Diseño web con HTML5 y CSS3
Autor: Manuel Torres Remon

© Derechos de autor registrados:
Empresa Editora Macro EIRL

© Derechos de edición, arte gráfico y diagramación reservados:
Empresa Editora Macro EIRL

Jefe de edición:
Magaly Ramon Quiroz

Diseño de portada:
Fernando Cavassa Repetto

Diseño y diagramación:
Fernando Cavassa Repetto

Edición a cargo de:
© Empresa Editora Macro EIRL
Av. Paseo de la República N.° 5613, Miraflores, Lima, Perú

Teléfono: (511) 748 0560
E-mail: proyectoeditorial@editorialmacro.com
Página web: www.editorialmacro.com

Primera edición: Abril 2018
Tiraje: 2000 ejemplares

Impresión:
Talleres gráficos de la Empresa Editora Macro EIRL
Jr. San Agustín N.° 612-624, Surquillo, Lima, Perú
Abril 2018

ISBN N.° 978-612-304-555-5
Hecho el Depósito Legal en la Biblioteca Nacional del Perú N.° 2018-04923

Prohibida la reproducción parcial o total, por cualquier medio o método, de este libro sin previa autorización de la Empresa Editora Macro EIRL.

MANUEL TORRES REMON

Licenciado en Informática. Se ha dedicado a la consultoría y docencia de cursos de tecnología desde hace 17 años. A su vez, ha brindado diversos talleres de capacitación en las instituciones más importantes de Lima (Perú).

Realizó sus estudios en la Universidad Peruana del Norte, en la Universidad Alas Peruanas y en el Instituto de Educación Superior Manuel Arévalo Cáceres, en las cuales siguió las carreras de Ingeniería de Sistemas, Educación y Computación e Informática, respectivamente. En todas estas instituciones recibió una esmerada formación profesional que ha podido demostrar ampliamente a lo largo de su trayectoria docente.

Actualmente, se desempeña como docente de tecnología en instituciones de educación superior como Cibertec, Manuel Arévalo Cáceres y Unimáster de la Universidad Nacional de Ingeniería (UNI); en todas ellas imparte los cursos de Programación, Base de Datos y Análisis de Sistemas.

Este libro está dedicado con mucho cariño y aprecio a mis pequeñas Ángela Victoria y Fernanda Ximena, quienes son y seguirán siendo mi fuente de inspiración; y también a mi esposa Luz por comprenderme en todo lo que me propongo.

Agradecimientos

Todo libro escrito es un sacrificio que debe ser valorado, y ello solo se dará si usted aprecia este material tanto como yo mismo lo aprecio. Escuché alguna vez decir: «La lectura de un libro enriquece de conocimientos y empobrece la ignorancia». Esto es muy cierto, aunque hoy en día un libro impreso es difícil de leer. Son pocas las personas que cargan un libro en la mano, pues han sido reemplazados por los dispositivos móviles. Por ello, mi primer agradecimiento es para usted, amigo lector, por adquirir este material, que ha significado tiempo y esfuerzo para mí.

Asimismo, deseo agradecer a la gran familia Macro por confiar nuevamente en mi persona para el desarrollo de este libro.

Índice



Introducción 21

1 HTML5 23

1.1 **Generalidades** 24

1.2 **Historia del HTML** 25

1.3 **Versiones del HTML** 25

1.4 **Características del HTML5** 26

1.5 **Funcionamiento del envío de un documento web** 28

1.6 **Navegadores web** 29

1.7 **Determinar si mi navegador está preparado para HTML5** 31

1.8 **Editores HTML** 33

1.9 **Comprendiendo la URL** 39

1.10 **Servidores web locales** 40

1.11 **Publicación local y navegación *offline*** 43

Caso desarrollado: Archivo web con HTML5 44

2 Estructurando documentos para la web 53

2.1 **Estructura de una etiqueta HTML5** 54
2.2 **Etiquetas obsoletas para HTML5** 55
2.3 **Elementos que cambian su significado** 56
2.4 **Los atributos de una etiqueta HTML5** 60
2.5 **Identificación de los atributos en una etiqueta HTML5** 61
2.6 **Estructura global de un documento web** 64
2.6.1 Especificación DOCTYPE 64
2.6.2 Iniciando la estructura web con <html> 65
2.6.3 Definición de cabecera con la etiqueta <head> 65
2.6.4 La etiqueta <body> 69
Caso desarrollado: Maquetación de un documento web HTML 5 básico 69
2.7 **Estructurando el cuerpo del documento web** 72
2.7.1 La etiqueta <header> 75
2.7.2 La etiqueta <section> 78
2.7.3 La etiqueta <nav> 82
2.7.4 La etiqueta <aside> 86
2.7.5 La etiqueta <footer> 88
2.8 **Elementos internos** 89
2.8.1 La etiqueta <article> 89
2.8.2 La etiqueta <hgroup> 92
2.8.3 La etiqueta <figure> 94
2.8.4 La etiqueta <details> 96
2.9 **Comentarios en HTML5** 97
2.10 **Otras La etiquetas** 98
2.10.1 La etiqueta <hr> 98
2.10.2 La etiqueta <blockquote> 99
2.10.3 La etiqueta <cite> 100
2.10.4 La etiqueta <address> 102

3 Hojas de estilos en cascada CSS3 105

3.1 **Breve historia del CSS** 106

3.2 **Características del CSS3** 107

3.3 **Elementos de la definición CSS** 108

3.4 **Funcionamiento básico de CSS3** 108

3.5 **Integración CSS en HTML** 109
- 3.5.1 Estilo en línea 109
- 3.5.2 Estilo a nivel de página 110
- 3.5.3 Estilo externo 111
- 3.5.4 Estilo importado 112

3.6 **Los comentarios en CSS** 113

3.7 **Selectores CSS** 114
- 3.7.1 Selector universal 114
- 3.7.2 Selector de etiqueta 115
- 3.7.3 Selector en descendencia 117
- 3.7.4 Selector de clase 118
- 3.7.5 Selector de identificación 118

3.8 **Selectores avanzados** 119
- 3.8.1 Selector hijo 119
- 3.8.2 Selector adyacente 120

3.9 **Agrupación de reglas CSS** 120

3.10 **Modelos de cajas** 121
- 3.10.1 Unidades relativas 122
- 3.10.2 Unidades absolutas 123

3.11 **Márgenes y rellenos** 125
- 3.11.1 Elementos de bloque 125
- 3.11.2 Elementos de línea 126

3.12 **Colores en CSS** 126
- 3.12.1 Color de fondo 128
- 3.12.2 Color al texto 128
- 3.12.3 Color al borde 129

3.13 **Bordes en CSS** 129
3.13.1 Ancho de borde 129
3.13.2 Color del borde 130
3.13.3 Estilos de borde 131

3.14 **Márgenes en CSS** 133

3.15 **Espaciado en CSS** 135

3.16 **Textos en CSS** 136

3.17 **Alineación en CSS** 138

3.18 **Enlaces en CSS** 140

3.19 **Posicionamiento CSS** 142
3.19.1 Posicionamiento normal 144
3.19.2 Posicionamiento relativo 144
3.19.3 Posicionamiento absoluto 145
3.19.4 Posicionamiento fijo 145
3.19.5 Posicionamiento flotante 146
Caso desarrollado 1 : Estilo de línea 146
Caso desarrollado 2 : Estilo a nivel de página 148
Caso desarrollado 3 : Estilo externo 150
Caso desarrollado 4 : Estilo importado 152
Caso desarrollado 5 : Posicionamiento normal 154
Caso desarrollado 6 : Posicionamiento relativo 156
Caso desarrollado 7 : Posicionamiento absoluto 158
Caso desarrollado 8 : Posicionamiento flotante 160

4 Formateando un texto web 163

4.1 **Formato básico** 164
4.1.1 Las etiquetas de cabecera <h1>...<h6> 164
4.1.2 La etiqueta de párrafo <p> 166
4.1.3 La etiqueta de cambio de línea
 169
4.1.4 La etiqueta de preformateado de texto <pre> 170

4.2 **Etiquetas para fragmentos de textos** 171
4.2.1 La etiqueta 171
4.2.2 La etiqueta 173

4.2.3 La etiqueta <mark> 175
4.2.4 La etiqueta <i> 176
4.2.5 La etiqueta <b> 178
4.2.6 La etiqueta 180
4.2.7 La etiqueta <cite> 182

4.3 Etiquetas de modificación de significado 184
4.3.1 La etiqueta <sup> 184
4.3.2 La etiqueta <sub> 185
4.3.3 La etiqueta <small> 187
4.3.4 La etiqueta <q> 187
4.3.5 La etiqueta <dfn> 187
4.3.6 La etiqueta <abbr> 188

4.4 Etiquetas orientadas al código 188
4.4.1 La etiqueta <time> 188
4.4.2 La etiqueta <code> 189
4.4.3 La etiqueta <var> 190
4.4.4 La etiqueta <samp> 190
4.4.5 La etiqueta <kbd> 191

Caso desarrollado 1 : Uso de la etiqueta <header> y <h1>...<h6> 191
Caso desarrollado 2 : Uso de la etiqueta
 193
Caso desarrollado 3 : Uso de la etiqueta <pre> 196
Caso desarrollado 4 : Uso de la etiqueta <pre> para listado 198
Caso desarrollado 5 : Uso de la etiqueta 200
Caso desarrollado 6 : Uso de la etiqueta 202
Caso desarrollado 7 : Uso de la etiqueta <blockquote> y <cite> 204
Caso desarrollado 8 : Uso de la etiqueta <dfn> y <abbr> 206
Caso desarrollado 9 : Uso de la etiqueta <time> 209
Caso desarrollado 10 : Uso de la etiqueta <code> 212
Caso desarrollado 11 : Uso de la etiqueta <var>, <samp> y <kbd> 214
Caso desarrollado 12 : Uso de la etiqueta <sup> 217
Caso desarrollado 13 : Uso de la etiqueta <mark> 219

5 Listas 225

5.1 **Trabajando con listas** 226

5.2 **La etiqueta <ul> - lista desordenada** 227

5.3 **La etiqueta <ol> - lista ordenada** 229

Caso desarrollado 1 : Listas desordenadas 234
Caso desarrollado 2 : Listas desordenadas con viñeta cuadrada 236
Caso desarrollado 3 : Listas desordenadas con viñeta circular hueca 238
Caso desarrollado 4 : Listas desordenadas sin viñeta 240
Caso desarrollado 5 : Listas desordenadas con imagen 242
Caso desarrollado 6 : Listas ordenadas con numeración decimal 244
Caso desarrollado 7 : Listas ordenadas con numeración romana 246
Caso desarrollado 8 : Listas ordenadas con numeración romana descendente 248

6 Enlaces y navegación 251

6.1 **Introducción** 252

6.2 **Estructura de la URL** 253

6.3 **Enlaces básicos** 254
6.3.1 Enlazando a otra página web 254
6.3.2 Enlazando a un *e-mail* 255

6.4 **Tipos de URL** 259
6.4.1 URL absoluto 259
6.4.2 URL relativo 260
6.4.3 Diferencias entre URL absoluta y relativa 260

6.5 **La etiqueta <base>** 267

6.6 **Atributos de la etiqueta <a>** 267
6.6.1 Atributo accesskey 267
6.6.2 Atributo hreflang 268
6.6.3 Atributo rel 270
6.6.4 Atributo target 271
6.6.5 Atributo title 272
6.6.6 Atributo type 272

6.7 **Implementación de enlaces por anclas** 272

Caso desarrollado 1: Uso de la etiqueta <a> y <nav> 274
Caso desarrollado 2: Uso de la etiqueta <base> 276
Caso desarrollado 3: Proyecto web - Agencia de empleos 278

7 Imágenes, audio y video 295

7.1 **Seleccionando de manera correcta las imágenes** 296
7.1.1 Imágenes GIF 297
7.1.2 Imágenes JPG 298
7.1.3 Imágenes de alta calidad en el buscador Google 299
7.1.4 Imágenes PNG 301

7.2 **Agregando imágenes con la etiqueta <img>** 302
7.2.1 Atributo src 302
7.2.2 Atributo alt 303
7.2.3 Atributo height y width 303

7.3 **Usando imágenes como enlaces** 305
7.3.1 Definición de mapas 307
7.3.2 La etiqueta <map> 308
7.3.3 Asociación entre la imagen y los mapas 311

7.4 **Agregando videos de YouTube** 313

7.5 **Agregando audio y video con HTML5** 316
7.5.1 La etiqueta <video> 317
7.5.2 La etiqueta <audio> 322
7.5.3 La etiqueta <object> 324
7.5.4 La etiqueta <param> 325

8 Manejo de tablas para la web 329

8.1 **Introducción a las tablas web** 330

8.2 **Elementos básicos y atributos de una tabla** 332

8.3 **La etiqueta <table>** 333
8.3.1 Atributo border 336
8.3.2 Atributo bordercolor 340

8.3.3 Atributo width 342
8.3.4 Atributo summary 345
8.3.5 Atributo cellspacing 346
8.3.6 Atributo cellpadding 350
8.3.7 Atributo align 352
8.3.8 Atributo background 354
8.3.9 Atributo bgcolor 355

8.4 **La etiqueta <tr>** 359
8.5 **Las etiquetas <td> y <th>** 364
8.6 **Atributo rowspan** 366
8.7 **Atributo colspan** 369
8.8 **Atributo headers** 371
8.9 **Atributo scope** 373
8.10 **Dividir la tabla en <thead>, <tbody> y <tfoot>** 375
8.10.1 La etiqueta <thead> 375
8.10.2 La etiqueta <tbody> 375
8.10.3 La etiqueta <tfoot> 375

8.11 **Agrupando columnas - la etiqueta <colgroup>** 378
8.12 **La etiqueta <caption>** 380

9 Manejo de formularios para la web 383

9.1 **Introducción a los formularios web** 384
9.1.1 Formulario de contacto 384
9.1.2 Formulario de encuesta de satisfacción 385
9.1.3 Formulario de solicitud de empleo o registro de datos 386
9.1.4 Formulario de reservación 386

9.2 **Creando un formulario - etiqueta <form>** 388
9.2.1 Atributo action 389
9.2.2 Atributo method 389
9.2.3 Atributo id 390
9.2.4 Atributo name 391

9.2.5 Atributo enctype 391
9.2.6 Atributo accept-charset 392
9.2.7 Atributo novalidate 393
9.2.8 Atributo target 393
9.2.9 Atributo autocomplete 394

9.3 **Control de textos para formularios** 395
9.3.1 La etiqueta <label> 395
9.3.2 Cuadro de texto simple 397
9.3.3 Cuadro de texto múltiple 402
9.3.4 Cuadro de password 405
9.3.5 Entrada de teléfono 406
9.3.6 Entrada de correo electrónico 407
9.3.7 Entrada de color 408
9.3.8 Entrada de fecha 409
9.3.9 La etiqueta <number> 412
9.3.10 La etiqueta <URL> 415
9.3.11 La etiqueta <range> 416

9.4 **Control de botones para formularios** 417
9.4.1 Usando imágenes para botones 419
9.4.2 La etiqueta <output> 420

9.5 **Control de cuadros de chequeo** 421

9.6 **Control de botones de radio** 422

9.7 **Control de cuadro combinado** 423

9.8 **Control de cuadro de lista** 424

9.9 **La etiqueta <datalist>** 425

9.10 **Control de archivos** 429

Caso desarrollado: Formulario de registro 426

9.11 **Control de datos ocultos** 430

Referencias bibliográficas **431**

Introducción



Internet se ha vuelto hoy en día en un elemento muy importante dentro de la vida de cada persona. Es interesante averiguar cómo era la vida antes del uso de Internet: no había redes sociales, videos de YouTube, televisión digital, etc.; en cambio, se veía televisión, incluso, en blanco y negro, o se escuchaba música solo desde la radio AM o FM. Actualmente, el concepto no es muy diferente, pues se realizan las mismas actividades que antes, pero con la diferencia de que, ahora, se emplean elementos muchos más modernos y rápidos. Uno de esos elementos es la muy usada página web que también cambia su concepto inicial de solo informar por medio de textos e imágenes hasta llegar a ser un medio muy importante tanto en la vida personal como laboral.

Una de las personas que influyó en esa tendencia es Tim Berners-Lee, quien introdujo la idea del HTML al mundo, dándola a conocer en el año 1989 como un producto frente a las deficiencias de aquel entonces. Precisamente, Tim propone dos tecnologías para mejorar la experiencia en la web frente a los problemas de la búsqueda de información en Internet y la necesidad humana de mejorar lo que se tiene: el HTTP y el HTML, respectivamente.

Respecto a la primera tecnología, si se analiza el concepto de HTTP, este alude a la necesidad de transferir información de cualquier tipo por la web; de allí sus iniciales: *hyperText transfer protocol*. El HTTP es un protocolo de transferencia usado para computadoras catalogadas como servidores web, estas reciben una petición de usuario como, por ejemplo, visualizar una determinada página web. Se observa que siempre comienza con http://, pues se hace referencia al protocolo que se usará cuando se haga la

petición al servidor web, dicho protocolo convierte y muestra un documento HTML en su navegador web cualquiera que este sea, como Internet Explorer, Google Chrome, Mozilla Firefox, etc.

La segunda tecnología propuesta es el HTML, que representa un lenguaje de marcado basado en etiquetas, que le indica al navegador web cómo debe mostrarse el documento web solicitado por un determinado usuario. Esta tecnología abrió el camino a lo que hoy se conoce como «navegar por la web», pues contiene etiquetas que permiten configurar una navegación fluida entre muchos documentos web que se encuentran en Internet. La unión de estas tecnologías hace que la experiencia en la Internet sea rápida y sencilla, haciendo que el usuario navegue lo más cómodo posible.

Tim Berners-Lee funda el W3C, llamado así por las iniciales de Consorcio World Wide Web, que permitió a los usuarios de todo el mundo guiarse mejor sobre el uso del HTML y que ofreció cuatro versiones iniciales y un nuevo estándar, llamado HTML5, que permitirá revolucionar la experiencia web en el planeta. En este libro se explicará las nuevas especificaciones que presenta HTML5 y que más adelante el lector podrá desarrollar mediante guías prácticas, desarrolladas paso a paso, usando las aplicaciones estándar y de mayor demanda entre los diseñadores web.

Ahora bien, se detallará brevemente en qué consiste cada capítulo. En el primer capítulo se hace referencia a los conceptos iniciales del HTML, su funcionamiento y la publicación de documentos web de forma local. En el segundo capítulo, el libro se enfoca en la estructura del documento web, es decir, se desarrolla el uso de las etiquetas HTML5 en la implementación de un documento web. En el tercer capítulo se reconocen todos los estilos que se pueden integrar con HTML5 para mejorar el aspecto del documento web. En el cuarto capítulo se aborda sobre la presentación que pueden tener los textos y la importancia que esta tiene en un documento web. En el quinto capítulo, se hace uso de las listas que permitirán ordenar los elementos dentro de un documento web. En el sexto capítulo se trata sobre cómo aplicar los enlaces web en todas sus formas. En el séptimo, se hace uso de las etiquetas multimedia para convertir la página web en un elemento mucho más interactivo con el usuario. En el octavo capítulo se incorporan las tablas como elemento principal en el diseño de un documento web. Finalmente, en el capítulo nueve se trata sobre todos los elementos de los formularios web.

CAPÍTULO I

HTML5



Competencias

- Reconoce los principales conceptos del HTML5.
- Implementa un sitio web haciendo uso de la aplicación XAMP.

Contenido

1.1 Generalidades
1.2 Historia del HTML
1.3 Versiones de HTML
1.4 Características del HTML5
1.5 Funcionamiento del envío de un documento web
1.6 Navegadores web
1.7 Determinar si mi navegador está preparado para HTML5
1.8 Editores HTML
1.9 Comprendiendo la URL
1.10 Servidores web locales
1.11 Publicación local y navegación offline
- **Caso desarrollado:** Archivo web con HTML5

1.1 Generalidades

Cuando se habla de HTML, al instante se lo asocia a una página web, pero esto se encuentra un poco lejos de la realidad, pues cuando se realiza una navegación web se hace uso de un documento web que está siendo interpretado por un servidor que transforma las etiquetas HTML en algo legible para el usuario. Según lo anterior, se puede decir, entonces, que un diseñador de páginas web utiliza el lenguaje HTML para la implementación de sus páginas. Los diseñadores web usan ciertos programas que generan páginas basadas en etiquetas HTML, luego están los navegadores web, los cuales permiten visualizar la interpretación de cada etiqueta, generando así un proceso de navegación entre páginas web. Todo este proceso será esclarecido en el presente libro, pues el propósito de esta obra es que el lector genere su propio documento web y pueda realizar las pruebas correspondientes en un navegador web.

En ese sentido, se debe considerar que el HTML fue diseñado bajo la necesidad de mostrar información por medio de Internet. Hoy en día esa necesidad ha hecho que la mayoría de dispositivos que muestran información web tengan un contenido HTML. Este lenguaje se ha convertido en un estándar a nivel mundial, pues todo documento web debe estar basado en sus etiquetas. Es posible embeberse con otras tecnologías, pero siempre estarán presentes sus etiquetas. Existe un consorcio que avala el reconocimiento que tiene HTML, dicha organización, denominada W3C o también llamada World Wide Web Consortium, está considerada como un organismo sin fines de lucro que define las normas sobre el uso de las etiquetas HTML.

¿Sabías que...?

El Consorcio WWW o, en inglés, World Wide Web Consortium (W3C) es un consorcio internacional que genera recomendaciones y estándares que aseguran el crecimiento de la World Wide Web a largo plazo. Este consorcio fue creado en octubre de 1994 y está dirigido por Tim Berners-Lee, el creador original del URL (*Uniform Resource Locator*, 'localizador uniforme de recursos'), del HTTP (*HyperText Transfer Protocol*, 'protocolo de transferencia de hipertexto) y del HTML (*Hypertext Markup Language*, 'lenguaje de marcado de hipertexto'), que son las principales tecnologías sobre las que se basa la web.

Fuente: <https://es.wikipedia.org/wiki/World_Wide_Web_Consortium>

Con respecto a Tim Berners-Lee, se puede decir que es considerado el padre de la web, ya que sus estudios permitieron el uso del protocolo HTTP, el cual mueve la información en Internet. Ante la evolución del HTML, funda en 1994 el Consorcio de la World Wide Web (W3C), esta se encarga de supervisar y estandarizar el desarrollo de las tecnologías sobre las que se fundamenta y rigen el mundo de la web.

1.2 Historia del HTML

Como ya se mencionó, el HTML es un lenguaje de definición de etiquetas. En sus inicios, por el año 1991, se creó con la finalidad de mostrar información de forma remota entre la comunidad de científicos, pues Tim Berners es considerado uno de ellos; así, este logró desarrollar el HTML cuando se encontraba laborando en el CERN (también denominado European Organization for Nuclear Research, 'Organización Europea para la Investigación Nuclear), en esta primera versión se presentaron 22 etiquetas de las cuales entre 11 y 12 aún se siguen usando en la versión actual.

Con el fin de establecer un estándar se crea la IEFT (Internet Engineering Task Force, 'Grupo de Trabajo de Ingeniería de Internet') en el año 1993, el cual definió la gramática que hoy presenta el HTML. En esos años se presentan bosquejos para el lenguaje de etiquetas, que tienen como resultado al HTML+. Luego, en el año 1994, presentan el HTML Working Group, el cual estandariza el uso del HTML con la versión 2.0 y 3.0. Una de las principales características en esta evolución es la incorporación de etiquetas para el manejo de tablas en un documento web.

A inicios del año 1997 se lanza la versión 3.2 del HTML bajo una nueva organización llamada W3C (World Wide web Consortion) fundada por Tim Berners, dicha organización es la encargada de generar recomendaciones y estándares que aseguran el uso correcto del HTML en el World Wide Web. A finales de ese año se lanza la versión 4 llamada HTML4, la principal característica de esta versión es la variante del lenguaje que presenta la forma transaccional, estricta y de marcos. En la variante transaccional se permitía el uso de etiquetas obsoletas dentro del lenguaje, mientras que en la estricta no, y la forma de marcos dividía la página web en sectores llamados *frames*.

Finalmente, en el año 2004 se crea la WHATWG (Web Hypertext Application Technology Working) para la implementación de una nueva tecnología basada en las etiquetas HTML5 y la incorporación de funcionalidades para elementos multimedia como los videos, audios, imágenes vectoriales, contenido 2D y 3D dentro de los documentos web.

1.3 Versiones del HTML

HTML no cuenta con muchas versiones, ya que su objetivo de implementar documentos web mediante etiquetas es su esencia principal y esta aún se mantiene. La diferencia entre las versiones es mínima, a excepción de la versión HTML5, la cual cambia lo tradicional por una tecnología moderna enfocada a los dispositivos móviles. Véase una lista de las versiones del HTML y una breve especificación:

- **HTML 2.0**

  Es considerada la primera versión oficial. Es aquí donde se formalizan todas las versiones pasadas del HTML. Esta versión se encuentra basada en el estándar del HTML llamado SGML, el cual contiene un lenguaje limpio enfocado en un alto porcentaje a la funcionalidad y no al diseño, tanto es así que no presenta un soporte para tablas.

- **HTML 3.0**
  Versión desarrollada a principios del año 1996 en conjunto con empresas comerciales como IBM, Microsoft, Netscape, Novell, SoftQuad, Spyglass y Sun Microsystems. Entre las principales características de esta versión se tiene la incorporación de etiquetas para tablas y microaplicaciones. Además, presenta una mejora en el aspecto de las imágenes, puesto que es posible colocar texto alrededor de ellas.

- **HTML 4.0**
  Es la versión que se mantuvo mayor tiempo en comparación con las demás versiones. Esto se debió principalmente a la incorporación de nuevos mecanismos como las hojas de estilo en cascada, el soporte para la ejecución de scripts como JavaScript, el manejo de marcos también llamados *frames*, el soporte mejorado para texto, la ampliación de nuevas características para el manejo de tablas y el soporte para el manejo de formularios.

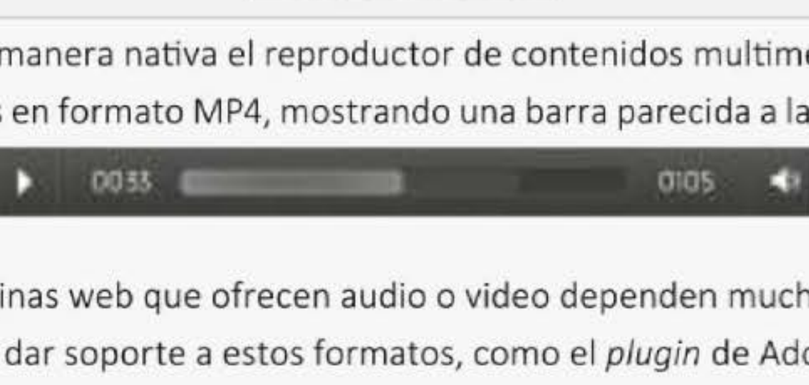
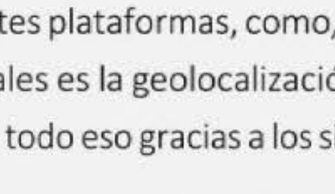
- **HTML5**
  Esta nueva versión define los nuevos estándares de desarrollo web. Propone un nuevo concepto sobre los documentos web, ya que no solo representa la actualización de las etiquetas HTML4, sino que muestra una nueva tecnología basada en la integración de los documentos web y elementos multimedia de manera profesional.

1.4 Características del HTML5

HTML5 se presenta como una propuesta moderna de la cual se plantearon ciertas reglas al momento de su implementación. A continuación, algunas de ellas:

- Su tecnología debe estar basada en HTML, CSS, DOM y JavaScript.
- Se debe incorporar la funcionalidad para elementos multimedia como audio, video e imágenes para dejar de lado el uso de Flash.
- Se deben agregar las funcionalidades para el tratamiento de errores como lo hacen los lenguajes de programación.
- Se deben incorporar etiquetas que reemplacen el uso de bloques de código para reducir así la cantidad de código en un documento web.
- Se debe ampliar la tecnología para que su uso no se limite solo a los documentos web, sino también a todos los dispositivos móviles.
- Todas las etiquetas dedicadas a la presentación del documento serán reemplazadas totalmente por el uso de hojas de estilo CSS.

Entonces, entre las principales características que presenta el HTML5 se tiene:

ESTRUCTURA DE LA WEB	HTML5 presenta etiquetas nuevas y usa algunas de sus versiones anteriores que permiten crear un mejor documento web, mucho más ordenado para el diseñador y con mayor rapidez para el navegador, logrando así un aumento del dinamismo en la página web. HTML5 no promueve el cambio total de las etiquetas, sino más bien implementa otras nuevas dejando obsoletas solo algunas etiquetas del HTML4. 
GRÁFICOS CON CANVAS	Canvas permite generar cualquier tipo de representación gráfica. Es un elemento incorporado en HTML5 que permite generar imágenes dinámicas por medio del script de JavaScript. Lo más resaltante de Canvas es la interacción que puede realizar el usuario frente a un documento web. 
CONTROL DE AUDIO Y VIDEO	HTML5 incorpora de manera nativa el reproductor de contenidos multimedia como sonidos en formato MP3 y videos en formato MP4, mostrando una barra parecida a la siguiente imagen: La mayoría de las páginas web que ofrecen audio o video dependen mucho de la instalación de *plugins* que permitan dar soporte a estos formatos, como el *plugin* de Adobe Flash. Este último era muy requerido por los diseñadores de páginas web, pero esto cambió completamente hasta la salida del HTML5, ya que hoy se permite ver videos o escuchar canciones *online* sin tener que descargarlas, solo manejando un reproductor de audio o video.
MANEJO DE GEOLOCALIZACIÓN	HTML5 está desarrollado para diferentes plataformas, como, por ejemplo, para la web, los móviles, etc. Una de las características principales es la geolocalización que permite determinar desde qué lugar se está visualizando un sitio web y todo eso gracias a los sistemas de referenciación como el GPS. Esto permitirá al usuario tener una rápida ubicación de la empresa que promociona la web, dando confianza al cliente al momento de realizar, por ejemplo, un *e-commerce*. 

BASE DE DATOS LOCALES	Se incorpora la tecnología de base de datos de forma local con la finalidad de navegar por una web sin necesidad de estar conectado a Internet. Este trabajo es conocido como *offline*.
WEB WORKERS	Se incorporan las API que permitirán trabajar en segundo plano los procesos que requieren mucho tiempo de procesamiento.
FORMULARIOS	Se incorporan nuevas funcionalidades para el manejo de formularios, como mostrar un calendario, una fecha, una hora, etc. Por ejemplo, incorpora la etiqueta `<datalist>`, el cual define una lista de opciones predefinidas para los controles de entrada en un documento web.

1.5 Funcionamiento del envío de un documento web

Véase cómo es el funcionamiento correcto para solicitar información de un documento web. Esto comienza desde la petición un usuario, al cual llamaremos cliente, hasta que lo recibe el servidor web.

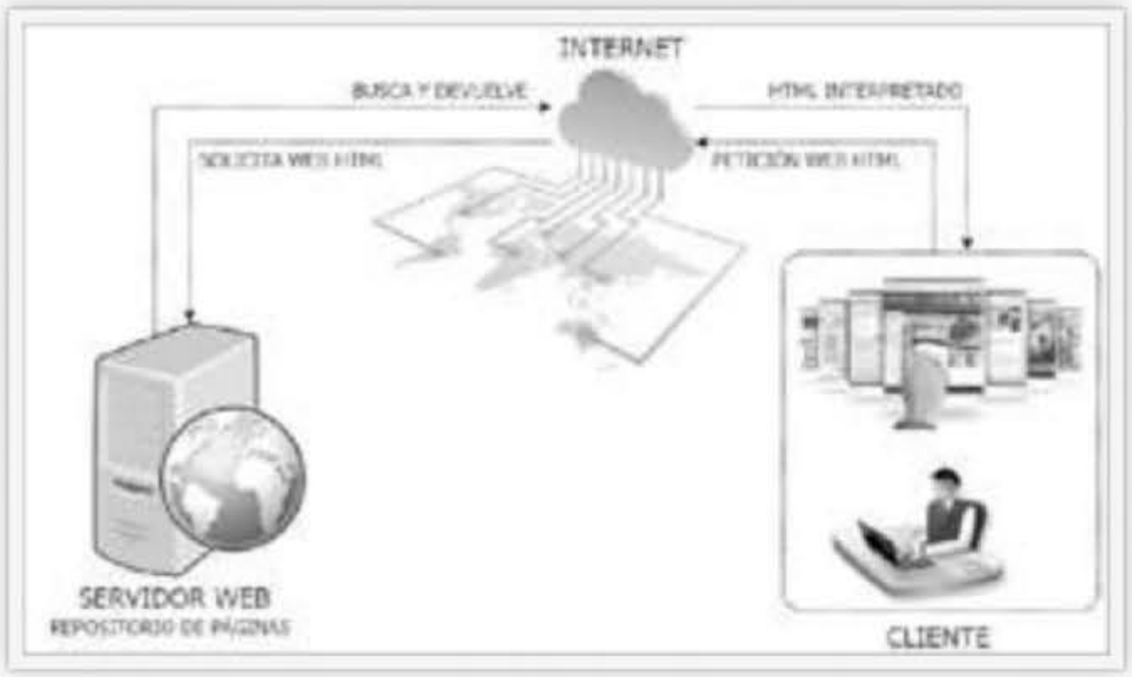


Figura 1.1 Funcionamiento de un documento web.
Fuente: Elaboración propia.

Todo inicia cuando el usuario ingresa a un navegador web, este podría ser Google Chrome o Firefox, en el cual se digita una dirección web (`<www.miweb.com>`); esta petición se realiza vía Internet; por lo tanto, hay que tener acceso a él. Seguidamente, dicha petición se transforma en una solicitud de búsqueda dentro del servidor web, el cual aloja diferentes tipos de documentos web. Para este caso, se hará un enfoque a los documentos de tipo HTML. Luego, estos son procesados e interpretados para pasar la información resultante al usuario que solicitó el servicio; hay que tener en cuenta que el acceso a Internet es un elemento muy importante para dicha transacción.

Un documento web se compone de etiquetas HTML, estas etiquetas tienen un patrón de funcionamiento, el mal uso de estas generará un error en la interpretación. Ahora bien, mientras no exista error,

el documento será mostrado al cliente no como etiquetas, sino como un documento web totalmente interactivo para el usuario. A continuación se muestra un script básico en HTML5:




```
<!DOCTYPE html>
<html lang='es'>
    <head>
        <title>Título de la WEB</title>
        <meta charset='UTF-8'>
    </head>
    <body>
        <header>
            <h1>Titulo de la WEB</h1>
        </header>
        <nav>
            <a href='seccion1.html'>SECCIÓN 1</a>
            <a href='seccion2.html'>SECCIÓN 2</a>
        </nav>
        <section>
            <article>
                <h2>CONTENIDO PRINCIPAL</h2>
                <p>Este es el contenido principal de mi web</p>
            </article>
        </section>
        <footer>
            <h4>Todos los derechos reservados</h4>
        </footer>
    </body>
</html>
```

1.6 Navegadores web

Es un programa que tiene como finalidad acceder a la web a petición de los usuarios. Este interpreta la información que se encuentra dentro de los diferentes archivos alojados en un servidor web para que puedan ser visualizados.

Normalmente, los documentos web se encuentran alojados en un repositorio dentro de un servidor web y estos solo podrán ser accedidos mediante un navegador web. Dichos documentos son también llamados páginas web, contienen los vínculos que enlazan, mediante un texto o una imagen, a otro documento web, a este proceso se la llama navegación.

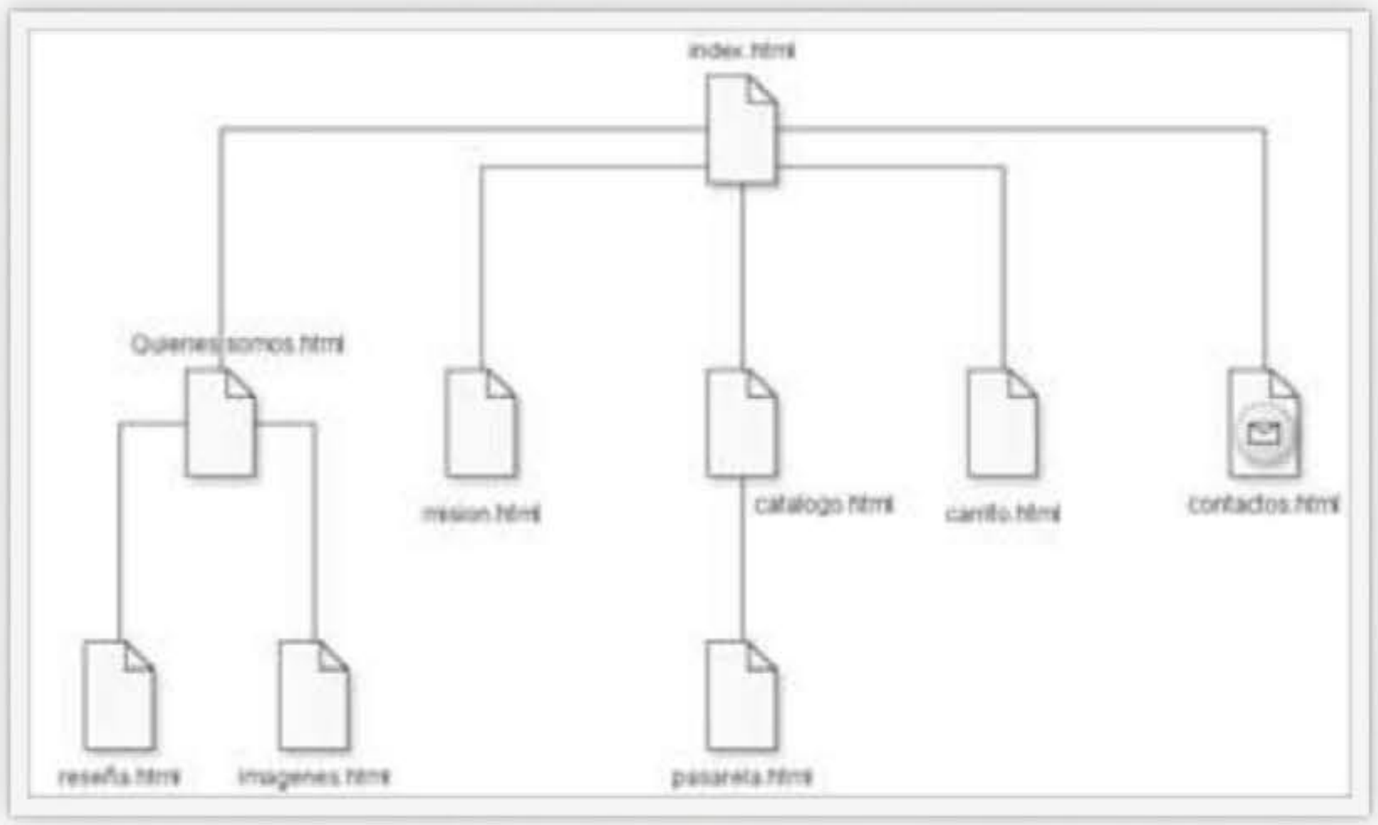


Figura 1.2 Mapa de un sitio web.
Fuente: Elaboración propia.

A continuación, se mencionan los principales navegadores web:

NAVEGADOR	DESCRIPCIÓN
 Google Chrome	Lanzado en el año 2008, es uno de los navegadores más usados y con un rápido crecimiento gracias a la alianza con Google y su sistema de publicidad muy bien implementada. Entre las ventajas que presenta Chrome se pueden citar: » La compatibilidad que tiene con los diferentes sistemas operativos como Windows, Ubuntu, Debian, Chrome OS, entre otros; así como también en sistemas operativos móviles como Android y iOS. » La seguridad que utiliza, la cual está apoyada en una lista negra donde se registran los sitios web que contienen *software* dañino, y que el navegador pone como advertencia al usuario. » La estabilidad en sus procesos, que se debe a la arquitectura multiproceso que posee, es decir, si uno de los procesos entra en conflicto, se cierra dicho proceso, pero los demás siguen activos. » La privacidad que posee a través de su modo incógnito, el cual hace que el navegador no registre ninguna actividad y borre las *cookies* usadas. » La instalación de *plugins* de terceros permite una mayor personalización del navegador, además que no requiere reiniciar después de instalar.
 Mozilla Firefox	Es un navegador de *software* libre desarrollado por la empresa Mozilla. Cuenta con todas las características que debe poseer un navegador web, como, por ejemplo, los aspectos de seguridad y la personalización a partir de *plugins* de terceros. Por otro lado, también trabaja con diferentes sistemas operativos y de distinta plataforma como la móvil. Mejora el aspecto que presentaba en las versiones anteriores, reduce el uso de memoria, y amplía la integración con las redes sociales mediante una API.

NAVEGADOR	DESCRIPCIÓN
Opera	Es considerado como uno de los navegadores web más seguros, estables y rápidos entre sus competidores. Presenta una gran compatibilidad con diferentes sistemas operativos. Fue desarrollado por la compañía Opera Software. A continuación se mencionan algunas características que lo diferencian de otros navegadores: » Administra y gestiona los correos electrónicos. » Descarga archivos mediante torrent. » Tiene un cliente IRC para realizar conversiones en tiempo real. » Permite la lectura actualizada de RSS. Ofrece un sistema de seguridad avanzada con tecnologías anti-*malware*, anti-*phishing* y cifrado de datos; asimismo, administra los archivos tipo *cookies*, de geolocalización y las contraseñas, eliminando los archivos necesarios para la protección de la información.
Safari	Navegador web que se ejecuta solo en algunos sistemas operativos, como Mac OS e iOS. Actualmente, existe una versión para Windows. Entre sus características se puede citar el manejo de una web totalmente *offline*. Asimismo, posee una integración a las redes sociales y la nube.
Internet Explorer	Una de sus principales características es que ya viene instalado en todos los sistemas operativos de Windows. Pero en los últimos años presentaba muchas deficiencias con respecto a la seguridad de su información es así que Microsoft decide no darle soporte y nace Edge. Microsoft Edge es un navegador web joven y moderno desarrollado por la misma Microsoft, el cual viene incluido en los sistemas operativos Windows 10. Este navegador presenta un nuevo aspecto distinto del Internet Explorer tradicional y adiciona nuevas características que superan a Chrome y Firefox.

1.7 Determinar si mi navegador está preparado para HTML5

Hasta ahora se ha estudiado algunas características que presentan los navegadores web, pero lo que realmente se necesita es analizar si estos navegadores usan todas las posibilidades que tiene HTML5. Para ello, hay que acceder al navegador y digitar la siguiente URL: **<http://html5test.com/>**, luego se analizarán los resultados. Ahora bien, se mencionan los resultados obtenidos de tres navegadores:

› **Microsoft Edge**

Para esta prueba, muestra un valor de 468 puntos según el test de evaluación sobre el uso de todos los recursos de HTML5.



› **FireFox versión 57.0.2**

Para esta prueba, muestra un valor de 486 puntos según el test de evaluación sobre el uso de todos los recursos de HTML5.



› **Google Chrome versión 64.0.3282.39**

Para esta prueba, Chrome arroja un valor de 520 puntos según el test de evaluación sobre el uso de todos los recursos de HTML5. Haciendo una comparación, Chrome presenta el más alto puntaje.



Descripción de los paneles mostrados en el test.

1.8 Editores HTML

Se le llama editor HTML a una aplicación que permite crear documentos web con extensión HTML. Hay que considerar que existen tres tipos de editores:

- **Simple:** Presenta un aspecto simple de edición sin formato. El diseño de un documento web se debe realizar únicamente con puro código, aquí se requiere un poco más de esfuerzo por parte del diseñador web.
- **Con ventanas:** Desarrolla el código en un área de trabajo específica y en otra área de la misma aplicación se puede visualizar el resultado del código, todo esto en tiempo real.
- **WYSIWYG:** Forma parte de los editores más complejos. Implementa un documento web en el que el usuario va colocando distintos elementos en una página y el propio *software* se encarga de generar el documento HTML. En la actualidad, existen numerosos editores de páginas web. Sin embargo, en ocasiones, es difícil encontrar el editor que se adapte a las necesidades propias de cada desarrollador o diseñador. A continuación, se enumeran los mejores editores de páginas web que existen en este momento:

Ejemplo 1

Código en Notepad++

Implemente una aplicación básica de HTML5 usando la aplicación NotePad++ que puede descargarse gratuitamente desde la siguiente dirección: <https://notepad-plus-plus.org/download/v7.5.3.html>.

Pasos:

1. Para ingresar a la aplicación, haga clic en su ícono.

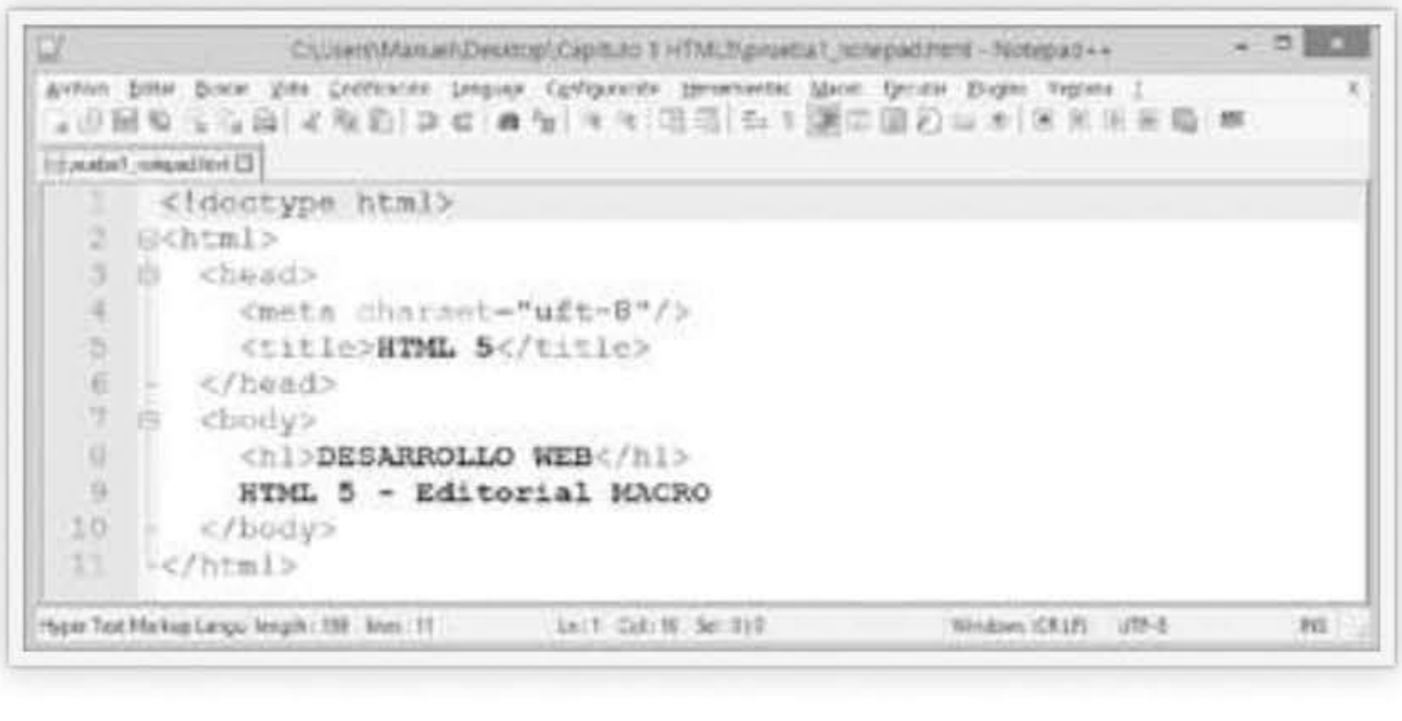


2. Coloque el siguiente código:

```
<!doctype html>
<html>
   <head>
      <meta charset="uft-8"/>
      <title>HTML 5</title>
   </head>
   <body>
      <h1>DESARROLLO WEB</h1>
      HTML 5-Editorial MACRO
   </body>
</html>
```

3. Luego grabe el código. Para ello, haga clic en **Archivo > Guardar como...** y asigne un nombre. Para este caso escriba **prueba1_notepad.html**. Este proceso hace que las etiquetas HTML tomen un determinado color dentro del editor con la finalidad de diferenciarlos.

4. El entorno debe mostrarse de la siguiente manera:



5. Para visualizar el resultado del código, primero elija un navegador web. En este caso, seleccione a Google Chrome. Para ello, haga clic derecho sobre el archivo web HTML y luego clic izquierdo en **Abrir con > Google Chrome**, tal como se muestra en la siguiente imagen:



6. Finalmente, el documento se mostrará de la siguiente manera:



Ejemplo 2

Sublime Text 3

Desarrolle una aplicación básica HTML5 usando la aplicación Sublime Text 3, la cual puede descargarse gratuitamente desde la siguiente dirección: **<https://www.sublimetext.com/3>**.

Pasos:

1. Ingrese a la aplicación haciendo clic en su ícono.

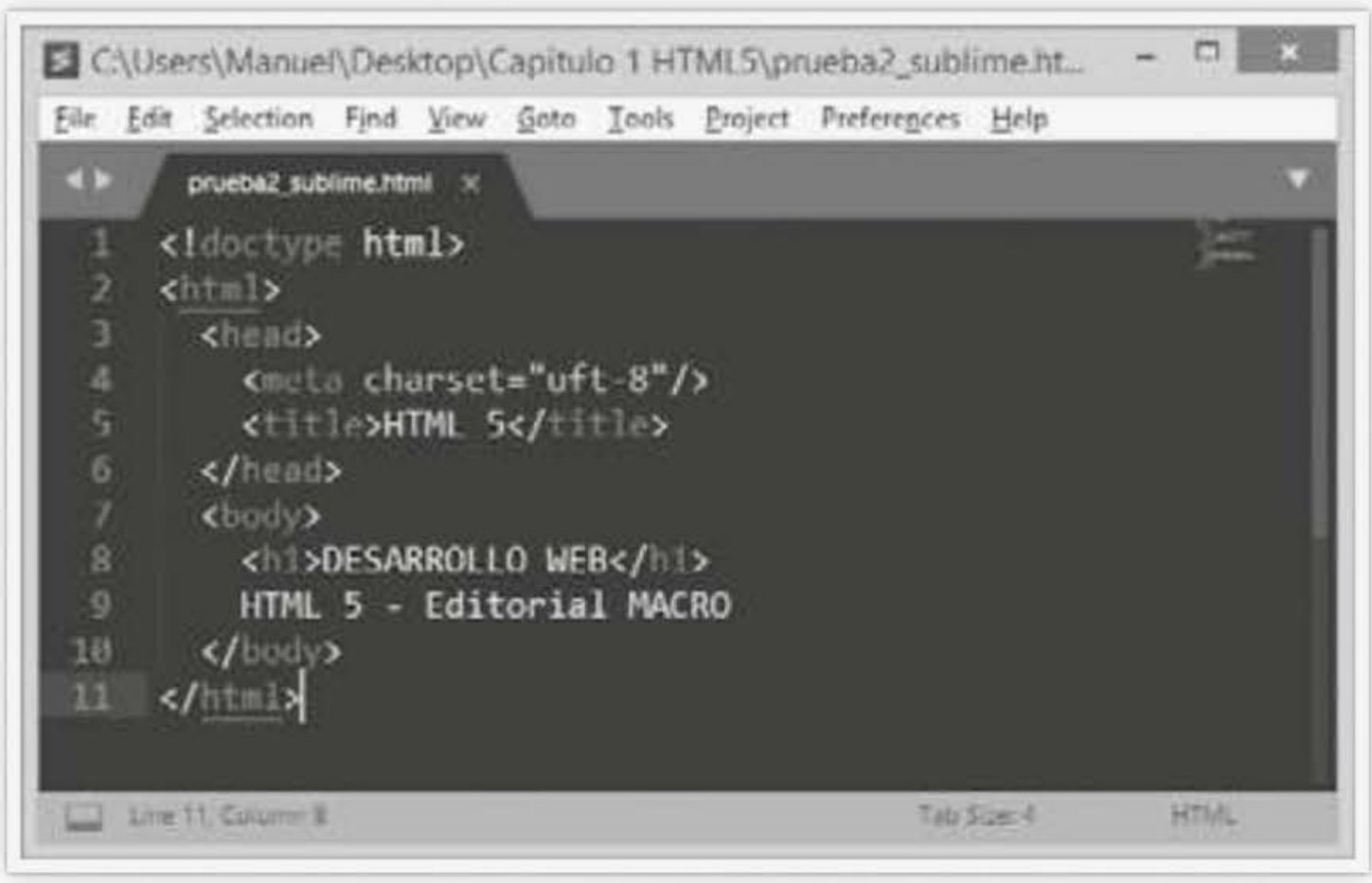


2. Coloque el siguiente código:

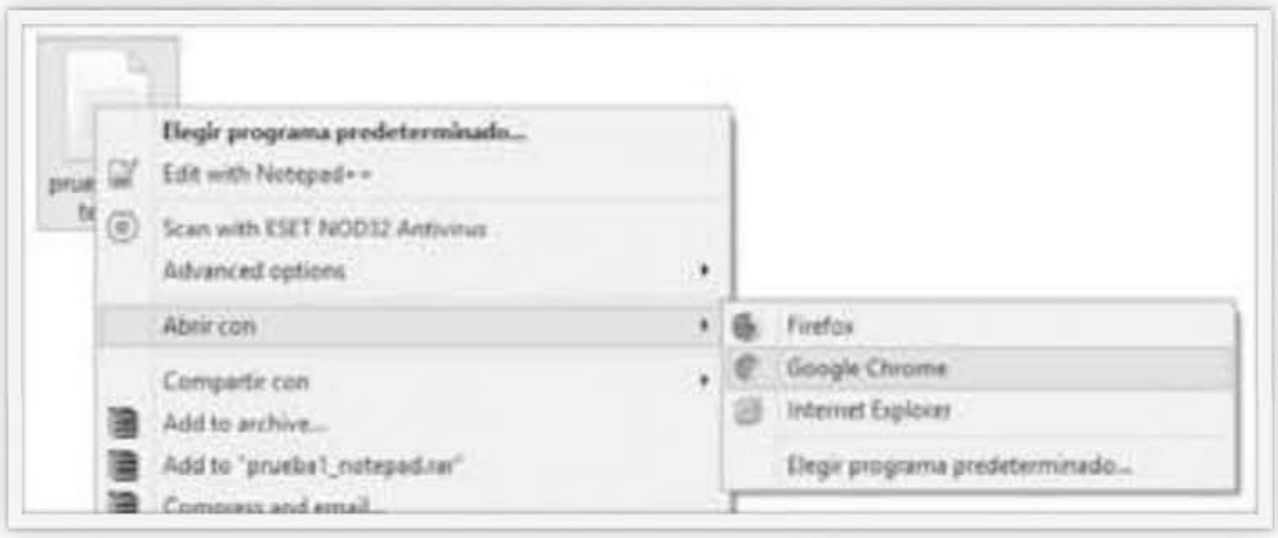
```
<!doctype html>
<html>
    <head>
        <meta charset="uft-8"/>
        <title>HTML 5</title>
    </head>
    <body>
        <h1>DESARROLLO WEB</h1>
        HTML 5-Editorial MACRO
    </body>
</html>
```

3. Para grabar el código, haga clic en **File > Save as...**, luego asigne un nombre. Para este caso, escriba **prueba2_sublime.html**.

4. El entorno debe mostrarse de la siguiente manera:



5. Para visualizar el resultado del código, primero seleccione un navegador web. Para este caso, elija a Google Chrome. Haga clic derecho sobre el archivo web HTML y luego clic izquierdo en **Abrir con > Google Chrome**, tal como se muestra en la siguiente imagen:



6. Finalmente, el documento se mostrará de la siguiente manera:



Ejemplo 3

Netbeans 8.2

Implemente una aplicación básica HTML5 usando la aplicación Netbeans 8.2 que puede descargarse gratuitamente desde la siguiente dirección: **<https://netbeans.org/downloads/>**.

Pasos:

1. Para ingresar a la aplicación, haga clic en su ícono.

2. Luego seleccione **File > New Project > HTML5/Javascript > HTML5/JS Application**, tal como se muestra en la siguiente imagen:



3. Seguidamente, haga clic en el botón siguiente para poder asignar un nombre al proyecto web. En este caso, como ejemplo, asigne el nombre **pjCapitulo1**. Si necesita modificar la ubicación de los archivos, seleccione el botón **Browse...**

4. Luego seleccione el botón siguiente para poder seleccionar una plantilla de sitio. En este caso, no se cuenta con un sitio web; por lo tanto, seleccione **No Site Template**.



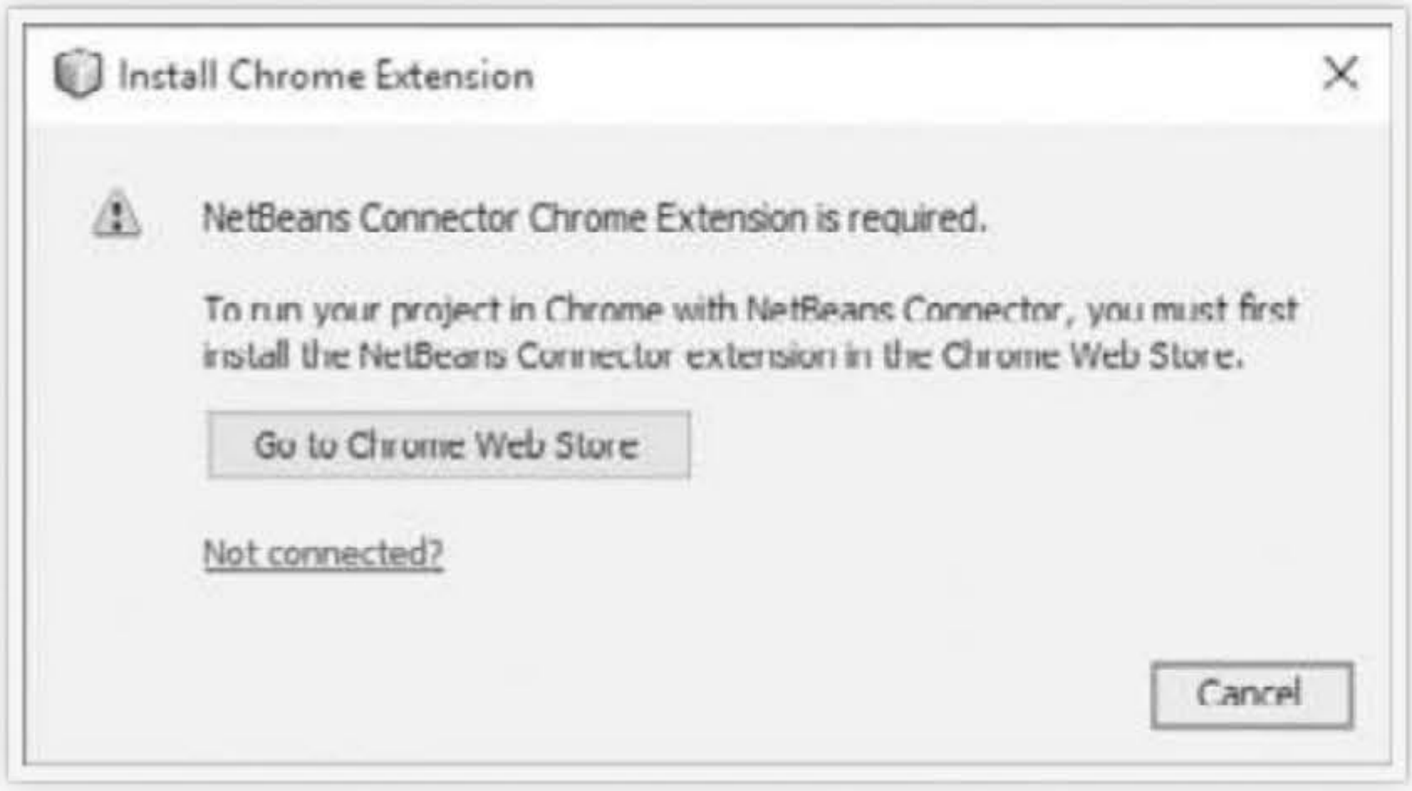
5. Seguidamente, seleccione **Finish** para iniciar el proyecto. La pantalla se presenta de la siguiente manera:

Como puede observarse se crea de forma automática el archivo **index.html**, este contiene código de forma predeterminada que se puede modificar y adecuar según las necesidades del diseñador.

6. Coloque el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <h1>DESARROLLO WEB</h1>
        HTML 5-Editorial MACRO
    </body>
</html>
```

7. Para visualizar el resultado del código, haga clic derecho sobre el archivo **index.html**. Luego seleccione **Run File**. Si es la primera vez que ejecuta un archivo html desde Netbeans, descargue la librería que permite conectar Google Chrome con la aplicación web, para esto debe seleccionar **Go to Chrome Web Store** como se muestra en la siguiente ventana:



8. Finalmente, el documento se mostrará de la siguiente manera:



1.9 Comprendiendo la URL

También llamado «localizador uniforme de recursos». La URL permite al navegador encontrar una dirección web en la Internet. La URL se compone de un nombre del servidor, probablemente la carpeta donde se encuentra el documento web a visualizar, el nombre del archivo web y el protocolo a usar para recuperar los datos.

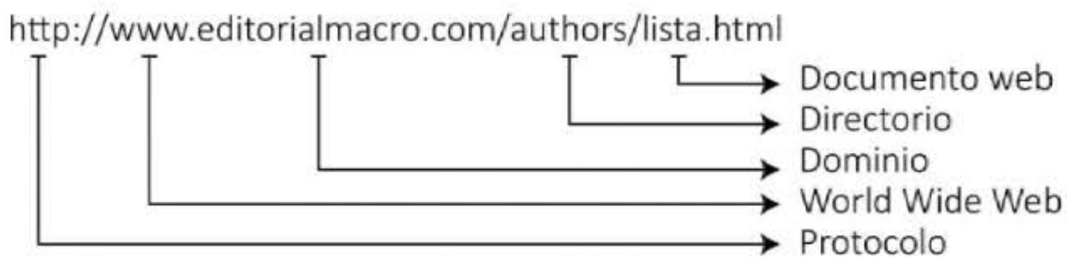


Figura 1.3 Componentes de la URL.
Fuente: Elaboración propia.

Donde:

- **http:** Es un protocolo de comunicación entre el navegador y el servidor web. Este define un conjunto de reglas para que el navegador y el servidor se comuniquen con claridad. También pueden emplearse cualquiera de las siguientes opciones:
  - **https:** Proporciona comunicaciones seguras en Internet.
  - **ftp:** Es un protocolo de transferencia de archivos.
  - **mailto:** Está destinado para direcciones *e-mail*.
  - **file:** Está determinado para recursos disponibles en la computadora o en una red local.
- **www:** Es el sistema de documentos de hipertexto que se encuentran enlazados entre sí y a los que se accede por medio de Internet.
- **com**: Es la extensión del dominio, el cual se clasifica en genéricos (**.com**, **.net**, **.org**, **.info**, **.biz**) y geográficos (**.pe**, **.es**, **.jp**, **.fr**, **.us**, etc.).
- **Directorio**: Es un conjunto de carpetas que permiten la organización de un proyecto web. Algunas veces reciben el nombre de sitios webs. Hay un lugar llamado «directorio principal» que permite alojar todos los archivos de un proyecto web, inclusive más directorios.
- **lista.html**: Es el nombre del documento web que se presentará en el navegador. Por defecto, el nombre es **index.html**, pero puede asignarse el nombre según la necesidad de la aplicación.

1.10 Servidores web locales

Un servidor web local es un servicio que se encuentra instalado en un equipo personal. Normalmente, es usado como reemplazo de un servidor web real que permitirá realizar pruebas de nuestras aplicaciones web.

Para realizar pruebas, se necesita de un servidor web local. Seguidamente, se muestra y desarrolla una lista de servidores (XAMPP y WAMP) que permitirán convertir una computadora personal en un servidor.

- **XAMPP**

  Es una distribución de Apache completamente gratuita y fácil de instalar, que contiene el sistema de gestor de base de datos MariaDB y los lenguajes de programación PHP y Perl. Es multiplataforma, por lo que funciona en diversos sistemas operativos como Windows, Linux, Mac e incluso el sistema Solaris.

  XAMPP Apache + MariaDB + PHP + Perl

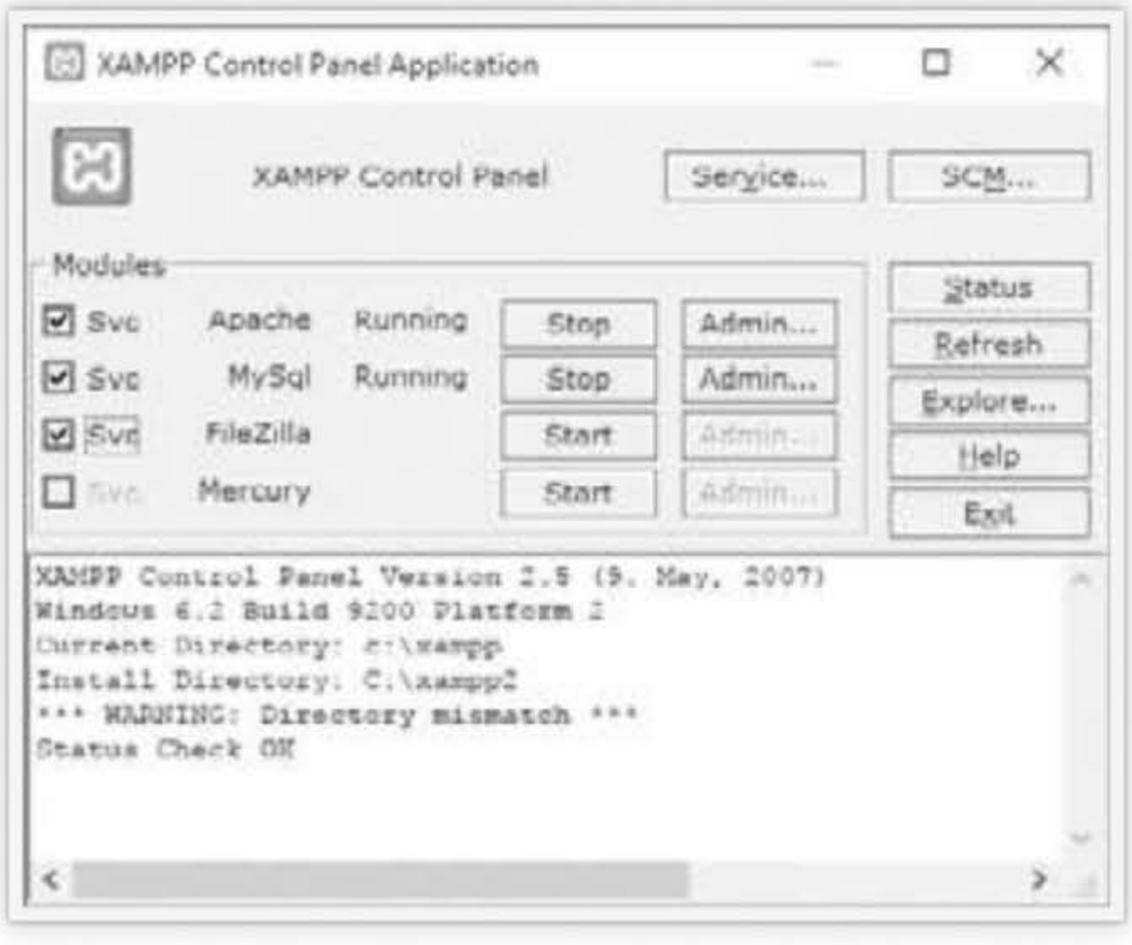
  Link de descarga: **<https://www.apachefriends.org/es/index.html>**.

Ejemplo 4

A continuación, se muestra la instalación de XAMPP.

Pasos:

1. Una vez instalada, habilite los servicios. Para este caso, solo será necesario habilitar **Apache**; para ello, seleccione el botón **Run**.



2. Para iniciar con el servicio web, seleccione el botón **Admin...** y desde ahí se podrá probar las aplicaciones web.



› WAMP

Es un paquete completo al estilo «apachefriends» que permite instalar y configurar fácilmente en el sistema todo lo necesario para comenzar proyectos web, se incluyen las últimas versiones del

servidor Web Apache2, el lenguaje de programación PHP y su administrador PhyMyAdmin, así como el servidor de base de datos MySQL.

Link de descarga: <https://sourceforge.net/projects/wampserver/files/>.

Ejemplo 5

A continuación, se muestra la instalación de WAMP.

Pasos:

1. En la instalación se mostrará la información de los servicios que se habilitarán. Por ejemplo, este *software* cuenta con el servidor Apache, el gestor de base de datos MySQL, el lenguaje de programación PHP y una aplicación de administración llamada PHPMyAdmin.



2. Para habilitar los servicios, seleccione la opción **Start All Services**. Luego, se mostrará un ícono al costado del reloj del sistema, el cual se presenta como una letra W de color verde.



1.11 Publicación local y navegación *offline*

Para realizar la publicación local se debe contar con un archivo de extensión HTML y, según el servidor seleccionado, alguna que otra acción específica para la publicación. En seguida, véanse cuáles serían los pasos para XAMPP.

- **Publicar con XAMPP**

  Véase el siguiente ejemplo:

Ejemplo 6

A continuación, se mostrará cómo publicar un documento web en XAMPP.

Pasos:

1. Seleccione el archivo HTML que hay que publicar y cópielo.
2. Ingrese a la carpeta **C:\xampp\htdocs** y cree una carpeta llamada **capitulo1**. Asegúrese que el nombre no contenga letras en mayúsculas ni espacios en blanco.
3. Luego, dentro de la carpeta **capitulo1**, pegue el archivo. El resultado de la acción debe mostrarse de la siguiente manera:



4. Finalmente, para poder ejecutar el documento web, ingrese a un navegador y coloque la siguiente dirección URL **localhost/capitulo1/index.html**, el navegador debe mostrarse de la siguiente manera:



Caso desarrollado
Archivo web con HTML5

Implemente un archivo web con HTML5 que muestre el siguiente diseño:

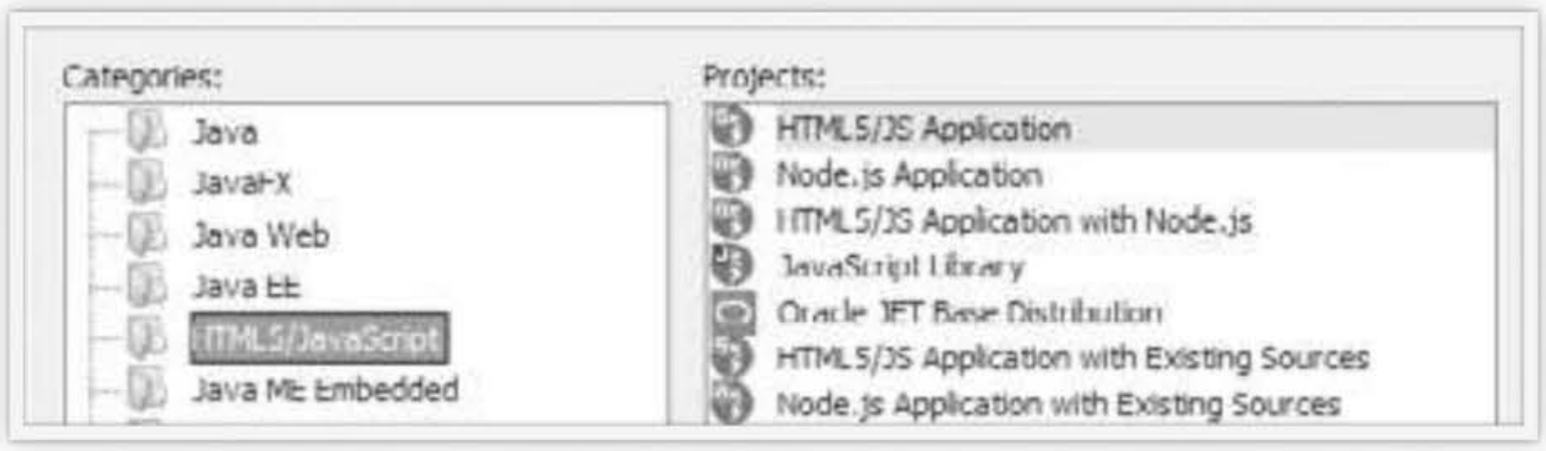


Consideraciones:

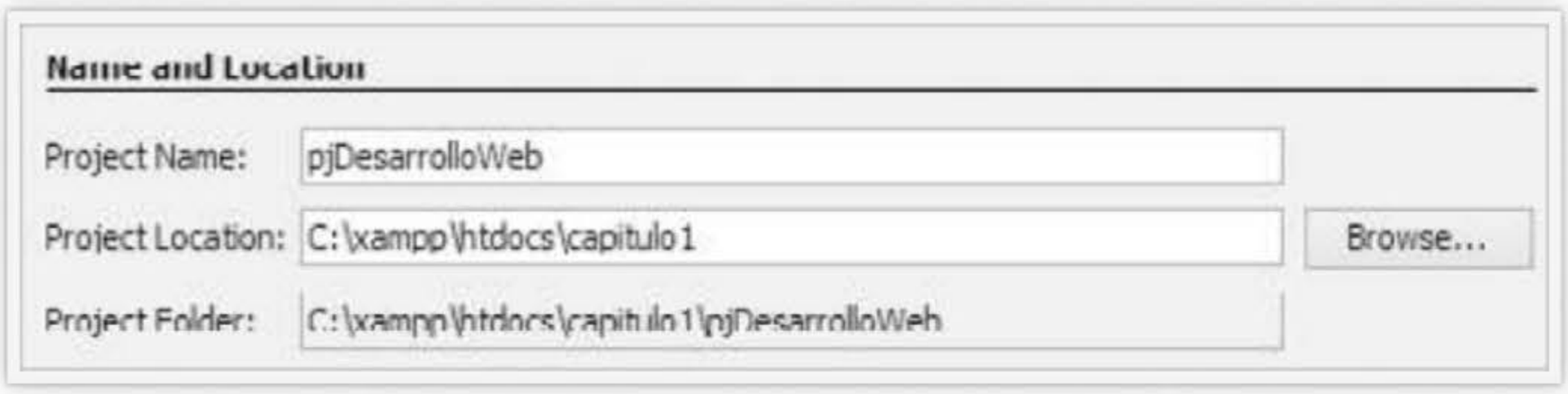
- Use como servidor a XAMPP.
- Use como editor HTML a Netbeans.

Solución:

1. Ingrese a Netbeans.
2. Seleccione **File > New Project**.
3. Luego elija **HTML5/Javascript > HTML5/JS Application**.



4. Asigne el nombre **pjDesarrolloWeb** al proyecto. Asimismo, direccione los archivos al directorio del servidor local. Para esto, seleccione el botón **Browse...** En este caso se ha creado una carpeta llamada **capitulo1** para una mejor organización.



5. En el siguiente paso se solicita seleccionar un sitio web, pero esto no es un requisito importante en el proyecto; por lo tanto, seleccione **No Site Template** y luego haga clic en el botón siguiente.



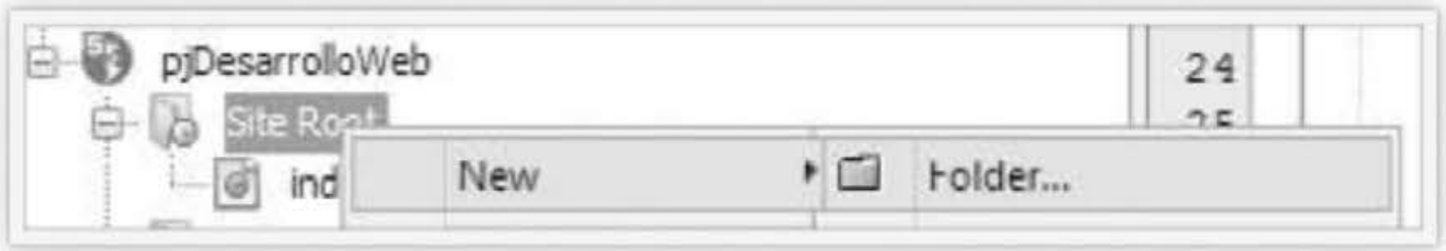
6. En el siguiente paso se nos solicita seleccionar herramientas para nuestro proyecto web, como sugerencia por el momento, desactive todas las opciones, tal como se muestra en la siguiente ventana:



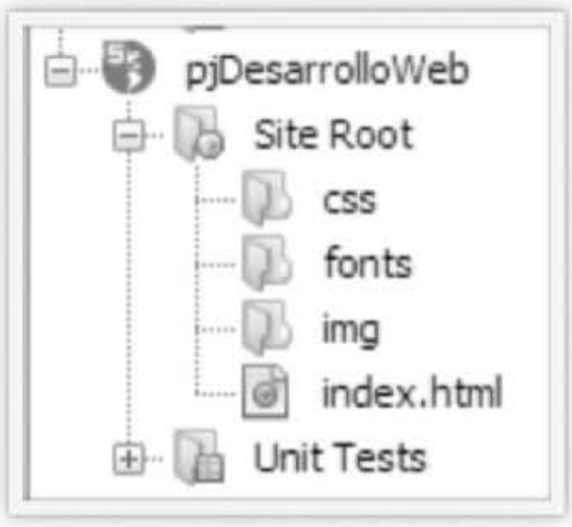
7. Para finalizar la etapa de selección del proyecto, seleccione el botón **Finish**, hay que tener en cuenta que el panel **Projects** debe mostrarse de la siguiente manera:



8. Luego, para una mejor organización de las imágenes y los estilos que se usarán en los documentos web, se debe crear algunas carpetas, para esto haga clic derecho sobre **Site Root** y seleccione **New > Folder...**



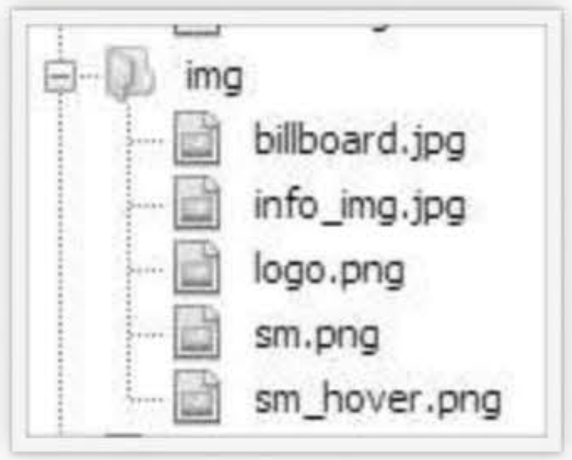
9. Asigne el nombre **css** a la carpeta para los archivos de tipo de hojas de estilo para el proyecto. Realice los mismos pasos para crear la carpeta **img** en donde se registrarán las imágenes que usará el proyecto y, finalmente, cree la carpeta **fonts** para el tipo de letra que se emplearán para los textos en la web. El panel **Projects** debe mostrarse de la siguiente manera:



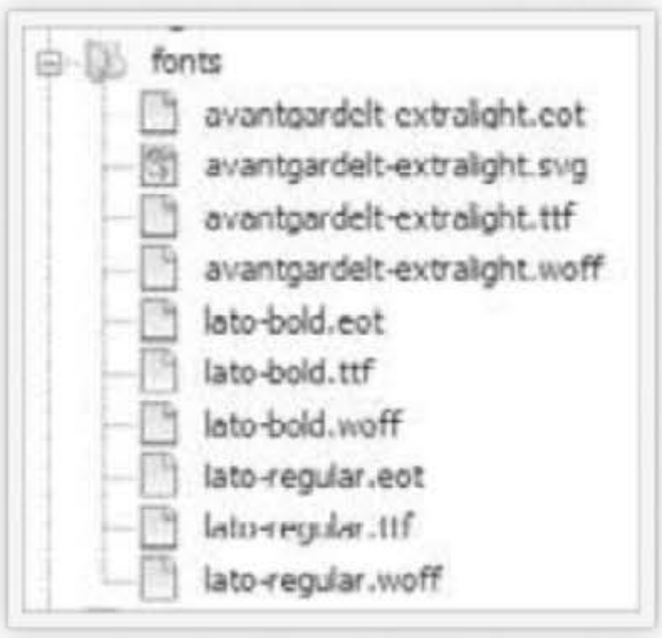
10. Ahora, cree un archivo de tipo hojas de estilo. Para esto, haga clic derecho sobre la carpeta **css**, luego seleccione **New > Cascading Style Sheet...**



11. En el caso de las imágenes del proyecto, seleccione los archivos de tipo **jpg** o **png** y cópielos a la carpeta **img**, como se muestra en la siguiente imagen:



12. En el proyecto, emplee diversos tipos de letras; para ello, copie las fuentes a la carpeta **fonts**, como se muestra en la siguiente imagen:



13. Finalmente, coloque los códigos dentro de los archivos correspondientes.

» En el documento **index.html**:

```
<!DOCTYPE html>
   <html lang='en'>
      <head>
         <title>Diseño Web con HTML5 y CSS3</title>
         <meta charset='utf-8'>
         <link rel='stylesheet' type='text/css' href='css/main_responsive.css'>
         <script type='text/javascript' src='js/main.js'></script>
</head>
<body>
   <header>
      <div class='wrapper'>
         <img src='img/logo.png' alt='Crafty' class='logo'>
         <a href='#' class='menu_icon' id='menu_icon'></a>
         <nav id='nav_menu'>
            <ul>
               <li><a href='#'>Introducción</a></li>
               <li><a href='#'>Historia</a></li>
               <li><a href='#'>Versiones</a></li>
               <li><a href='#'>Características</a></li>
            </ul>
         </nav>
         <ul class='social'>
            <li><a class='fb' href='#'></a></li>
            <li><a class='twitter' href='#'></a></li>
            <li><a class='gplus' href='#'></a></li>
         </ul>
      </div>
   </header>
```

```
<section class='billboard'>
        </section>
      <footer>
        <p class='rights'>Copyright ©HTML5-All rights reserved</p>
      </footer>
   </body>
</html>
```

» En el documento **css**:

```
/* Fonts */
@font-face {
   font-family: 'lato_regular';
   src: url('../fonts/lato-regular.eot');
   src: url('../fonts/lato-regular.eot?#iefix') format('embedded-opentype'),
     url('../fonts/lato-regular.woff') format('woff'),
     url('../fonts/lato-regular.ttf') format('truetype'),
     url('../fonts/lato-regular.svg#LatoRegular') format('svg');
   font-weight: normal;
   font-style: normal;
}

@font-face {
   font-family: 'lato_bold';
   src: url('../fonts/lato-bold.eot');
   src: url('../fonts/lato-bold.eot?#iefix') format('embedded-opentype'),
   url('../fonts/lato-bold.woff') format('woff'),
   url('../fonts/lato-bold.ttf') format('truetype'),
   url('../fonts/lato-bold.svg#LatoBold') format('svg');
   font-weight: normal;
   font-style: normal;
}

@font-face {
   font-family: 'avantgardelt-extralight';
   src: url('../fonts/avantgardelt-extralight.eot');
   src: url('../fonts/avantgardelt-extralight.eot?#iefix')
   format('embedded-opentype'),
   url('../fonts/avantgardelt-extralight.woff') format('woff'),
   url('../fonts/avantgardelt-extralight.ttf') format('truetype'),
   url('../fonts/avantgardelt-extralight.svg#avantgardelt-extralight')
   format('svg');
   font-weight: normal;
   font-style: normal;
}
```

```
/* General CSS*/
.wrapper{
   max-width: 1200px;
   hei   ght: auto;
   margin: 0 auto;
   position: relative;
}

::selection {color:#ffffff;background:#353434;}
::-moz-selection {color:#ffffff;background:#353434;}

.feature a,
.cta a.cta_btn,
.testimonials #t_navigation a,
header ul.social li a,
header nav ul li a{
   transition: all .2s linear;
   -webkit-transition: all .2s linear;
   -moz-transition: all .2s linear;
   -o-transition:all .2s linear;
}

/* header */
header{
   width: 100%;
   height: 80px;
   background: #353434;
}

header .logo{
   float: left;
   margin-top: 21px;
   width: 60px;
   height: 50px;
}

header .menu_icon{
   display: block;
   width: 35px;
   height: 35px;
   background: url('../img/menu_icon.png') no-repeat;
   float: right;
   margin-top: 23px;
   display: none;
}
```

```
header nav{
    overflow: hidden;
    display: inline-block;
    margin: 20px 0 0 40px;
    padding: 13px 40px;
    border-left: 1px #404040 solid;
    z-index: 9999;
}

header nav ul{
    list-style: none;
}

header nav ul li{
    float: left;
    margin-left: 35px;
    font-size: 16px;
    font-family: 'lato_regular', arial;
    letter-spacing: 1px;
}

header nav ul li:first-child{
    margin-left: 0;
}

header nav ul li a {
    text-decoration: none;
    color: #fff;
}

header nav ul li a:hover{
    color: #65bf95;
}

header ul.social{
    float: right;
    list-style: none;
    margin-top: 23px;
}

header ul.social li{
    float: left;
    margin-left: 10px;
}

header ul.social li a{
    display: block;
    width: 35px;
    height: 35px;
    background: url('../img/sm.png') no-repeat;
}
```

```
header ul.social li a:hover{
    background: url('../img/sm_hover.png') no-repeat;
}

header ul.social li a{
    display: block;
    width: 35px;
    height: 35px;
}

header ul.social li a.fb{
    background-position: 0 0;
}

header ul.social li a.twitter{
    background-position: -36px 0;
}

header ul.social li a.gplus{
    background-position: -72px 0;
}

/* Billboard */
.billboard{
    width: 100%;
    height: 580px;
    background: url('../img/billboard.jpg') no-repeat center center;
    background-size: cover;
    -webkit-background-size: cover;
    -moz-background-size: cover;
    -o-background-size: cover;
    overflow: hidden;
}

.billboard .caption{
    margin-left: 515px;
    margin-top: 220px;
}

.billboard .caption p{
    display: inline-block;
    color: #000;
    font-family: 'avantgardelt-extralight', arial;
    font-weight: lighter;
    font-size: 35px;
    padding: 20px;
    background: rgba(255, 255, 255, 0.6);
    margin-bottom: 2px;
    text-transform: uppercase;
    z-index: 8888;
}
```

```
footer .rights{
    color: #3b3b3b;
    font-family: 'lato_regular', arial;
    font-size: 14px;
    text-align: center;
    margin-top: 30px;
    line-height: 25px;
}
```

CAPÍTULO 2

Estructurando documentos para la web



Competencias

- Implementa un documento web básico usando las etiquetas de HTML5.
- Emplea mejores prácticas en la implementación de una web con etiquetas grupales.

Contenido

2.1 Estructura de una etiqueta HTML5
2.2 Etiquetas obsoletas para HTML5
2.3 Elementos que cambian su significado
2.4 Los atributos de una etiqueta HTML5
2.5 Identificación de los atributos en una etiqueta HTML5
2.6 Estructura global de un documento web
- Caso desarrollado: Maquetación de un documento web HTML 5 básico

2.7 Estructurando el cuerpo del documento web
2.8 Elementos internos
2.9 Comentarios en HTML5
2.10 Otras etiquetas

2.1 Estructura de una etiqueta HTML5

HTML5 es un lenguaje de marcas o también llamado etiquetas el cual está compuesto por un conjunto de elementos que son la base de su estructura. En el diseño de un documento web se puede encontrar párrafos, videos, imágenes, enlaces y elementos adicionales, como Javascript, todos ellos controlados mediante etiquetas. A continuación, se muestra la composición de un elemento web:

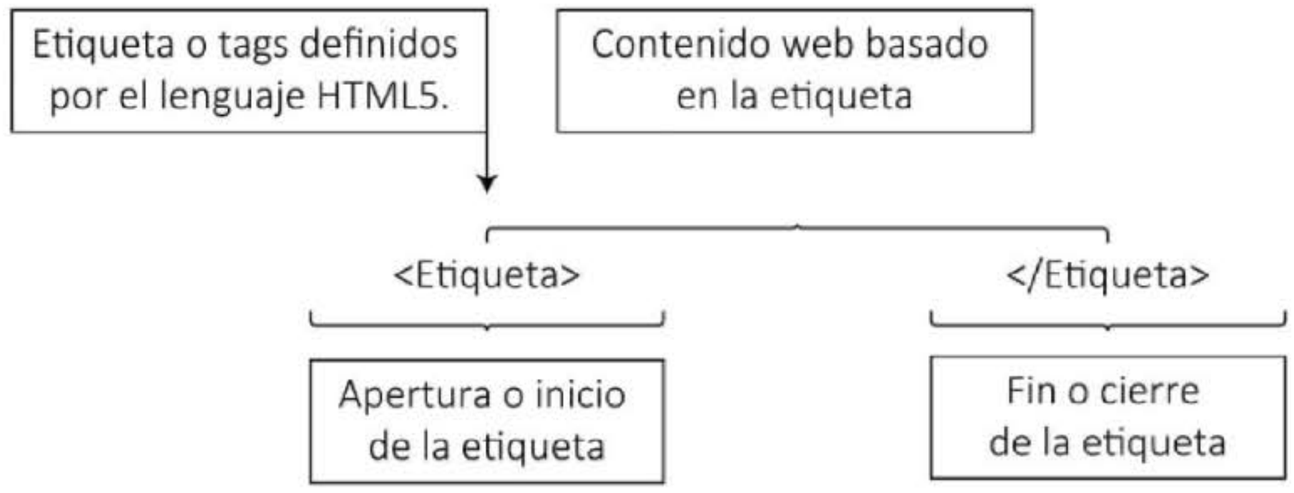


Figura 2.1 Estructura de una etiqueta web.
Fuente: Elaboración propia.

- **Características de las etiquetas**
  - » Las etiquetas son estandarizadas por la W3C; por lo tanto, la especificación formal se encuentra en su web oficial.
  - » No hay distinción entre mayúsculas ni minúsculas. Esto quiere decir que una etiqueta escrita en mayúscula o minúscula será interpretada de la misma forma por el navegador.
  - » Normalmente se usan dos etiquetas: una de inicio y otra de fin. Este última usa una barra inclinada para diferenciar el inicio del fin.
  - » Las etiquetas obsoletas no han sido deshabilitadas por los navegadores web, estas aún se siguen usando en HTML5.

Ejemplo 1

Véanse dos tipos de etiquetas:

- **<h1> Web HTML5 </h1>**
  Muestra el texto "Web HTML5" al estilo de título 1, obteniéndose el siguiente resultado:

  Web HTML5

- ** Lenguaje de etiquetas HTML5 **
  Muestra el párrafo "Lenguaje de etiquetas HTML5", se consigue siguiente formato:

  Lenguaje de etiquetas *HTML5*

  También, existen algunas etiquetas que tienen elementos que permiten configurar el resultado del elemento web.

Ejemplo 2

Véanse dos etiquetas con elementos:

- **<img src="html5.png" alt="html5">**
  Muestra la imagen **html5.png** gracias a la especificación del archivo en el atributo **src**, mientras que el atributo **alt** especifica el mensaje **html5** al posicionar el puntero del mouse por encima de la imagen.

- **<img src="html5.png">**
  Muestra una imagen que permite enlazar a una página web, es decir, al seleccionar la imagen esta se enlaza a una URL.

2.2 Etiquetas obsoletas para HTML5

La incorporación del HTML5 al desarrollo web no fue pensado para reemplazar todas las etiquetas existentes, sino más bien para agregar nuevas funcionalidades mediante nuevas etiquetas. Pero hay algunas que han quedado obsoletas o rezagadas gracias a la incorporación de las hojas de estilo CSS. Véase una lista de las principales etiquetas que ya no serán usadas en HTML5:

Tabla 2.1 Etiquetas obsoletas para HTML5.

ETIQUETA OBSOLETA	DESCRIPCIÓN	NUEVA FUNCIONALIDAD
acronym	Permite representar elementos acrónimos.	<abbr>
applet	Permite insertar un script externo dentro del documento web, como, por ejemplo, integrarlo con Java.	<object>
basefont	Permite determinar el tamaño de fuente predeterminado.	CSS
big	Permite determinar el tamaño de texto grande.	CSS
center	Permite centrar el contenido dentro de un documento web.	CSS
dir	Permite listar sin orden específico.	
font	Permite establecer un tipo de fuente a un texto.	CSS
frame	Permite dividir un documento web en marcos.	-
frameset	Permite dividir en grupos un documento web en marcos.	-
noframe	Permite adicionar contenido alternativo para marcos web.	-
strike	Permite tachar un bloque te texto web.	CSS
tt	Permite mostrar el texto como teletype o espaciado simple.	CSS
u	Permite mostrar el texto como subrayado.	CSS
xmp	Permite definir un texto preformateado.	<pre>

Fuente: Elaboración propia.

Asimismo, también se han vuelto obsoletos algunos atributos correspondientes a ciertas etiquetas que se citan a continuación:

Tabla 2.2 Atributos obsoletos para HTML5.

ATRIBUTO OBSOLETO	DESCRIPCIÓN	ETIQUETAS ASOCIADAS
align	Especifica un tipo de alineación a un elemento web.	<caption>, <img>, <table>, <hr>, , <h1...h6>, <p>
alink	Especifica el color del link activo.	<body>
background	Especificar el color de fondo.	<body>
bgcolor	Especifica el color de fondo de un elemento web.	<body>, <table> <tr> <td> <th>
clear	Especifica el posicionamiento de un elemento web.	
compact	Especifica una lista que muestra el efecto de la más pequeña de lo normal, esto es, reduciendo el espacio entre líneas y la sangría de la lista.	,
color	Especifica el código de color.	<basefont>, <font>
border	Especifica el borde de un elemento de tipo imagen u objeto.	<img>, <object>
hspace	Especifica un valor para el espaciado de tipo horizontal.	<img>,<object>
link	Especifica un enlace web.	<body>
noshade	Especifica que no se muestre la línea en 3D.	<hr>
nowrap	Especifica el contenido de una celda de la tabla no se ajusta.	<td>, <th>
size	Especifica el tamaño de un elemento textual.	<basefont>, <font>, <hr>
text	Define el color de texto dentro de un documento web.	<body>
vlink	Especifica el color del enlace visitado.	<body>
width	Especifica el ancho de un elemento web.	<hr>, <pre>, <td>, <th>
vspace	Especifica el espacio vertical de una imagen o un objeto.	<img>, <object>

Fuente: Elaboración propia.

2.3 Elementos que cambian su significado

En HTML5 se ha reacomodado la funcionalidad de algunas etiquetas, es decir, ya no tienen el mismo significado entre las versiones anteriores y el HTML5. Véase una lista de las etiquetas modificadas en su funcionalidad.

- **<small>**

  En HTML5 la etiqueta **<small>** se usará para referenciar a las condiciones legales, la información de *copyright*, etc.

Ejemplo 3

Véase su uso en HTML5:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <small>
         Todos los derechos reservados Copyright @2018.
      </small>
   </body>
</html>
```

El navegador lo muestra de la siguiente manera:

Todos los derechos reservados Copyright @2018.

- **<b>**

En HTML5 la etiqueta **<b>** se empleará cuando se desee resaltar algún aspecto de un párrafo. La norma indica que la etiqueta **<b>** debe usarse como último recurso y de preferencia debe emplearse los encabezados **<h1...h6>** o los elementos ****, **** o **<mark>** en función del tipo de información que se desea resaltar.

Ejemplo 4

Véase su uso en HTML5:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <p>
      Favor de compartir esta información con su <b>equipo de trabajo</b> y
      consultar en el servidor las estadísticas de notas para detectar las
      secciones que se encuentren con bajo rendimiento y puedan reforzar
      los temas.
      </p>
   </body>
</html>
```

El navegador lo muestra de la siguiente manera:

Favor de compartir esta información con su **equipo de trabajo** y consultar en el servidor las estadísticas de notas para detectar las secciones que se encuentren con bajo rendimiento y puedan reforzar los temas.

› **<i>**

En HTML5 la etiqueta <i> permite aplicar el estilo cursiva a un texto. Por ejemplo, a un nombre propio, una frase en otro idioma, una idea, un pensamiento.

Ejemplo 5

Véase su uso en HTML5:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <p>
      Les invitamos a participar en la <i>Serie de Webinars sobre Enseñanza
      y Aprendizaje en Educación Superior</i>, donde se presentarán 6
      Webinars (4 en español, 1 en portugués y 1 en inglés) que estamos
      seguros serán de mucho interés para su futuro profesional.
      </p>
   </body>
</html>
```

El navegador lo muestra de la siguiente manera:

Les invitamos a participar en la *Serie de Webinars sobre Enseñanza y Aprendizaje en Educación Superior*, donde se presentarán 6 Webinars (4 en español, 1 en portugués y 1 en inglés) que estamos seguros serán de mucho interés para su futuro profesional.

› **<s>**

En HTML5 la etiqueta <s> aplica un tachado a un texto, el cual indicará que un párrafo que ya no es relevante o fiable.

Ejemplo 6

Véase su uso en HTML5:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <p>Contáctenos</p>
      <p><s>Teléfono anterior: (511) 512-0000</s></p>
      <p><strong>Teléfono nuevo: (01) 544-0000</strong></p>
   </body>
</html>
```

El navegador lo muestra de la siguiente manera:

Contáctenos

~~Teléfono anterior: (511) 512-0000~~

Teléfono nuevo: (01) 544-0000

- **<style>**

  En HTML5 la etiqueta **<style>** permite asignar un estilo a un determinado contenido. Esto ya no es recomendable, ya que actualmente se sugiere que ello se realice mediante enlaces a los archivos **css**. Solo en situaciones excepcionales se podrá aplicar estilos a los elementos utilizando su atributo **style**.

Ejemplo 7

Véase su uso en HTML5:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
      <style>
         body {
            color: white;
            background-color:red;}
      </style>
   </head>
   <body>
```

```
        <p>Manejo de Estilos</p>
    </body>
</html>
```

El navegador muestra de color rojo el fondo y las letras de color blanco.



2.4 Los atributos de una etiqueta HTML5

Hay que diferenciar que muchas de las etiquetas que presenta HTML5 presentan opciones estándar que dan la sensación de ser predefinidas. Pero hay otras que necesitan una especificación como, por ejemplo, la etiqueta **<image>**, que no solo se debe definir el nombre del archivo que hay que mostrar sino también el tamaño de la imagen, el tipo, etc.

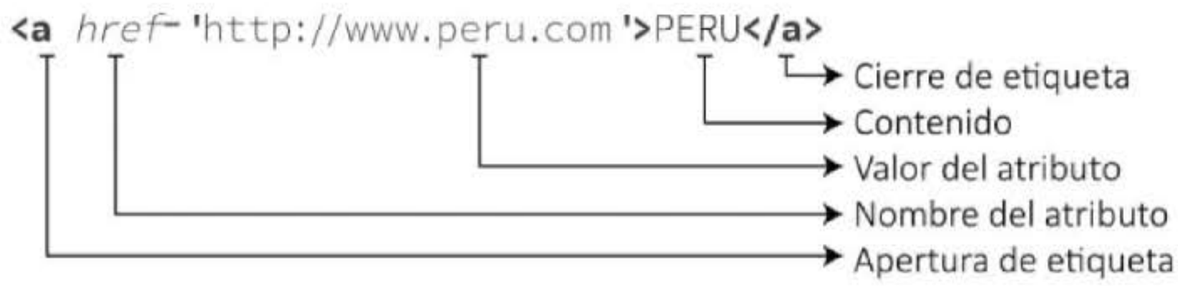


Figura 2.2 Estructura de una etiqueta HTML5.
Fuente: Elaboración propia.

Ejemplo 8

Véase el código para mostrar una imagen en un documento web especificando el alto y ancho del mismo.

```
<!DOCTYPE html>
<html>
    <head>
        <title>Manejo de atributos</title>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <img src='imagen/bandera.jpg' width='259' height='194'>
    </body>
</html>
```

Se debe tener en cuenta que las imágenes que se usen dentro de un proyecto web deben estar separados del documento web. Es así, que se recomienda crear una carpeta **imágenes** que permita organizar el proyecto de la mejor manera. Por otro lado, al hacer referencia a la imagen, esta se deberá colocar

en el directorio de origen. Opcionalmente, se puede especificar el tamaño de la imagen mediante el atributo **width** y **height**.

2.5 Identificación de los atributos en una etiqueta HTML5

Véase la descripción de los atributos pertenecientes a las etiquetas HTML5.

› **Atributo id**

Este atributo permite asignar un identificador único a un elemento que está en el script de una página web, el cual es analizado por el *Standard Generalized Mark-up Language* (SGML).

Por ejemplo, se puede usar el atributo **id** para distinguir las características de dos párrafos:

```
<p id='correo'>mtorres@gmail.com</p>
<p id='telefono'>982-565425</p>
```

Para entender mejor por qué se le asigna un identificador único a cada elemento HTML5 véase el siguiente listado sobre los usos que se le puede dar:

» Como selector para las hojas de estilo CSS.
» Como destino para vínculos de hipertexto.
» Como medio para hacer referencia a un elemento en particular desde un script.
» Como nombre de un elemento OBJECT implementado por un lenguaje de programación.
» Seguidamente, algunas recomendaciones al momento de definir el atributo **id**:
» Todo nombre de identificador debe empezar con una letra seguidamente de más caracteres e inclusive números.
» No está permitido el uso de caracteres especiales para el nombre del identificador a excepción del guion bajo (_).
» Las letras aceptadas son A...Z, a...z, números enteros y la combinación de ellos.
» Los nombres de los identificadores deben ser únicos por página.

Ejemplo 9

Véanse los códigos para aplicar el color azul a la especificación del correo electrónico, y el color rojo a la especificación del teléfono.

Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
#correo {
    color: blue;
}

#telefono {
    color: red;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Manejo de SMALL</title>
        <meta charset='UTF-8'>
        <link href='estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width,initial-scale=1.0'>
    </head>
    <body>
        <p id='correo'>mtorres@gmail.com</p>
        <p id='telefono'>982-565425</p>
    </body>
</html>
```

- **Atributo class**

  Este atributo permite asignar un nombre de clase a un elemento web. Podría decirse que el elemento web pertenece a esta clase. El atributo **class** se puede usar como selector para hojas de estilo, especialmente, cuando se desea asignar información de estilo a un conjunto de elementos web.

Ejemplo 10

Véase cómo aplicar el color verde al texto de tipo informativo; azul, al texto de tipo advertencia, y color rojo, al texto de error.

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
p.info { color: green }
p.advertencia { color: blue }
p.error { color: red }
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Manejo de CLASS</title>
        <meta charset='UTF-8'>
        <link href='estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <p class='info'> Mensaje de Información</p>
        <p class='advertencia'> Mensaje de Advertencia</p>
        <p class='error'> Mensaje de Error</p>
    </body>
</html>
```

› **Atributo title**

Permite establecer un título a un elemento web, es decir, ofrece información sobre el elemento cuando el puntero del mouse se posiciona sobre este.

Ejemplo 11

Véase cómo asignar un título a un enlace de correo electrónico. En este caso, cuando se posiciona el puntero del mouse encima del enlace, se debe mostrar el mensaje **Contáctenos**.

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de TITLE</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <address>
         <a>href='mailto:servicioalcliente@macro.com'
            title='Contáctenos'>Servicio al Cliente
         </a>
      </address>
   </body>
</html>
```

› **Atributo style**

Permite habilitar una especificación CSS directamente sobre el elemento web. Se recomienda su uso en pequeñas especificaciones dentro del documento web.

Ejemplo 12

Véase cómo asignar el estilo directo sobre un elemento de tipo párrafo aplicándole un tipo de letra Tahoma, tamaño de letra 14px y color azul.

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de STULE</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <p style='font-family:'Tahoma'; font-size: 14px; color:blue'>
         Uso del atributo Style
      </p>
   </body>
</html>
```

- **Atributo lang**
  Habilita la especificación del lenguaje contenido en un elemento web. El valor por defecto es "desconocido", ya que no se puede determinar cuál es el lenguaje que puede tener una página web.

Ejemplo 13

Véase cómo especificar el lenguaje EN para un párrafo en inglés y ES para un párrafo en español.

```
<!DOCTYPE html>
<html>
   <head>
      <title>Manejo de SMALL</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <p lang='en'>This is a paragraph in English.</p>
      <p lang='es'>Este es un párrafo en español.</p>
   </body>
</html>
```

2.6 Estructura global de un documento web

2.6.1 *Especificación DOCTYPE*

La declaración **DOCTYPE** le informa al navegador con qué versión del lenguaje de marcación está escrita dicha página. Esto lo hace basándose en la especificación *Document Type Definition* más conocido como DTD.

La DTD especifica las reglas usadas por el lenguaje de marcación, es decir, el HTML5 para el caso que se está estudiando, con el fin de que los navegadores puedan presentar el contenido correctamente y entender los elementos que lo componen. Véase una lista de definiciones anteriores al HTML5:

- **DOCTYPE para la versión XHTML 1.0 strict**
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

- **DOCTYPE para la versión XHTML 1.0 transitional**
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

- **DOCTYPE para la versión XHTML 1.0 frameset**
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">

- **DOCTYPE para la versión HTML 4.01 strict**
  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd">

- **DOCTYPE para la versión HTML 4.01 transitional**
  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">

- **DOCTYPE para la versión HTML 4.01 frameset**
  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
  "http://www.w3.org/TR/html4/frameset.dtd">

- **DOCTYPE para la versión HTML5**
  <!DOCTYPE html>

Como puede visualizarse en HTML5 se reduce totalmente la especificación del **DOCTYPE**, pero debe tenerse en cuenta que, en todas sus versiones, esta siempre se escribe en mayúsculas.

2.6.2 *Iniciando la estructura web con <html>*

Representa la raíz de un documento web, se utiliza para indicar al navegador que el documento que se visualiza está elaborado en el lenguaje HTML.

Es también la etiqueta raíz del resto de etiquetas del lenguaje, es decir, el resto de etiquetas del lenguaje HTML5 está contenida dentro de ella, con la única excepción de la etiqueta correspondiente al tipo de documento. El formato inicial que debe contener un documento web con HTML5 es:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
   </head>
   <body>
   </body>
</html>
```

La etiqueta **<html>** indica el inicio desde el cual se podrá estructurar el documento web. Para indicar el final de la especificación se debe colocar el cierre de la etiqueta, el cual es **</html>**.

2.6.3 *Definición de cabecera con la etiqueta <head>*

La etiqueta **<head>** permite definir la cabecera del documento web. Normalmente, contiene información que no es visible para el usuario. Esta etiqueta pertenece a el estructura principal de un documento

web; por lo tanto, debe ser colocado antes de la etiqueta **<body>** y dentro de **<html>** o **<!DOCTYPE html>**. El formato completo de un documento web se presenta en el siguiente script:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
   </head>
   <body>
   </body>
</html>
```

Ahora bien, véase una lista de los elementos que puede contener la cabecera:

- **<title>**
  Permite definir el título del documento web, este será mostrado en la barra de título de la ventana que contiene al navegador.




```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <title>Desarrollo web con HTML5 y CSS3</title>
   </head>
   <body>
   </body>
</html>
```

- **<base>**
  Permite definir una dirección por defecto para la carga de todos los links que contiene un documento web; por lo tanto, solo se debe especificar una sola vez por documento.

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <base target='_blank' href='http://www.miservidor.com.pe' />
   </head>
   <body>
   </body>
</html>
```

<meta>

Establece una clasificación de las páginas web, en especial, para las búsquedas por Internet. Los metadatos tratan sobre el contenido de la web, la descripción de la misma, el autor, etc. Esto contribuye a que más gente pueda encontrar la página por los metabuscadores como Google.

La información puede ser extraída de forma automática por los principales robots, bases de datos e indexadores automáticos de páginas web que existen en Internet. La etiqueta **<meta>** es una etiqueta vacía, es decir, comienza y termina en el mismo punto; por lo tanto, no hay que cerrarla con **</meta>**.

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <meta name='author' content='Edit. Macro' lang='es'>
      <meta name='description' content='Libro HTML5'>
      <meta name='keywords' content='HTML5,CSS'>
   </head>
   <body>
   </body>
</html>
```

- El atributo **name** define el tipo de información que proporciona la etiqueta **<meta>**. Los valores más comunes son:
  - **Author**: Dispone la asignación de información sobre el autor del documento web.
  - **Description**: Habilita la opción para poder describir el contenido web.
  - **Keywords**: Dispone el uso de palabras clave sobre el documento web.
  - **Robots**: Habilita la indexación de la página web. Presenta las siguientes opciones en su contenido: **index** para indicar al robot de búsqueda que es un índice de página y que lo añada a la base de datos; **all**, que indica que se indexen todas las páginas web, y **follow**, que señala que se tienen que seguir todos los enlaces que están implementados en la página web. Por ejemplo, **<meta name="robots" content="index,all,follow">**.
- El atributo **content** permite determinar el contenido según el tipo de información especificado en el atributo **name**.
- El atributo **lang** configura el idioma en un documento web. Por ejemplo, **<meta name="author" content="Edit. Macro" lang="es">**.
- El atributo **charset** permite especificar la codificación de caracteres del documento, es decir, podrá habilitar el uso de las tildes y la letra "ñ" de acuerdo a nuestra región. Por ejemplo, **<meta charset="utf-8" />**.

<link>

Permite asociar la página web a un archivo externo. Es usado frecuentemente para los enlaces con archivos CSS.

Primero, se debe contar con un archivo llamado **estilos.css**, el cual debe contener, por ejemplo, el siguiente código:

```
body {
    background-color:yellow;
}

p {
    color:blue;
}
```

El código anterior afectará al color de fondo del documento web y al color del párrafo que se muestre. Para esto se debe tener el siguiente código HTML:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilos.css' rel='stylesheet' type='text/css'/>
    </head>
    <body>
        <p>Manejo de estilos CSS dentro del mismo documento</p>
    </body>
</html>
```

Permite implementar un conjunto de estilos que solo pueden ser usados dentro del mismo documento web.

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <style type='text/css'>
            body {
                background-color:yellow;
            }
            p {
                color:blue;
            }
        </style>
    </head>
    <body>
        <p>Manejo de estilos CSS dentro del mismo documento</p>
    </body>
</html>
```

2.6.4 *La etiqueta <body>*

Esta etiqueta es un espacio de trabajo que contiene todo lo referente a la visibilidad dentro de un documento HTML5, como, por ejemplo, textos, hipervínculos, imágenes, tablas, listas, etc., es decir, todo lo que se pueda colocar en un documento web. La característica principal de la etiqueta **<body>** es su compatibilidad con todos los navegadores web actuales y de seguro que también con todos los que vengan.

La etiqueta **<body>** puede contener:

- Elementos estándares HTML5 como párrafos, formularios, imágenes, tablas, listas, etc.
- Enlaces a otras páginas web.
- Script de los Lenguajes de Programación como Java, ASP, PHP, etc.
- Objetos incrustados como audios, videos YouTube, etc.

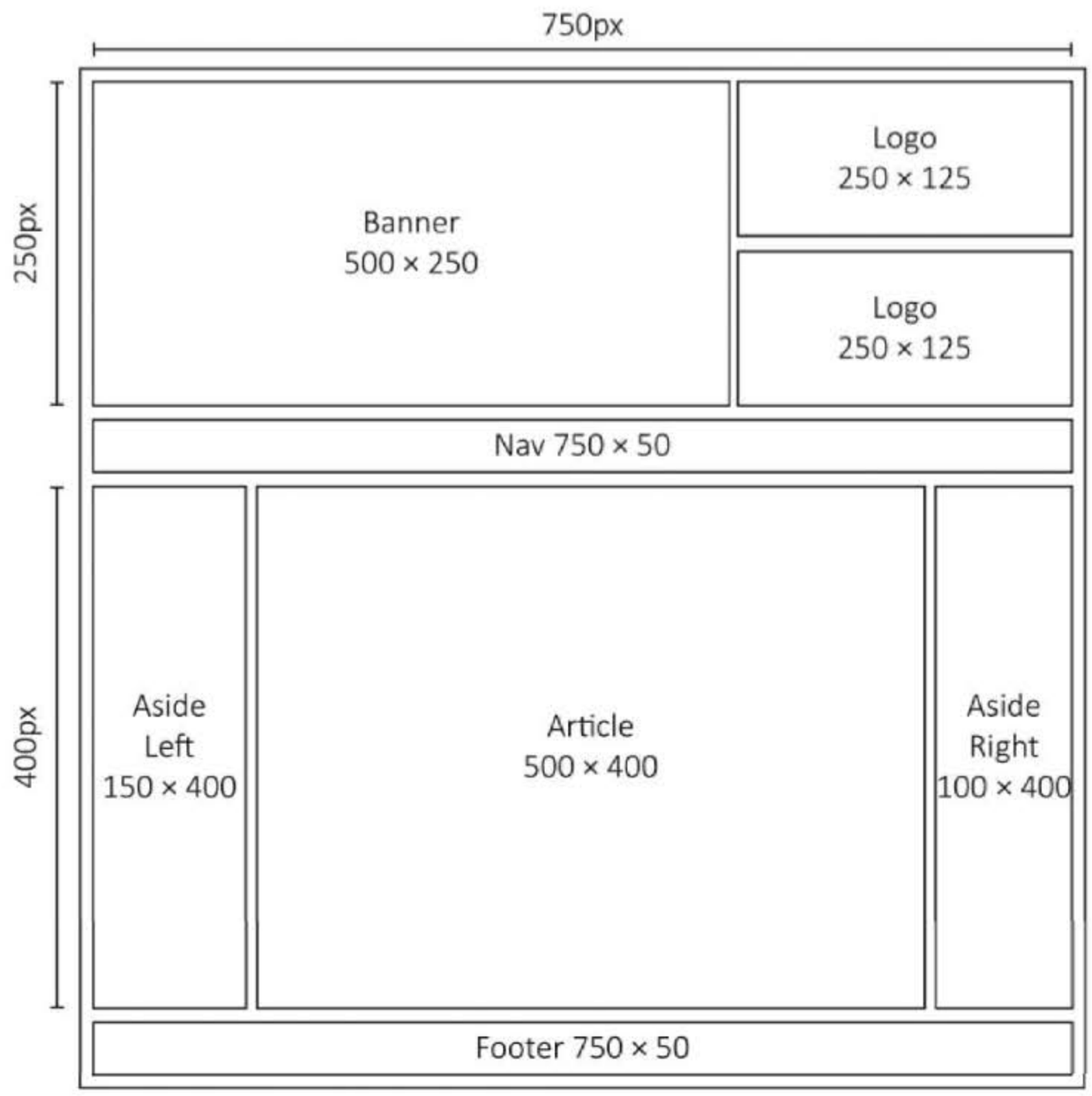
Caso desarrollado

Maquetación de un documento web HTML5 básico

Implemente una aplicación web con HTML5 que permita distribuir las etiquetas que componen el cuerpo del documento web usando estilos CSS.

Consideraciones:

Tenga en cuenta la siguiente distribución de elementos y el tamaño de cada uno, tal como se muestra en la siguiente imagen:



Solución:

1. Cree el archivo **estilos.css** con el siguiente código:

```
header{
    background-color: yellow;
    width: 750px;
    height: 250px;
}

nav{
    background-color: blue;
    width: 750px;
    height: 50px;
}

main{
    background-color: green;
    width: 750px;
    height: 400px;
}

footer{
    background-color: orange;
    width: 750px;
    height: 50px;
}

section#banner{
    background-color: red;
    width: 500px;
    height: 250px;
    float: left;
}

section#logo{
    background-color: gray;
    width: 250px;
    height: 125px;
    float: left;
}

section#login{
    background-color: magenta;
    width: 250px;
    height: 125px;
    float: left;
}
```

```
aside#left{
    background-color: purple;
    width: 150px;
    height: 400px;
    float: left;
}

aside#right{
    background-color: red;
    width: 100px;
    height: 400px;
    float: left;
}

article{
    background-color: lightblue;
    width: 500px;
    height: 400px;
    float: left;
}
```

2. Cree el documento web **index.html** correspondiente con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Maquetación Básica</title>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <section id='banner'>BANNER</section>
            <section id='logo'>LOGO</section>
            <section id='login'>LOGIN</section>
        </header>
        <nav>BARRA DE MENU</nav>
        <main>
            <aside id='left'>LADO IZQUIERDO</aside>
            <article>ARTICULO</article>
            <aside id='right'>LADO DERECHO</aside>
        </main>
        <footer>PIE DEL DOCUMENTO</footer>
    </body>
</html>
```

2.7 Estructurando el cuerpo del documento web

El contenido de **<body>** ha sido mejorado para HTML5 haciéndolo semántico, es decir, HTML5 ya ha puesto un orden a todos sus etiquetas, las cuales tienen una tarea específica y deja de lado las ambigüedades de otras versiones. Véanse las etiquetas más relevantes de HTML5.

Ejemplo 14

A continuación, se muestran las secciones visibles de la estructura de un documento web y sus respectivos códigos.

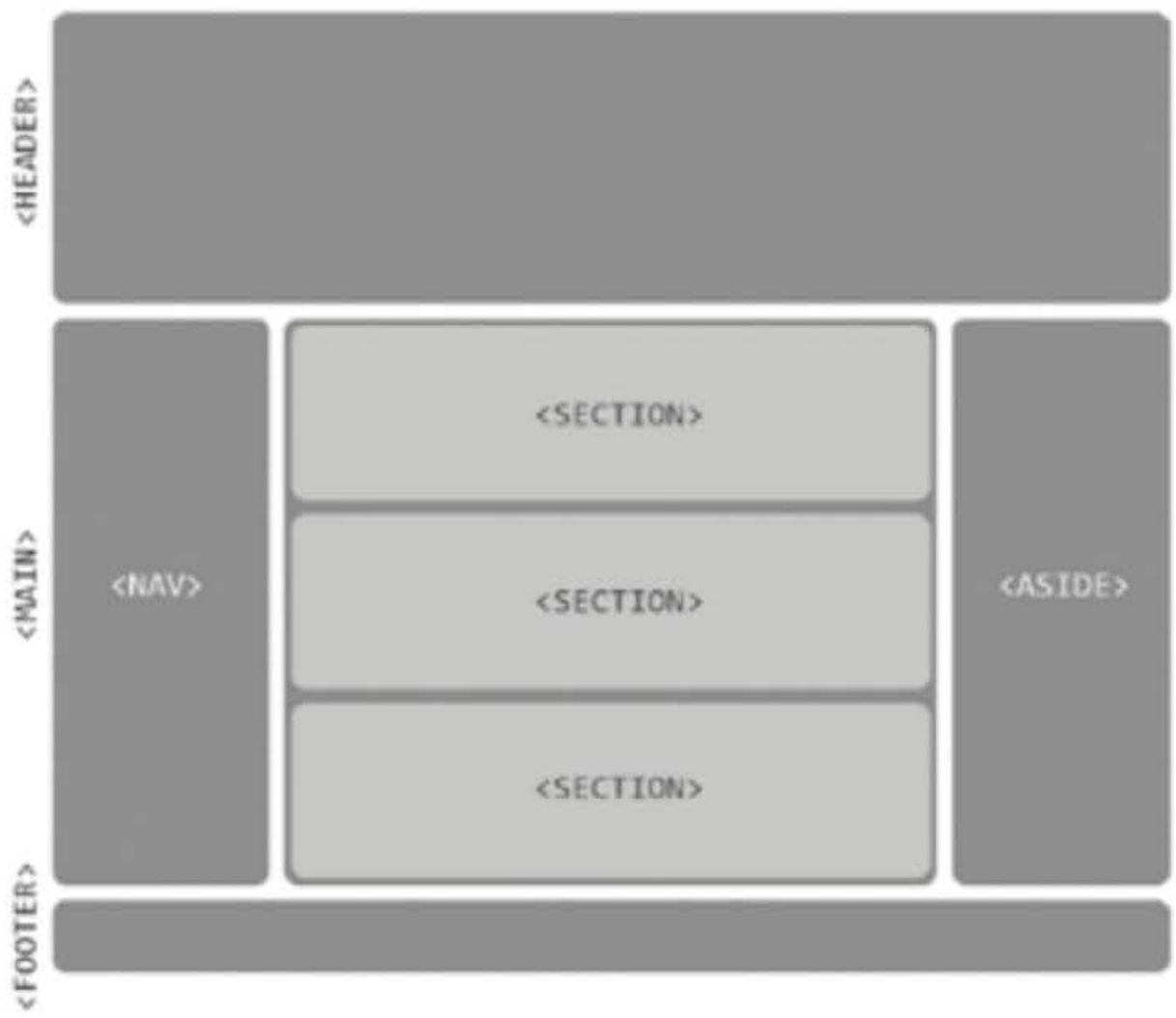


Figura 2.2 Partes de un documento web HTML5.
Fuente: Elaboración propia.

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
header{
    background-color: gray;
    border: 5px chartreuse;
    width: 770px;
    height: 200px;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}
```

```
main{
    width: 780px;
    height: 400px;
}

nav{
    background-color: gray;
    width: 150px;
    height: 390px;
    float: left;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}

section{
    background-color: gray;
    width: 450px;
    height: 390px;
    float: left;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}

section#arriba{
    background-color: burlywood;
    width: 440px;
    height: 120px;
    float: left;
    margin: 5px;
}

section#medio{
    background-color: burlywood;
    width: 440px;
    height: 120px;
    float: left;
    margin: 5px;
}

section#abajo{
    background-color: burlywood;
    width: 440px;
    height: 120px;
    float: left;
    margin: 5px;
```

```
}

aside{
    background-color: gray;
    width: 150px;
    height: 390px;
    float: left;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}

footer{
    background-color: gray;
    width: 770px;
    height: 50px;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}
```

En el documento web **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header></header>
        <main>
            <nav> </nav>
            <section>
                <section id='arriba'></section>
                <section id='medio'></section>
                <section id='abajo'></section>
            </section>
            <aside> </aside>
        </main>
        <footer>
        </footer>
    </body>
</html>
```

2.7.1 *La etiqueta <header>*

Permite mostrar información de cabecera útil para un documento web o para las secciones que tenga. No solo se usa para imprimir elementos en la parte superior del documento, sino también se emplea para el encabezado de cualquier sección del documento. Su formato es:

```
<header>
<!–contenido de la cabecera–>
</header>
```

Normalmente, un **<header>** se usa para:

- implementar encabezados en un documento web con las etiquetas **<h1...h6>**;
- mostrar títulos o subtítulos en un **section**, **article**, **aside**, **footer**, etc;
- presentar imágenes como *banners* o logotipos;
- mostrar un menú de opciones.

Ejemplo 15

Implemente el siguiente encabezado usando estilos CSS.



Tenga en cuenta que el archivo **estilo.css** debe encontrarse dentro de una carpeta llamada **css**.

Códigos:

En el archivo **estilos.css** se agrega el siguiente código:

```
header {
    background: #333;
    height: 100px;
}

h1{
    font-family: tahoma;
    color: white;
    line-height: 60px;
    margin: 10px;
}

h4{
    font-family: tahoma;
    color: yellow;
    margin: 10px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <h1>Diseño Web con HTML5</h1>
            <h4>Editorial MACRO</h4>
        </header>
    </body>
</html>
```

Ejemplo 16

Implemente un encabezado en el que se muestre una imagen.



Tome en cuenta que el archivo **estilo.css** debe encontrarse en una carpeta **css** dentro del proyecto web, asimismo la imagen debe ubicarse en la carpeta **img**.

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
img {
    width: 750px;
    height: 120px;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
```

```
    <body>
        <header>
            <img src='img/banner_peru.jpg'/>
        </header>
    </body>
</html>
```

Ejemplo 17

Implemente el siguiente encabezado:



Debe tenerse en cuenta que el archivo **estilo.css** debe encontrarse dentro de una carpeta llamada **css**.

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
body {
    margin: 0;
    padding: 0;
    font-family: Helvetica, Arial, sans-serif;
    color: #666;
    background: #f2f2f2;
    font-size: 1em;
    line-height: 1.5em;
}

nav {
    float: right;
}

nav ul {
    margin: 0;
    padding: 0;
    list-style: none;
    padding-right: 20px;
}

nav ul li {
    display: inline-block;
    line-height: 40px;
}

nav ul li a {
    display: block;
    padding: 0 10px;
    text-decoration: none;
}
```

```
nav ul li a:hover {
    background: #0b76a6;
}
```

En el documento web se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <nav>
                <ul>
                    <li><a href='principal.html'>Principal</a></li>
                    <li><a href='quienes.html'>Quienes somos</a></li>
                    <li><a href='contacto.html'>Contactos</a></li>
                </ul>
            </nav>
        </header>
    </body>
</html>
```

2.7.2 *La etiqueta <section>*

El uso de la etiqueta **<section>** agrupa elementos según su contenido estableciendo, como su nombre lo dice, una sección. Esta etiqueta crea bloques a partir de diferentes elementos, donde se puede incluir un encabezado y un pie. Su formato es:

```
<section>
<!– contenido de section –>
</section>
```

Principales características de la etiqueta **<section>**:

- Un documento web podría dividirse en secciones para un listado de noticias, información de contactos, etc.
- Asegura que todo lo que se encuentre dentro de una etiqueta **<section>** tenga relación.
- Se usa para crear contenidos independientes dentro de un documento web.

Ejemplo 18

Implemente la siguiente sección dentro de un documento web haciendo uso de CSS para el formateo de los textos. Asimismo, la sección debe tener un ancho de 400px.

Manejo de SECTION

¿Cómo se usa?

El uso de la etiqueta <section> agrupa elementos según su contenido estableciendo, como su nombre lo dice, una sección. Esta etiqueta crea bloques a partir de diferentes elementos, donde se puede incluir un encabezado y un pie.

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```css
/* Estilo de Fondo */
body {
    background: #d3d3d3;
    font-size: 100%;
    margin: 10px;
    font-family:tahoma;
    color: #444;
}

/* Estilo para la seccion */
section{
    max-width:400px;
}
```

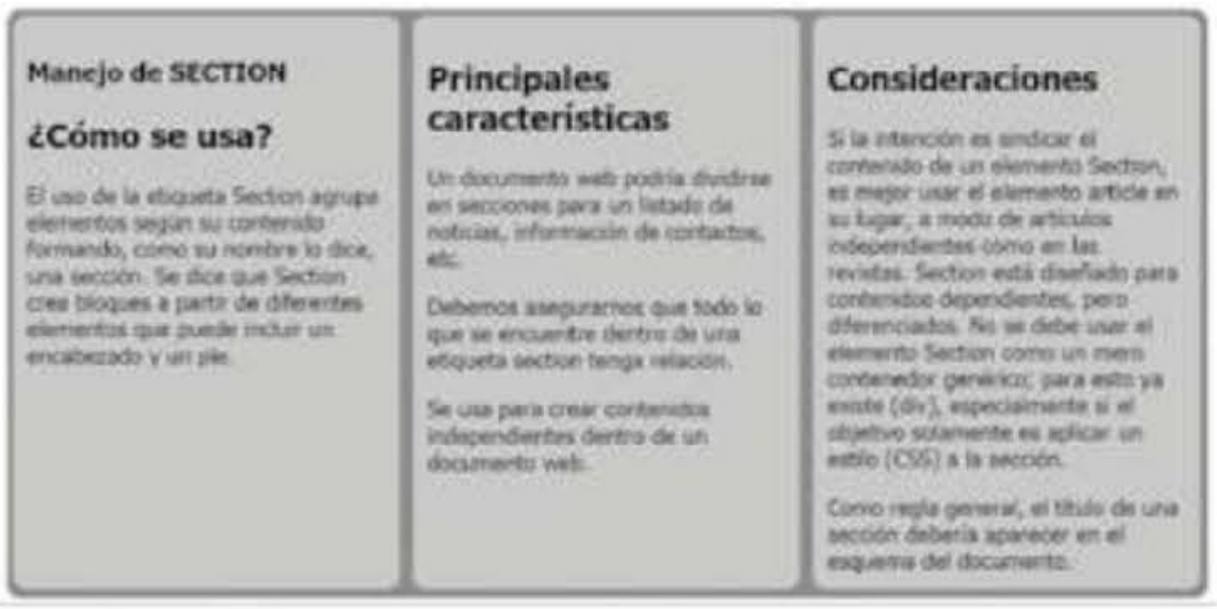
En el documento web se escribe el siguiente código:

```html
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h1>Manejo de SECTION</h1>
            </header>
```

```
        <article>
            <h2>¿Cómo se usa?</h2>
            <p>
                El uso de la etiqueta <section> agrupa elementos según su contenido
                estableciendo, como su nombre lo dice, una sección. Esta etiqueta
                crea bloques a partir de diferentes elementos, donde se puede incluir
                un encabezado y un pie.
            </p>
        </article>
    </section>
</body>
</html>
```

Ejemplo 19

Implemente la siguiente sección dentro de un documento web en la cual se muestren textos divididos en tres secciones del mismo tamaño.



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
*{
    font-family: tahoma;
}

section{
    background-color: gray;
    width: 870px;
    height: 400px;
    float: left;
    margin: 5px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}
```

```
section#izquierda{
    background-color: burlywood;
    width: 260px;
    height: 370px;
    float: left;
    padding: 10px;
}

section#medio{
    background-color: burlywood;
    width: 260px;
    height: 370px;
    float: left;
    padding: 10px;
}

section#derecha{
    background-color: burlywood;
    width: 260px;
    height: 370px;
    float: left;
    padding: 10px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <section id='izquierda'>
                <header>
                    <h1>Manejo de SECTION</h1>
                </header>
                <article>
                    <h2>¿Cómo se usa?</h2>
                    <p>
                        El uso de la etiqueta Section agrupa elementos según su contenido
                        formando, como su nombre lo dice, una sección. Se dice que Section
                        crea bloques a partir de diferentes elementos que puede incluir un
                        encabezado y un pie.
                    </p>
                </article>
            </section>
```

```
<section id='medio'>
    <article>
        <h2>Principales características</h2>
        <p>
            Un documento web podría dividirse en secciones para un listado de
            noticias, información de contactos, etc.
        </p>
        <p>
            Debemos asegurarnos que todo lo que se encuentre dentro de una
            etiqueta section tenga relación.
        </p>
        <p>
            Se usa para crear contenidos independientes dentro de un documento web.
        </p>
    </article>
</section>
<section id='derecha'>
    <article>
        <h2>Consideraciones</h2>
        <p>
            Si la intención es sindicar el  contenido de un elemento Section,
            es mejor usar el elemento article en su lugar, a modo de artículos
            independientes como en las  revistas. Section está diseñado para
            contenidos dependientes, pero diferenciados. No se debe usar el
            elemento Section como un mero contenedor genérico; para esto ya
            existe (div), especialmente si el  objetivo solamente es aplicar un
            estilo (CSS) a la sección. Como regla general, el título de una
            sección debería aparecer en el esquema del documento.
        </p>
    </article>
</section>
</section>
</body>
</html>
```

2.7.3 *La etiqueta <nav>*

Según la W3C la etiqueta NAV representa una sección de un documento web que contiene enlaces o links a otras páginas. Hay que tener en cuenta que toda página web debe contar con vínculos para hacerlo navegable para el usuario. Su formato general se presenta a continuación:

```
<nav>
    <a href='principal.html'>principal</a>
    <a href='conocenos.html'>conócenos</a>
    <a href='imagenes.html'>imágenes</a>
</nav>
```

Todos los enlaces web no necesariamente deben encontrarse dentro de la etiqueta **<nav>**, puesto que hay etiquetas del HTML5 que pueden soportarlo, como es el caso de **<footer>**, que puede referenciar a enlaces que mencionen, por ejemplo, las políticas de la página, el correo del webmaster o los enlaces dentro del mismo documento.

Una de las razones que da el W3C para utilizar este elemento tiene que ver con el creciente uso de los dispositivos móviles, las *tablets* y otros dispositivos que permiten la navegación por Internet. De esta manera, cuando en uno de estos dispositivos se abre una página con una etiqueta **<nav>**, un enlace puede no pintarse en la pantalla y, en vez de ello, mostrar el despliegue de un menú cuando se pincha sobre él; así, de esta forma, se mostrará únicamente el contenido más relevante y el contenido menos importante se ocultará, que es una característica que presentan los dispositivos móviles frente a los documentos web.

Principales características de la etiqueta **<nav>**:

- Especifica claramente cuáles son los elementos de navegación de un documento web.
- Se puede configurar un navegador para que, al iniciar un documento web, salte al menú del sitio web, esto dejaría obsoleto a los enlaces "Saltar enlaces" o "Ir al contenido principal".
- Un navegador web visto desde un dispositivo, por ejemplo, un móvil o una *tablet*, puede mostrar los elementos de navegación de forma independiente o cuando el usuario pulse una tecla del dispositivo.
- El elemento **<nav>** funciona tal como lo haría una etiqueta ** ** del HTML4. Ya no será necesario escribir ** ** o ** **.

Ejemplo 20

Implemente el siguiente menú de opciones de forma horizontal:



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
body {
    margin: 0em;
    padding: 0em;
    background: #1EB0D7;
    font-family: 'Tahoma';
    font-size: 10pt;
    color: #525252;
}

nav li {
    display: inline-block;
}
```

```
nav a {
    display: block;
    float: left;
    margin-right: 0.12em;
    padding: 1em 1.5em;
    background: #178EAE;
    box-shadow: inset 0px -2px 0px 0px rgba(0,0,0,0.25);
    text-decoration: none;
    text-shadow: 1px 1px 0px #0C749C;
    text-transform: uppercase;
    font-size: .90em;
    color: #FFFFFF;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <nav>
            <ul>
                <li><a href='#'>Inicio</a></li>
                <li><a href='#'>Nosotros</a></li>
                <li><a href='#'>Servicios</a></li>
                <li><a href='#'>Recursos</a></li>
                <li><a href='#'>Contáctenos</a></li>
            </ul>
        </nav>
    </body>
</html>
```

Ejemplo 21

Implemente el siguiente menú de opciones de forma vertical:



Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
body {
    background: #d3d3d3;
    font-size: 100%;
    margin: 10px;
    font-family:tahoma;
    color: #444;
}

/* Estilo para la lista */
nav ul {
    background: #e8e8e8;
    padding: 0;
    width: 25%;
}

/* Estilo para los elementos de la lista */
nav li a {
    display: block;
    padding: 1.25em;
    border-bottom: 1px solid #D2D2D2;
}

/* Estilo al posicionar el puntero encima del enlace*/
nav a:hover {
    color: #fff;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <nav>
            <ul>
                <li><a href='#'>Home</a></li>
                <li><a href='#'>Productos</a></li>
                <li><a href='#'>Empresa</a></li>
                <li><a href='#'>Contáctenos</a></li>
            </ul>
        </nav>
    </body>
</html>
```

2.7.4 *La etiqueta <aside>*

Su trabajo es similar a la etiqueta **<section>**, la diferencia radica en que siempre será un bloque añadido a los bloques que tiene al lado. Se podría decir que son datos extra dentro de la estructura del documento web basada en HTML5. Su formato es:

```
<aside>
    <!– contenido de la sección aside –>
</aside>
```

Ejemplo 22

Implemente el siguiente formato usando **<aside>** de tal forma que el lado izquierdo y derecho muestren un texto descriptivo mientras que la parte central muestre una imagen.



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
body {
    font-family: 'Tahoma';
}

section{
    width: 880px;
    height: 400px;
}

aside#left{
    background-color: #e8e8e8;
    width: 200px;
    height: 400px;
    float: left;
    margin: 3px;
    padding: 10px;
}
```

```
aside#center{
    background-color: #e8e8e8;
    width: 400px;
    height: 400px;
    float: left;
    margin: 3px;
    padding: 10px;
}

aside#right{
    background-color: #e8e8e8;
    width: 200px;
    height: 400px;
    float: left;
    margin: 3px;
    padding: 10px;
}
```

En el documento web **index.html** se escribe el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <aside id='left'>
                <h1>Etiqueta ASIDE</h1>
                Su trabajo es similar a la  etiqueta SECTION, lo diferencia  que siempre
                será un bloque añadido a los bloques que tiene al lado. Se podría
                decir que son datos extra dentro de la estructura del documento web
                basado en HTML5.
            </aside>
            <aside id='center'>
                <img src='img/logo_html.png' width='400' height='400'/>
            </aside>
            <aside id='right'>
                <h1>También...</h1>
                El ASIDE representa una sección de una página que  tiene relación
                con el contenido que está mostrando la página web y que podría ser
                considerado por separado de ese contenido. Estas secciones son a menudo
                representadas como barras laterales y también llamadas Sidebar.
            </aside>
        </section>
    </body>
</html>
```

2.7.5 *La etiqueta <footer>*

Representa el "pie" o una sección de un documento web. Se parece mucho al pie de página de un documento en Word. Un **<footer>** puede insertarse varias veces dentro de un mismo documento. Hay que tener en cuenta que cuando el **<footer>** está al final del documento web puede contener información sobre el webmaster, las condiciones o términos para el uso del documento o los derechos del autor, mientras que si el **<footer>** se encuentra colocado en una **<section>** o **<article>**, contendrá datos finales propios de la sección o artículo correspondiente, así como también la fecha u hora de la publicación.
Su formato es:

```
<footer>
    <!-contenido del pie->
</footer>
```

¿Qué podemos colocar en un **<footer>**?

- Nombre del autor del documento o diseñador web.
- Enlaces a contenidos relacionados con el documento web.
- Información de *copyright* o avisos legales.
- Información de contacto con la empresa usando la nueva etiqueta HTML5 llamada **<address>**.

Ejemplo 23

Implemente el siguiente pie a un documento web.



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
/* Estilo para todo el documento */
body{
    font-size:13px;
    font-family:Tahoma, Geneva, sans-serif;
    color:#FFF;
    background-color:#fff;
}

/* Estilo para la alineación de los textos*/
p#izquierda{
    float:left;
}

p#derecha{
    float:right;
}
```

```
a {
    color: yellow;
}

/* Estilo para el pie */
footer {
    height: 40px;
    margin:10px;
    padding: 10px;
    background-color:#000;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <footer>
            <p id='izquierda'>Copyright &copy; 2018 - Todos los derechos reservados-
            <a href='http://www.editorialmacro.com'>Editorial Macro</a></p>
            <p id='derecha'>Diseñado por Lic. Manuel Torres R.</p>
        </footer>
    </body>
</html>
```

2.8 Elementos internos

2.8.1 *Etiqueta <article>*

Representa un contenido de regular importancia en el documento web que puede ser distribuido de forma independiente. Quiere decir, seguirá manteniendo su sentido semántico en otros documentos web. Su formato es:

```
<article>
    <!- contenido del article ->
</article>
```

Principales características de la etiqueta **<article>**:

- Un documento web podría contener un número indeterminado de artículos, ya que cada uno posee independencia frente a los demás artículos, inclusive dentro del mismo documento web.
- En un artículo se puede colocar etiquetas como **<header>**, **<section>** y **<footer>**.
- La diferencia entre **<article>** y **<section>** es que la primera tiene una cultura semántica más fuerte que la segunda.

Ejemplo 24

Elabore un documento web en el cual se visualice dos artículos: el primero dedicado a las características de la etiqueta **<section>**, y el otro, a la etiqueta **<article>**.



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
/* Estilo para todo el documento */
body {
    font-family: 'Tahoma';
}
/* Estilo para el título del documento */
p {
    margin: 8px;
    text-align:justify;
}
/* Estilo para los articulos */
article {
    margin: 10px;
    padding: 5px;
    float:left;
    width: 30%;
    height: 450px;
    background:#CFC;
    text-align:center;
}
/* Estilo para definir ancho y alto de la imagen */
img{
    width:400px;
    height:250px;
}
```

En el archivo **index.html** se escribe el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <h1>Desarrollo con HTML5</h1>
        <ARTICLE>
            <h3>Etiqueta SECTION</h3>
            <p>
            Un documento web podría dividirse en secciones para un listado de noticias,
            información de contactos, etc.
            </p>
            <p>
            Debemos asegurarnos que todo lo que se encuentre dentro de una etiqueta
            SECTION tenga relación.
            </p>
            <p>
            Se usa para crear contenidos independientes dentro de un documento web.
            </p>
            <img src='img/seccion.png' />
        </article>
        <ARTICLE>
            <h3>Etiqueta ARTICLE</h3>
            <p>
                Un documento web podría contener un número indeterminado de artículos,
                ya que cada uno posee independencia frente a los demás artículos
                inclusive dentro del mismo documento web.
            </p>
            <p>
                En un artículo podrías colocar etiquetas como HEADER, SECTION y FOOTER.
            </p>
            <p>
                La diferencia entre ARTICLE y SECTION es que un ARTICLE tiene una cultura
                semántica más fuerte que un SECTION.
            </p>
            <img src='img/article.png' />
        </article>
    </body>
</html>
```

2.8.2 *La etiqueta <hgroup>*

Permite agrupar un conjunto de elementos pertenecientes a un conjunto de encabezados para que puedan ser manejados como un solo bloque. Su formato general es:

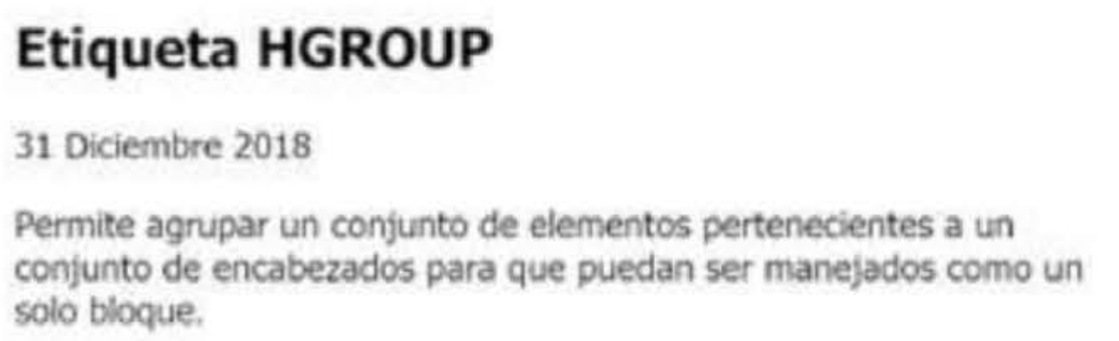
```
<hgroup>
    <h1> … </h1>
    <h2> … </h2>
</hgroup>
```

Principales características:

- Las etiquetas que hay que usar dentro del **<hgroup>** pueden ser del **<h1>** hasta el **<h6>**, que sirven para asignar un formato a los títulos de nuestras secciones dentro de un documento web.
- No compromete el valor semántico de cada etiqueta.
- Dentro de un **<hgroup>** solo se pueden implementar etiquetas de tipo encabezado.
- Es reconocida como la etiqueta que establece frases.
- También permite agrupar metadata o algún otro texto auxiliar cuyo contenido no se desea que aparezca en la estructura del documento.

Ejemplo 25

Implemente un documento web que permita agrupar el título y la fecha, y que imprima un texto fuera de la agrupación.



Códigos:

En el archivo **estilos.css** se agrega el siguiente código:

```
body{
    font-size:13px;
    font-family:Tahoma;
    color:#000;
}

section{
    width: 400px;
}
```

En el documento web **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <section>
         <hgroup>
            <h1>Etiqueta HGROUP</h1>
            <p><time datetime='2018-12-31'>31 Diciembre 2018</time></p>
         </hgroup>
            <p>Permite agrupar un conjunto de elementos pertenecientes a un conjunto
            de encabezados para que puedan ser manejados como un solo bloque.</p>
      </section>
   </body>
</html>
```

Ejemplo 26

Implemente el siguiente pie al documento web agrupando el título, la línea de división y la fecha.



Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
body{
   font-family:Tahoma;
   color: white;
   text-align: center;
}

section{
   width: 750px;
   background: black;
}

span#titulo{
   font-size:15px;
   margin: 5px;
}
```

```
span#subtitulo{
    font-size:12px;
    margin: 5px;
}

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

En el documento web **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <hgroup>
                    <span id='titulo'>DESARROLLO WEB </span>
                    <span id='subtitulo'>HTML5 y CSS3</span>
                    <hr>
                </hgroup>
                <p><time datetime='2018-12-31'>31 de diciembre del 2018</time></p>
            </header>
            <p>Todos los derechos reservados @2018-EditorialMacro</p>
        </section>
    </body>
</html>
```

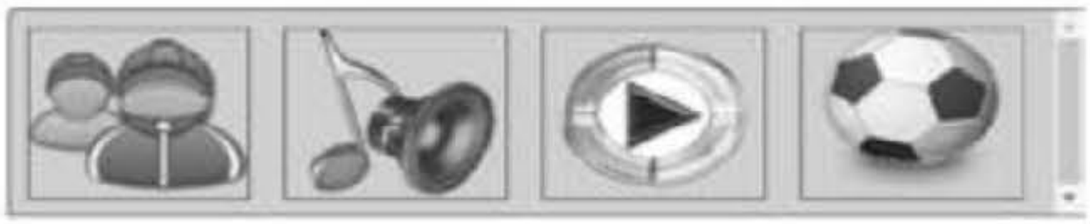
2.8.3 *La etiqueta <figure>*

Representa un contenido independiente de otros elementos web. Su uso no se limita únicamente a una imagen, sino también a una ilustración, diagrama, fragmento de código o un esquema al cual se hace referencia en el texto principal del documento web. Su formato general es:

```
<figure>
    <img scr='image.jpg'> … </img>
</figure>
```

Ejemplo 27

Implemente una aplicación web que muestre cuatro imágenes dentro de un marco.



Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

```
/*Estilo de la barra*/
.barra-panel {
    overflow:auto;
    position:absolute;
    left:10px;
    top:80%;
    width:98.5%;
    right:20;
    height:120px;
    z-index:1;
}

.borde-Panel {
    border: 1px solid RGB(0,0,0);
    background: RGB(215,201,168);
}

.punta-Izq {
    border-top-left-radius: 5px;
}

.punta-Der {
    border-top-right-radius: 5px;
}

.barra-Scroll {
    width: 3630px;
}

/*Estilo del marco de las imágenes*/
figure {
    border: 1px solid RGB(0,0,0);
    width: 130px;
    height: 100px;
    float: left;
    margin: 10px;
}
```

```
/*Estilo de las imágenes*/
img{
    width: 130px;
    height: 100px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section class='barra-panel borde-Panel punta-Izq punta-Der'>
            <section class='barra-Scroll'>
                <figure><img src='img/usuario.png'></figure>
                <figure><img src='img/audio.png'></figure>
                <figure><img src='img/video.png'></figure>
                <figure><img src='img/juegos.png'></figure>
            </section>
        </section>
    </body>
</html>
```

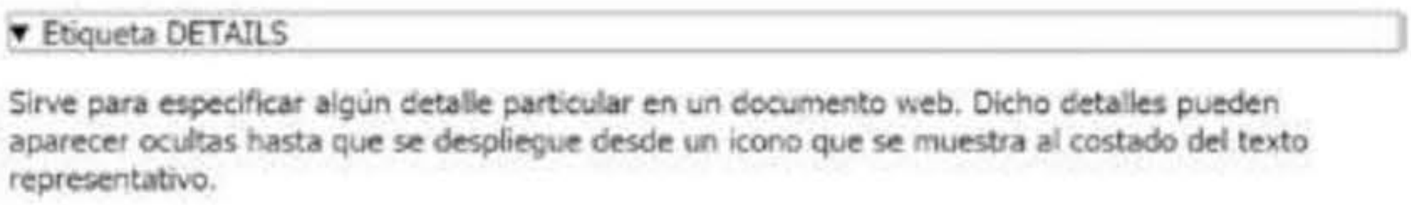
2.8.4 *La etiqueta* <details>

Sirve para especificar algún detalle en particular en un documento web. Dicho detalle puede aparecer oculto hasta que se despliegue un elemento a través de un ícono que se muestra al costado del texto representativo. Su formato general es:

```
<details>
    <summary>
</details>
```

Ejemplo 28

Implemente una aplicación web que muestre un texto en detalle.



Códigos:

En el archivo **estilos.css** se coloca el siguiente código:

```
body{
    font-family: tahoma;
    font-size: 12px;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <details>
            <summary>Etiqueta DETAILS</summary>
            <p>Sirve para especificar algún detalle particular en un documento web.
            Dicho detalles pueden aparecer ocultas hasta que se despliegue desde un
            icono que se muestra al costado del texto representativo.</p>
        </details>
    </body>
</html>
```

2.9 Comentarios en HTML5

Los comentarios son líneas de código que no son interpretados por el navegador web. Por lo tanto, su integración dentro del documento web responde a otras necesidades no necesariamente semánticas de HTML5. Presenta dos formatos, estos son:

```
<!– Comentario de una línea–>
```

Se caracteriza por pertenecer siempre a una misma línea, son comentarios normalmente cortos que no afectan visualmente al código HTML5 de un documento web.

```
<!– Comentario
de varias
lineas –>
```

Se caracteriza por usar más de una línea. La idea es colocar cierto número de líneas de código que no será interpretado por el navegador web.

Véase una lista de posibilidades en las que se puede hacer uso de los comentarios dentro de un documento web:

- Mostrar un mensaje simple en la cual se explique alguna sección del código:

```
<!–Implementando script con HTML5–>
```

- Alertar al desarrollador sobre los cuidados que debe tener en una sección del código:

```
<!– Aquí ingrese un código válido –>
```

- Especificar una sección del documento web como una cabecera, un artículo, etc.

```
<!–Aquí empieza la cabecera–>
<!–Fin de la cabecera–>
```

- Anular la interpretación del código desde el navegador, es decir, indicar que no se ejecutará ni se interpretará el código enmarcado en el comentario.

```
<body leftmargin='20px' topmargin='20px' rightmargin='20px' bottommargin='20px'>
<!–
   <section>
La etiqueta BODY es un espacio de trabajo que contiene todo lo visible de un
documento HTML hacia el usuario; tal como textos, hipervínculos, imágenes,
tablas, listas y todo lo que se pueda colocar en un documento web. La
característica principal de la etiqueta BODY es que es compatible con todos los
navegadores web actuales y es seguro que también con los que vengan.
   </section>
–>
</body>
```

2.10 Otras etiquetas

2.10.1 *La etiqueta <hr>*

Permite separar el contenido entre párrafos usando una línea horizontal, la cual tiene características que pueden ser controladas por el diseñador web. Su formato es **<hr>** o también **</hr>**. Una de las características de la etiqueta **<hr>** es que no es necesario cerrar la etiqueta, ya que no posee contenido. Véase cómo cambiar sus características:

Ejemplo 29

- **<hr** *size*="2"**>**
  Define el grosor de la línea a dos píxeles.

- <**hr** size="10">
  Define el grosor de la línea a diez píxeles.

- <**hr** color="#FF0000">
  Define el color de la línea.

- <**hr** width="75%">
  Definición del ancho de la línea, para este caso se le asigna el 75% del total del ancho que tiene el documento web. Cuando no se especifica un ancho, automáticamente se asigna 100%.

- <**hr** noshade>
  Definición de una línea sin sombra.

2.10.2 *La etiqueta <blockquote>*

Permite establecer citas como tipo de párrafo pertenecientes a un sitio web diferente, destacando dichas citas con una especie de sangrado en los lados derecha e izquierda. Su formato es:

```
<blockquote cite='' title=''>
   <!– contenido del blockquote –>
</blockquote>
```

Donde:

- **cite:** Es la propiedad que permite definir la dirección URL de donde proviene el párrafo.
- **title:** Es la propiedad que establece un título al bloque de párrafo que será mostrado al posicionar el puntero del mouse sobre el párrafo.

Ejemplo 30

Implementa el siguiente documento web haciendo uso de la etiqueta **<blockquote>**.

Etiqueta BLOCKQUOTE

El elemento blockquote permite a los autores insertar citas en forma de bloques de contenido, usualmente compuestas por un párrafo, un grupo de párrafos o un conjunto de muchos otros elementos, incluyendo imágenes (img), tablas (table) y artículos (article), entre otros. Esta es la principal diferencia entre este elemento y q, que está diseñado para citar únicamente líneas de texto.

Recuperado de, HTML QUICK

Códigos:

En el documento **estilos.css** se coloca el siguiente código:

```
blockquote{
    font-family: tahoma;
    font-size: 12px;
}
```

El documento **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <BLOCKQUOTE>
         <p>
            Etiqueta BLOCKQUOTE
         </p>
         <p>
          El elemento blockquote permite a los autores insertar citas en forma de
          bloques de contenido, usualmente compuestas por un párrafo, un grupo de
          párrafos o un conjunto de muchos otros elementos, incluyendo imágenes
          (img), tablas (table) y artículos (article), entre otros. Esta es la
          principal diferencia entre este elemento y q, que está diseñado para citar
          únicamente líneas de texto.
         </p>
         <footer>
            <p>
             Recuperado de, <a href="http://www.htmlquick.com/es/reference/tags/
             blockquote.html'>HTML QUICK</a>
          </p>
         </footer>
      </BLOCKQUOTE>
   </body>
</html>
```

2.10.3 *La etiqueta <cite>*

Permite insertar información referente a una determinada cita en el documento web. Esta puede ser el nombre del autor, una obra literaria, etc. Su formato es:

```
<cite>
   <!- contenido de cite ->
</cite>
```

A continuación, se muestran las características más importantes de esta etiqueta:

- Muestra el texto en cursiva para diferenciarlo de los demás;
- No permite hacer saltos de línea ni anterior ni posterior dentro de un documento web.

Ejemplo 31

Implemente el siguiente párrafo en un documento web haciendo uso de la etiqueta **<cite>**.

> El elemento *cite* representa a una cita del título de una obra. Este elemento puede ser usado para citar los *títulos de libros, artículos científicos, ensayos, pinturas, esculturas, obras de teatro, canciones, películas, series de TV, videojuegos, etc.*

Códigos:

En el archivo **estilos.css** se escribe el siguiente código:

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

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <p>
        El elemento <cite>cite</cite> representa a una cita del título de una obra.
        Este elemento puede ser usado para citar los <cite>títulos de libros,
        artículos científicos, ensayos, pinturas, esculturas, obras de teatro,
        canciones, películas, series de TV, videojuegos, etc</cite>.
        </p>
    </body>
</html>
```

2.10.4 *La etiqueta <address>*

Permite definir información de contacto del autor o los autores de un documento web. También podría definirse en la sección o artículo en la que aparece. Hay que considerar que los navegadores modifican el aspecto **<address>**, aplicando cursiva a todo su contenido. Su formato es:

```
<address>
    <!-contenido del address->
</address>
```

¿Qué podemos colocar en un **<address>**?

- La información acerca de la empresa dueña del documento web.
- El autor del artículo o del sitio web, también conocido como webmaster.
- Los números de teléfono, códigos postales y direcciones de correo electrónico.

Ejemplo 32

Implemente el siguiente párrafo haciendo uso de la etiqueta **<address>**.

Etiqueta ADDRESS

El contenido de un elemento address provee información de contacto para el autor de la sección o documento, como una dirección postal, una dirección de correo electrónico, un número telefónico, una ciudad, un país, un código postal, etc.

Tomado de: HTML QUICK
Libro HTML5 y CSS3
Editorial Macro-Perú

Códigos:

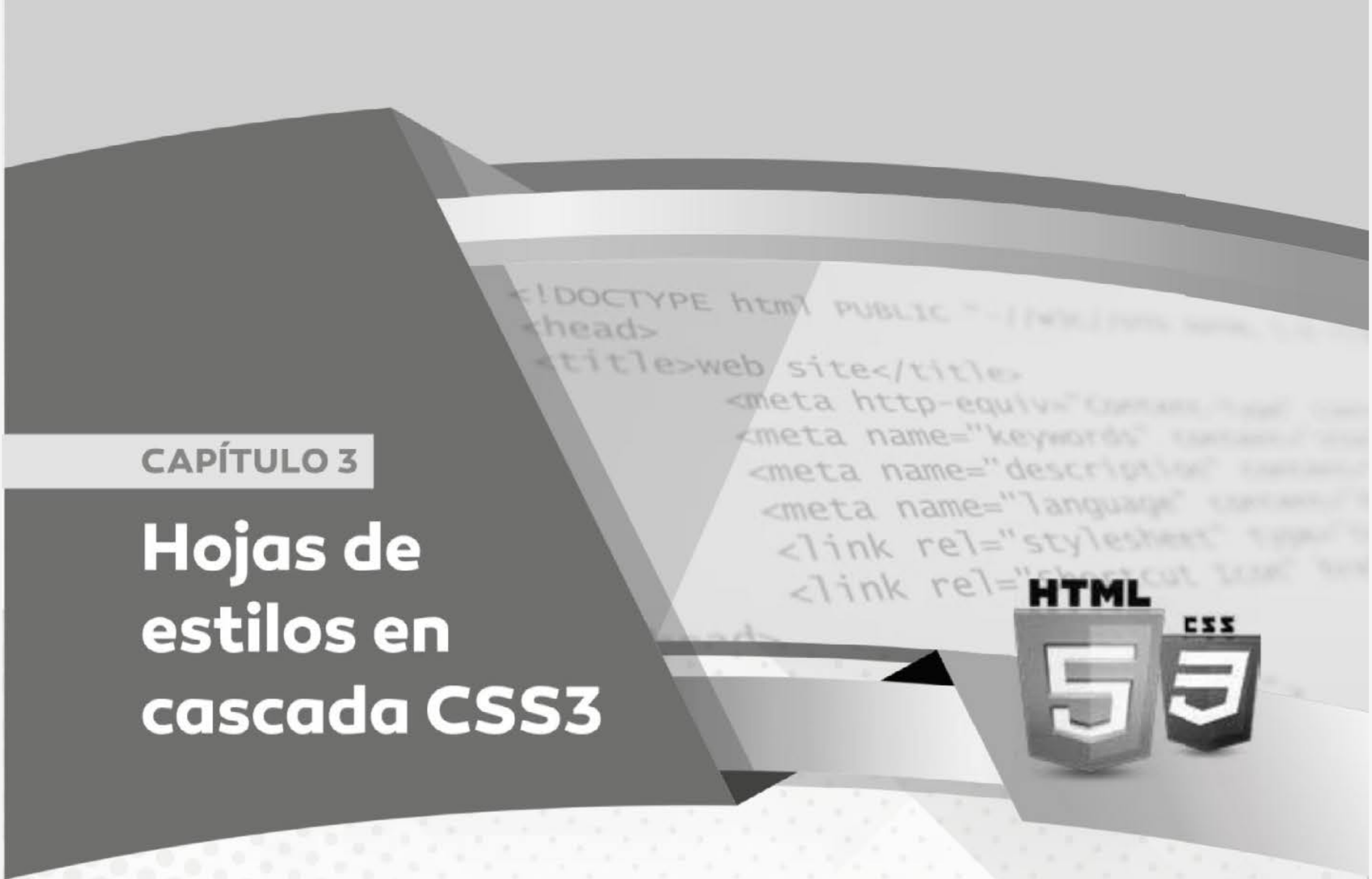
En el archivo **estilos.css** se coloca el siguiente código:

```
body{
    font-family: tahoma;
    font-size: 12px;
}
```

En el documento web se escribe el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilos.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
```

```
<body>
   <article>
      <header>
         <h2>Etiqueta ADDRESS</h2>
      </header>
      <p>
      El contenido de un elemento address provee información de contacto para el
      autor de la sección o documento, como una dirección postal, una dirección
      de correo electrónico, un número telefónico, una ciudad, un país, un código
      postal, etc.
         </p>
         <footer>
            <address>
             Tomado de: <a href="http://www.htmlquick.com/es/reference/tags/address.
             html'>HTML QUICK</a>
                <br/>
                Libro HTML5 y CSS3<br/>
                Editorial Macro-Perú<br/>
            </address>
         </footer>
      </article>
   </body>
</html>
```



CAPÍTULO 3

Hojas de estilos en cascada CSS3

Competencias

- Implementa adecuadamente estilos CSS3.
- Define correctamente los estilos de un documento web de manera profesional usando CSS3.

Contenido

3.1 Breve historia del CSS
3.2 Características del CSS3
3.3 Elementos de la definición CSS
3.4 Funcionamiento básico de CSS3
3.5 Integración CSS en HTML
3.6 Los comentarios en CSS
3.7 Selectores CSS
3.8 Selectores avanzados
3.9 Agrupación de reglas CSS
3.10 Modelos de cajas
3.11 Márgenes y rellenos
3.12 Colores en CSS
3.13 Bordes en CSS
3.14 Márgenes en CSS
3.15 Espaciado en CSS
3.16 Textos en CSS
3.17 Alineación en CSS
3.18 Enlaces en CSS
3.19 Posicionamiento CSS
- Caso desarrollado 1: Estilo de línea
- Caso desarrollado 2: Estilo a nivel de página
- Caso desarrollado 3: Estilo externo
- Caso desarrollado 4: Estilo importado
- Caso desarrollado 5: Posicionamiento normal
- Caso desarrollado 6: Posicionamiento relativo
- Caso desarrollado 7: Posicionamiento absoluto
- Caso desarrollado 8: Posicionamiento flotante

3.1 Breve historia del CSS

Es un lenguaje de hojas de estilos que sirve para controlar el aspecto o presentación de un documento web. Se podría decir que separa el contenido web de su presentación y se usa para documentos web complejos.



Figura 3.1 Composición de un documento web.
Fuente: Elaboración propia.

La World Wide Web Consortium, conocida como W3C, propuso la creación de un lenguaje de hojas de estilos que permitiera definir los formatos de diseño para el lenguaje HTML. Ante ello, se presentaron nueve propuestas de las cuales solo quedaron dos: el CHSS (*Cascading HTML Style Sheets*) y la SSP (*Stream-based Style Sheet Proposal*). Finalmente, se unieron para definir el nuevo lenguaje de estilos, llamado *Cascading Style Sheets* también llamado CSS.

Una de las principales características que presenta las hojas de estilo es que dentro de un proyecto web se puede crear solo un archivo de estilo en la que se definan todos los estilos que usará el proyecto y que se podrán emplear por medio de una asociación con el documento web. Esto genera menos líneas de código y un orden dentro del proyecto:

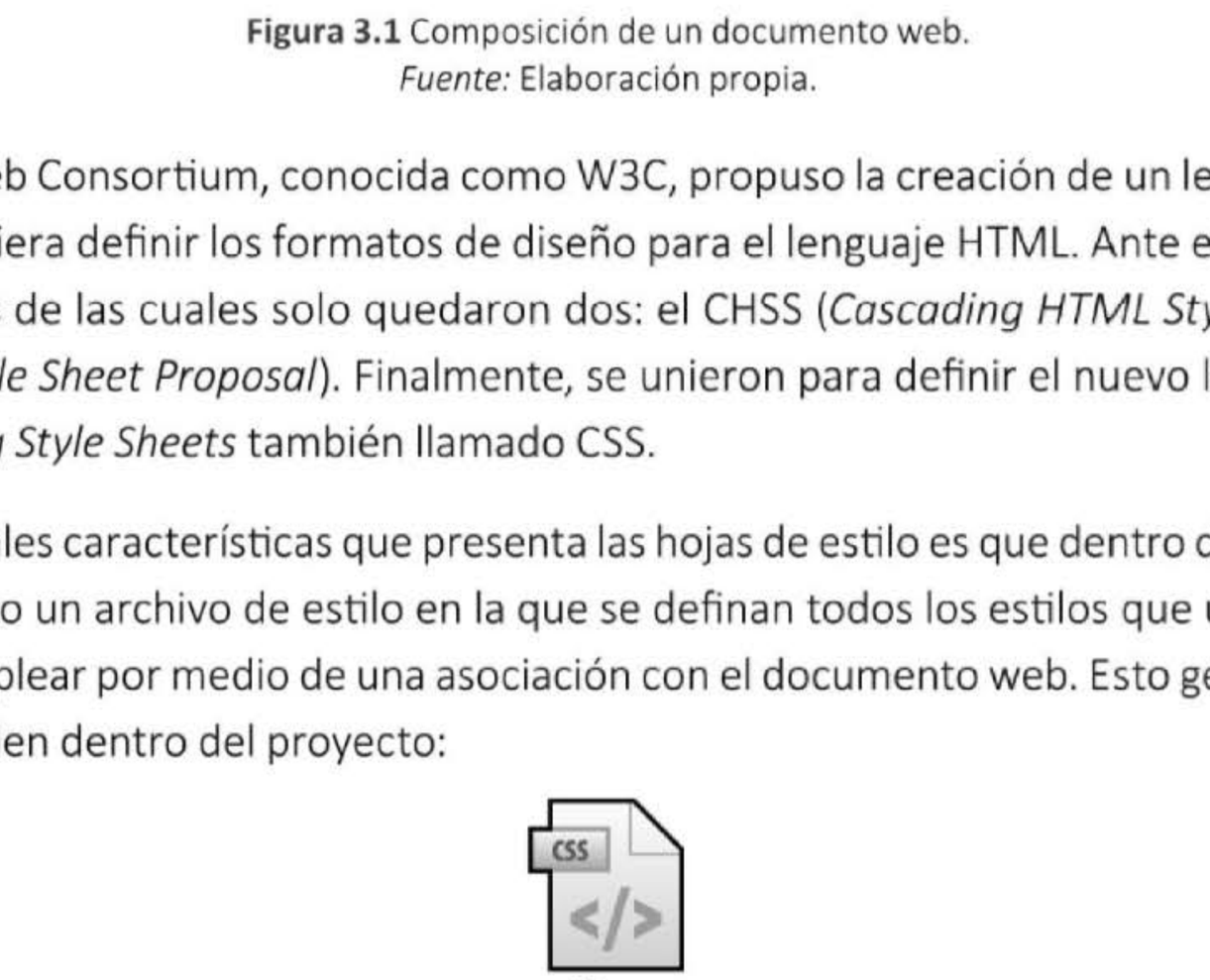


Figura 3.2 Distribución de archivos en un proyecto web.
Fuente: Elaboración propia.

Con respecto a las versiones anteriores a CSS3, se tiene lo siguiente:

Tabla 3.1 Versiones del CSS.

VERSIÓN	VIGENCIA	FUNCIONALIDADES
CSS1	1996-1998	» Gestión de fuentes de letras, así como su tipo, tamaño, énfasis y otras funcionalidades. » Color de texto, fondos, bordes u otros elementos. » Atributos del texto como espaciado entre palabras, letras, líneas y otras funcionalidades. » Gestión de alineación de textos, imágenes, tablas y otras funcionalidades. » Gestión de propiedades de las cajas como margen, borde, relleno o espaciado. » Gestión y propiedades de identificación y presentación de listas.
CSS2	1998-2008	» Gestión de capas para posicionamiento relativo, niveles y otras funcionalidades. » Incorporación de media *types*. » Soporte para las hojas de estilo auditivas. » Gestión de textos bidireccionales, sombras y otras funcionalidades.

Fuente: Elaboración propia.

3.2 Características del CSS3

CSS3 tiene características muy importantes que aportan al diseño y presentación de un documento web, evitando así el uso de aplicaciones de terceros para su definición. Véanse las principales características:

- Define un gradiente de color en el borde de los elementos con CSS.
- Incluye bordes redondeados a través del atributo **border-radius**, que define la curvatura que debe tener el borde de un elemento.
- Permite gestionar la distribución de múltiples imágenes de fondo.
- Incluye todos los colores RGB.
- Define el **word-wrap** que sirve para romper las palabras que son demasiado largas y que no caben enteras por la anchura de una caja.
- Controla el maquetado de textos en varias columnas.
- Utiliza imágenes como bordes de los elementos de la página.
- Crea sombras con el atributo **box-shadow**.
- Añade el resplandor exterior a los elementos con la propiedad **box-shadow**.
- Controla la posición de una imagen de fondo con respecto al borde, **padding** o el contenido del elemento.
- Controla el contenido de un documento web cuando algún elemento sobrepasa los límites de un contenedor, ya sea horizontal o vertical.
- Incluye **@font-face**, que permite utilizar cualquier tipografía en un documento web.

3.3 Elementos de la definición CSS

A continuación, se describe los elementos de una declaración CSS:

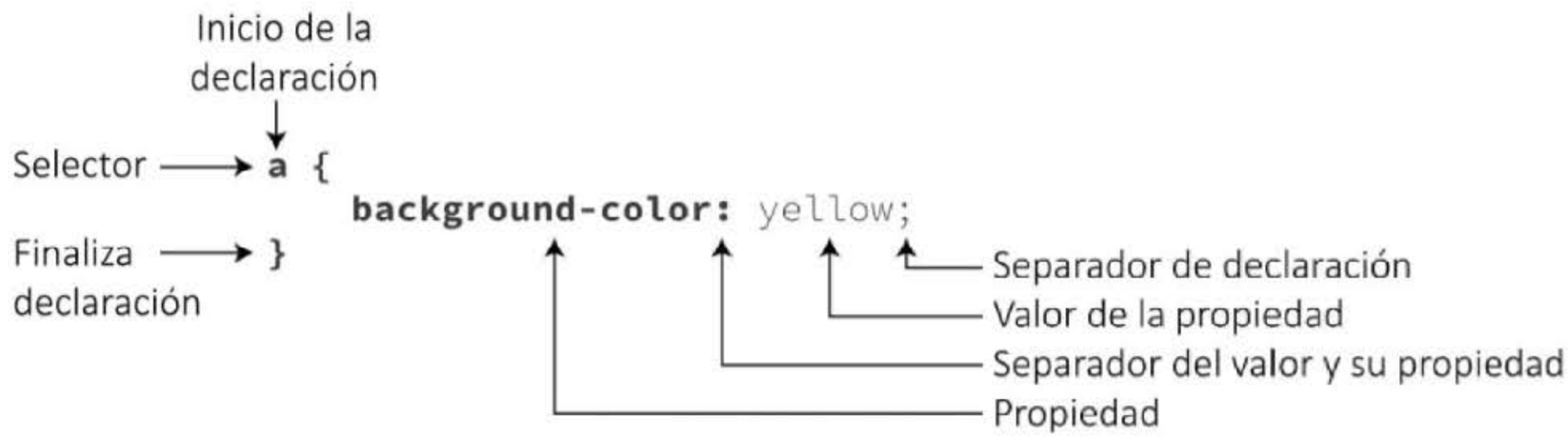


Figura 3.3 Elementos de la definición del CSS.
Fuente: Elaboración propia.

Donde:

ELEMENTO	DESCRIPCIÓN
Selector	Aquí se especifica el elemento que hay que definir. Este puede ser el nombre de una etiqueta HTML o un nombre particular del proceso. » **Etiqueta:** Hace referencia a una etiqueta HTML podría ser `<a>`, `<hr>`, `<header>`, `<section>`, etc. » **#Identificador:** Hace referencia a un nombre particular al que se le puede asignar a un grupo de propiedades. » **.clase:** Hace referencia a una clase con un nombre particular al que se le puede asignar a un grupo de propiedades.
Inicio de la declaración	La llave abierta indica el inicio de la definición del estilo.
Propiedad	Es la característica que el selector va a modificar. Podría ser un color de letra, el tipo de letras, etc.
Separador del valor y su propiedad	El separador indica el inicio del valor asignado a la propiedad.
Valor de la propiedad	Es el dato que se asigna a la propiedad.
Separador de declaración o de propiedad	El punto y coma indica la finalización de la propiedad y el inicio de otra. Debe tenerse en cuenta que cada definición de propiedad debe separarse por punto y coma.
Finaliza declaración	La llave cerrada indica la finalización de la declaración de propiedades.

3.4 Funcionamiento básico de CSS3

Un archivo de tipo CSS contiene toda la definición de diseño que puede tener las etiquetas y amplia todas las posibilidades que pueda presentar HTML5. Antes de la adopción de CSS, los diseñadores de páginas web debían definir el aspecto de cada elemento dentro de las etiquetas HTML de la página, haciendo que las líneas de código crezcan y que esto sea un poco engorroso para controlar los errores dentro del documento. Véase el siguiente código:

```
<font color='blue' face='tahoma' size='5pt'>
   Contenido textual
</font>
```

El código anterior muestra una definición donde el tipo de letra es Tahoma; el tamaño, 5pt, y el color de la letra, azul. Esta implementación no permite aplicar con facilidad el mismo formato a otros elementos similares, pues se tendría que copiar la asignación a cada párrafo que necesite la misma especificación. Ahora véase la implementación usando el código CSS:

```
p{
    color:blue;
    font-family:'tahoma';
    font-size: 5pt;
}
```

Para usarlo en un párrafo HTML, véase el siguiente código:

```
<p>Contenido textual</p>
```

La etiqueta `<p>` define un párrafo y, al encontrarse esta especificada en el archivo CSS, el formato se aplicará a todas las propiedades definidas.

3.5 Integración CSS en HTML

Existen cuatro formas de integrar CSS3 a un documento web. A continuación, su desarrollo.

3.5.1 *Estilo en línea*

Aquí se define el estilo directamente en la misma definición de la etiqueta HTML. Esta implementación presenta una desventaja al momento de actualizar el estilo, ya que un cambio en un estilo solo afectará a la etiqueta implementada. Su formato es:

```
<etiqueta style='atributo1: valor; atributo2: valor'>
    Contenido de la etiqueta
</etiqueta>
```

Ejemplo 1

Véase la siguiente implementación:

```
<body style='background-color: #ff0000'> </body>
```

El script permite asignar el color rojo como fondo en un documento web. Hay que tener en cuenta que la asignación de color se puede realizar mediante un código que inicia con el símbolo # o también podría especificarse el nombre del color, por ejemplo, **"background-color: red"**.

```
<p style='font-family:'Tahoma'; font-size:24pt; font-style:bold; text-align:center'>
    Estilo aplicado a una etiqueta html
</p>
```

El script permite asignar a un párrafo el tipo de letra Tahoma con un tamaño de 24pt, centrado y en negrita.

3.5.2 *Estilo a nivel de página*

En un documento web pueden existir diferentes definiciones de estilos, los cuales pueden ser administrados de forma global. Para ello, la definición de estilos se debe realizar dentro de la etiqueta **<head>** para que esta sea reconocida única y exclusivamente por la página web que la contenga. Véase la definición de estilos a nivel de página:

```
<style>
   etiqueta 1{
      atributos;
   }
   etiqueta 2{
      atributos;
   }
</style>
```

Ejemplo 2

Ahora bien, véase la siguiente implementación:

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv='content-type' content='text/html; charset=utf-8' />
      <title>manejo de estilos</title>
      <style>
         body {
            background-color: #ff0000;
         }

         p {
            font-family:'Tahoma';
            font-size:24px;
            font-style:inherit;
            text-align:center;
         }
      </style>
   </head>
   <body>
      <p>Estilo aplicado a una etiqueta html</p>
   </body>
</html>
```

El script aplica un color de fondo rojo a la página y para los párrafos un tipo de letra Tahoma con un tamaño de 24pt, con estilo negrita y con alineación centrada.

3.5.2 *Estilo a nivel de página*

En un documento web pueden existir diferentes definiciones de estilos, los cuales pueden ser administrados de forma global. Para ello, la definición de estilos se debe realizar dentro de la etiqueta **<head>** para que esta sea reconocida única y exclusivamente por la página web que la contenga. Véase la definición de estilos a nivel de página:

```
<style>
   etiqueta 1{
      atributos;
   }
   etiqueta 2{
      atributos;
   }
</style>
```

Ejemplo 2

Ahora bien, véase la siguiente implementación:

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv='content-type' content='text/html; charset=utf-8' />
      <title>manejo de estilos</title>
      <style>
         body {
            background-color: #ff0000;
         }

         p {
            font-family:'Tahoma';
            font-size:24px;
            font-style:inherit;
            text-align:center;
         }
      </style>
   </head>
   <body>
      <p>Estilo aplicado a una etiqueta html</p>
   </body>
</html>
```

El script aplica un color de fondo rojo a la página y para los párrafos un tipo de letra Tahoma con un tamaño de 24pt, con estilo negrita y con alineación centrada.

3.5.3 *Estilo externo*

Un archivo externo genera un orden dentro del proyecto web, ya que contendrá definiciones de estilo para todos los documentos web que lo necesitan, es decir, en este modo los estilos definidos no son propiedad de un solo documento, sino de todo el proyecto. Una gran ventaja que presenta esta forma de implementación es que la modificación en uno de los estilos afectará a todos los documentos que usan dicho estilo. El código de referencia para enlazar la página web con el archivo CSS se escribe de la siguiente manera:

```
<link href='archivo.css' rel='stylesheet' type='text/css'/>
```

Ejemplo 3

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
    background-color: #ff0000;
}

p {
    font-family:'Tahoma';
    font-size:24px;
    font-style:bold;
    text-align:center;
}
```

El script implementa dentro del archivo **estilo.css** las características para el fondo del documento y los párrafos. Ahora véase cómo hacer referencia al archivo **estilo.css** (texto resaltado) desde otro documento llamado **index.html**:

```
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv='content-type' content='text/html; charset=utf-8' />
        <link rel='stylesheet' type='text/css' href='estilo.css'>
        <title>manejo de estilos</title>
    </head>
    <body>
        <p>Estilo aplicado a una etiqueta html de forma externa</p>
    </body>
</html>
```

3.5.4 *Estilo importado*

Las diferentes formas de asociar el archivo CSS con el HTML hacen pensar que la mejor manera de integrarlos es haciendo que el primero se encuentre separado del segundo. Aplicando lo anterior, se llama estilo importado a la forma en que se invoca al archivo CSS dentro del archivo HTML. Este estilo presenta el siguiente formato:

```
<style type='text/css'>
   @import url ('nombre del archivo css');
</style>
```

Ejemplo 4

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
   background-color: #ff0000;
}

p {
   font-family:'Tahoma';
   font-size:24px;
   font-style:bold;
   text-align:center;
}
```

El script implementa dentro del archivo **estilo.css** las características para el fondo del documento y los párrafos. Ahora véase cómo se hace la referencia al archivo **estilo.css** (texto resaltado) mediante la importación.

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv='content-type' content='text/html; charset=utf-8' />
      <title>manejo de estilos</title>
      <style type='text/css'>
         @import url('estilo.css');
      </style>
   </head>
   <body>
      <p>Estilo aplicado a una etiqueta html de forma importada</p>
   </body>
</html>
```

3.6 Los comentarios en CSS

Los comentarios se caracterizan por agregar, dentro del script CSS, una línea que no será interpretada al ejecutar la página web. Un comentario se podría usar para:

- Colocar algún mensaje para recordar alguna instrucción.
- Anular algún código implementado dentro del script CSS.
- Para definir el nombre del autor y la fecha de creación del script CSS.

Véase el formato para la implementación de un comentario dentro de la definición de un archivo CSS:

```
/*
   Cuerpo
   del
   comentario
*/
```

O también podría realizarse en una sola línea:

```
/* Cuerpo del comentario de una sola linea */
```

Ejemplo 5

En el archivo **estilo.css** se escribe el siguiente código:

```
body {
   background-color: #ff0000; /*define color de fondo*/
}

p {
   font-family:'Tahoma'; /*define una fuente*/
   font-size:24px;       /*define el tamaño en 24 pixeles*/
   font-style:inherit;   /*define el texto en negrita*/
   text-align:center;    /*define alineación centrada de texto*/
}
```

En el script se observa que los comentarios al lado derecho del código explican el funcionamiento de cada línea de estilo CSS. Estos comentarios no son interpretados por el navegador web. Ahora bien, como ejemplo, en el mismo script puede anularse la definición de la fuente y su tamaño mediante el siguiente código:

```
body {
   background-color: #ff0000;
}

p {
   /* font-family:'Tahoma';
   font-size:24px; */
   font-style:inherit;
   text-align:center;
}
```

3.7 Selectores CSS

Un selector identifica un elemento dentro del documento web; contiene, además, ciertas propiedades previamente inicializadas que luego serán interpretadas por el navegador y mostradas en el diseño del documento web.

Para las hojas de estilos, un selector es considerado como un patrón que permite apuntar a elementos HTML específicos del documento web. Gracias a los diferentes tipos de selectores se puede alcanzar dichos elementos sin importar su tipo, posición o contexto.

3.7.1 *Selector universal*

Es un selector que se aplica a todos los selectores en un documento web. Esto significa que la definición de estilos se aplicará a todos los elementos compatibles. Su formato es el siguiente:

```
* {
    propiedad: valor;
}
```

En la siguiente figura, se puede ver que se define el color rojo dentro del selector universal. Esto quiere decir que todos los textos que se encuentren dentro del documento web serán mostrados con dicho color.

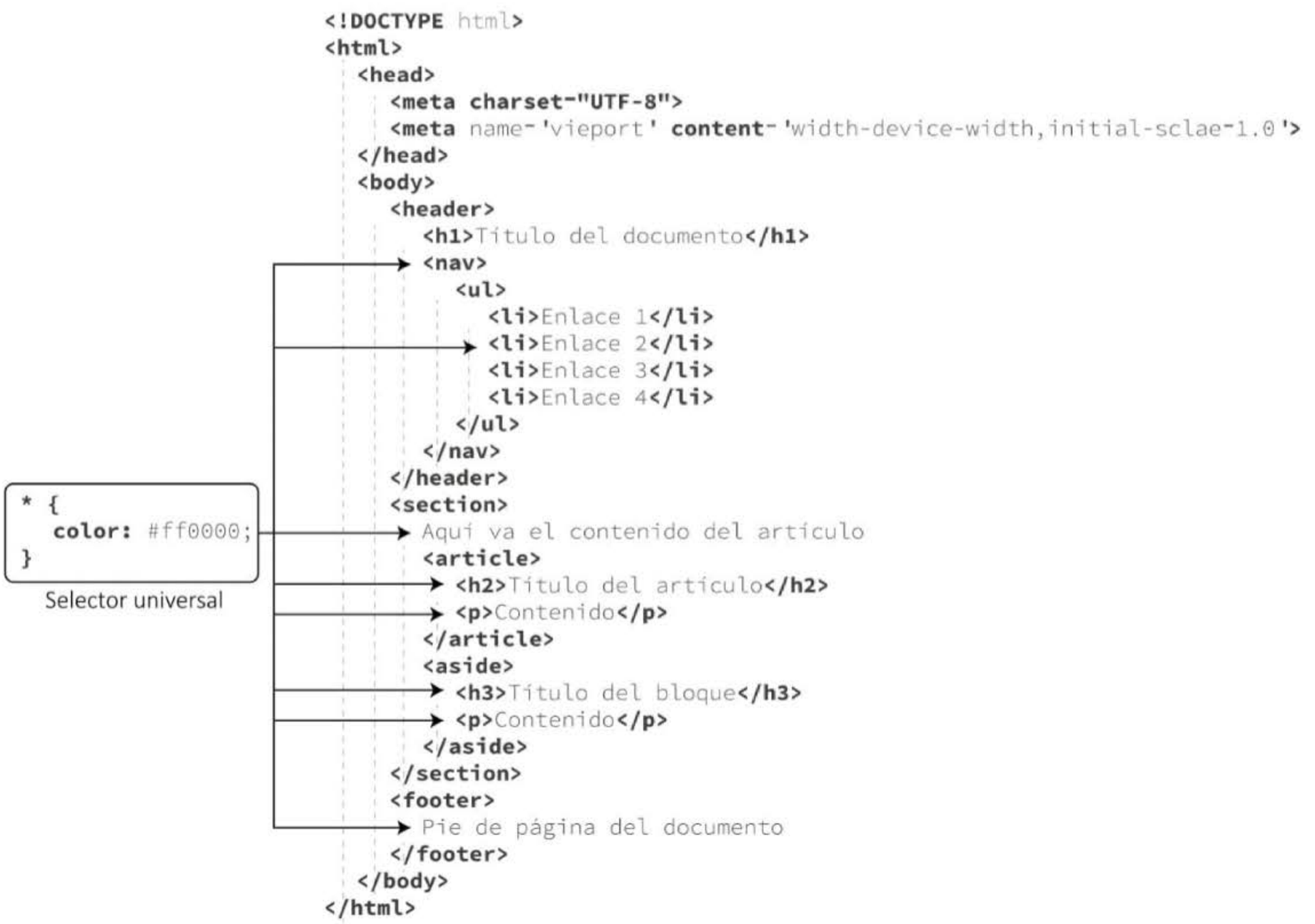


Figura 3.4 Aplicación del selector universal.
Fuente: Elaboración propia.

Ejemplo 6

Ahora bien, véase el siguiente código:

```
* {
    margin: 0;
    padding: 0;
}
```

El script define de forma general la eliminación de los márgenes y relleno de todos los elementos HTML que contiene el documento web.

```
* {
    color: blue;
    text-align: center;
    font-size: 1.5em;
}
```

Lo anterior define de forma general el color azul, el centrado y el tamaño de 1.5em a todos los textos que se encuentre definidos dentro del documento web.

```
* [lang-en] {
    color:#FF0000;
}
```

El script anterior define de forma general el color rojo a todos los elementos cuya especificación en su atributo de lenguaje sea el inglés de EE.UU.; por ello, se indica con **en**.

3.7.2 *Selector de etiqueta*

Este tipo de selector permite uniformizar los estilos. Para ello, se aplicará un estilo por etiqueta y este será ejecutado para todo el documento web. Cuando la etiqueta coincida con el selector del CSS, se aplicará el estilo definido. Su formato es:

```
selector {
    propiedad:valor;
}
```

En la siguiente figura, se puede ver que se define un color para la etiqueta **<h1>** y otro para la etiqueta **<p>**; por lo tanto, todos los demás textos que se encuentren en el documento tendrán asignado esos colores por defecto.

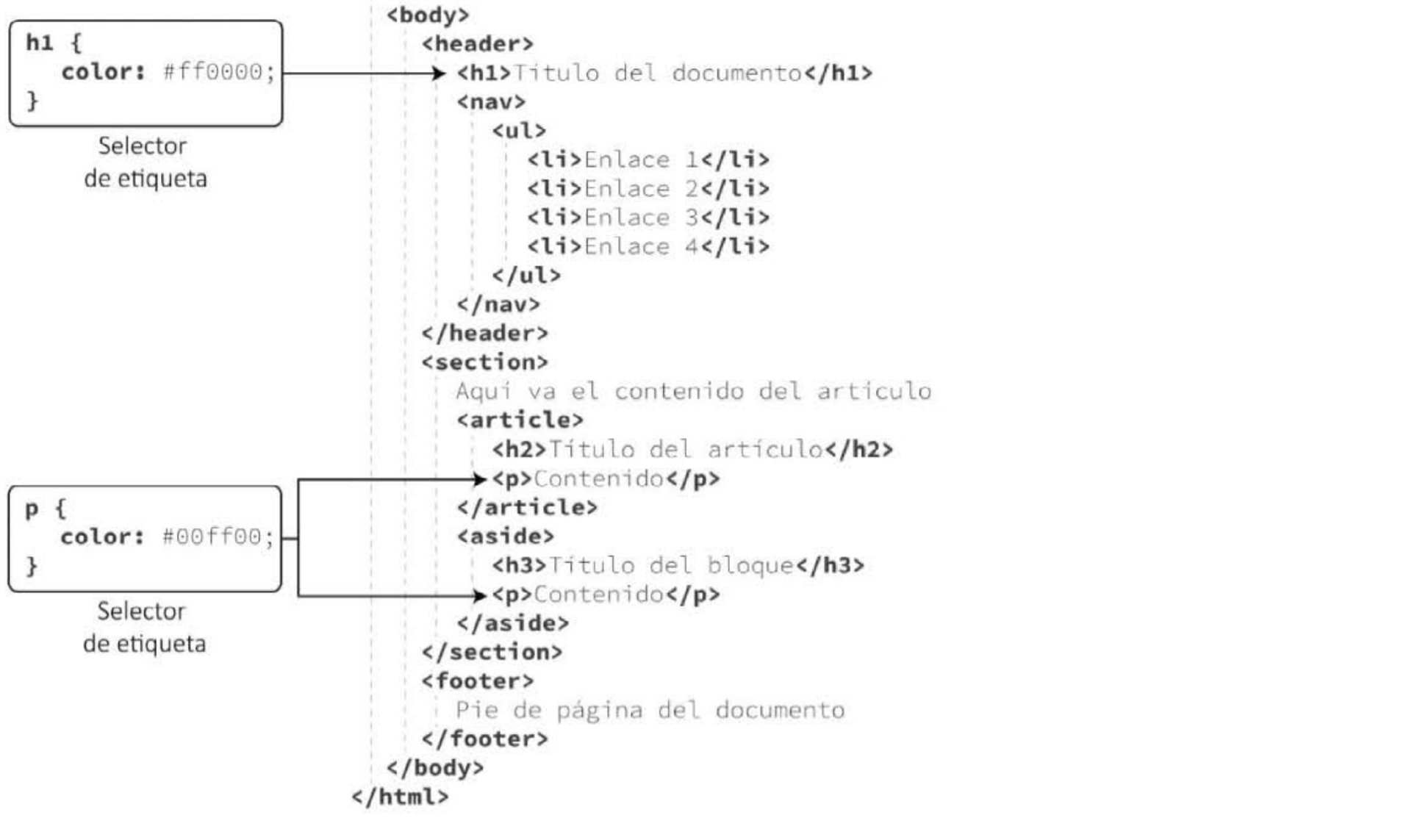


Figura 3.5 Aplicación del selector de etiqueta.
Fuente: Elaboración propia.

Ejemplo 7

Ahora bien, véase el siguiente código:

```
p{
    color: #00f;
    font-family: tahoma;
    size: 12pt;
}
```

El script define estrictamente propiedades a la etiqueta <p>. Si estas propiedades resultan ser comunes entre varios elementos, se les puede agrupar. Por ejemplo, véase lo siguiente:

```
h1 {
    color: red;
    font-weight: normal;
    font-family: Arial, Helvetica, sans-serif;
}
```

```
h2 {
   color: red;
   font-weight: normal;
   font-family: Arial, Helvetica, sans-serif;
}
h3 {
   color: red;
   font-weight: normal;
   font-family: Arial, Helvetica, sans-serif;
}
```

Como se puede observar, el script anterior define de las mismas propiedades para tres etiquetas diferentes. Este código podría resumirse de la siguiente manera:

```
h1, h2, h3 {
   color: red;
   font-weight: normal;
   font-family: Arial, Helvetica, sans-serif;
}
```

3.7.3 *Selector en descendencia*

Selecciona los elementos que se encuentran dentro de otros elementos. Un elemento es descendiente de otro cuando se encuentra entre las etiquetas de apertura y de cierre del otro elemento. El formato de selector en descendencia es el siguiente:

```
selector1 selector2 selector3 ... selectorN {
   Definición de estilos;
}
```

Ejemplo 8

Véase el siguiente script:

```
p a {
   font-size:18pt;
   color:#00f;
   font-family:tahoma;
}
```

El script define un estilo a la etiqueta `<a>` que se encuentra en descendencia con la etiqueta `<p>`, donde se aplica el color azul, el tipo de letra Tahoma y con un tamaño de 18pt. Para poder usar dicha definición se debe usar el siguiente código:

```
<body>
   <p>
      probando el selector de descendencia
   </p>
```

```
    <p>
        <a href='http://www.google.com'>Ir a google</a>
    </p>
    <a href='http://www.hotmail.com'>Ir a hotmail</a>
</body>
```

Debe tenerse en cuenta que el script CSS solo será aplicable si la etiqueta **<a>** se encuentra dentro de la etiqueta **<p>**; caso contrario, no se aplicará. Entonces, si se analiza el código, el estilo solo será aplicable al texto **Ir a google**.

3.7.4 *Selector de clase*

Permite asignar un estilo a los elementos web cuyo atributo contenga **class**. Se recomienda que el nombre que se asigne debe tener relación con las propiedades especificadas. También, para mayor facilidad de búsqueda, se puede agrupar a las clases que tengan el mismo objetivo. Su formato es el siguiente:

```
.nombre_del_identificador {
    Definición de estilos;
}
```

Ejemplo 9

Véase lo siguiente:

```
.formato {
    color: red;
    font-family: tahoma;
    font-size: 24pt;
}
```

El script anterior permite definir un tipo de letra, tamaño y color bajo el nombre de la clase **formato**. Para hacer referencia a la clase se debe anteponer un punto. Ahora bien, para implementarlo, debe anotarse el siguiente código:

```
<p class='formato'>
    Selector de clase
</p>
```

3.7.5 *Selector de identificación*

Aplica un estilo a los elementos web cuyo atributo contenga **id**. Se distingue porque inicia con el símbolo **(#)** es uno de los selectores más comunes en CSS. Este tipo de selector se caracteriza por no permitir la reutilización. Su formato es el siguiente:

```
#identificador {
    Definición del estilo;
}
```

Ejemplo 10

Véase lo siguiente:

```
#formato {
    color: red;
    font-family: tahoma;
    font-size:24px;
}
```

El script define el identificador **formato** con las siguientes características: color rojo, tipo de letra Tahoma y tamaño 24pt. Estas podrán ser aplicadas a los textos dentro de un documento web. Ahora bien, se muestra su implementación en HTML:

```
<p id='formato'>
   Selector de identificación
</p>
```

3.8 Selectores avanzados

CSS presenta diversas formas de implementar selectores. Véase algunas características nuevas que presenta:

3.8.1 *Selector hijo*

Se le llama así a la definición de estilos directamente dependiente de otro selector. Trabaja de la misma manera que los selectores descendentes, la diferencia es la forma de aplicación del estilo. Su formato es:

```
selectorpadre > selectorhijo {
   Definicion de los estilos;
}
```

Ejemplo 11

Véase el siguiente script:

```
p > a {
    font-size:18px;
    color:#00f;
    font-family:tahoma;
}
```

El script define un tamaño, tipo de letra y color a la etiqueta **<a>** solo si esta se encuentra dentro de la etiqueta **<p>**. Únicamente así se aplicará la descendencia. El siguiente código muestra el uso del selector:

```
<p>
   <a href='http://www.google.com'>ir a google</a>
</p>
```

3.8.2 *Selector adyacente*

Se le llama así a la definición de estilos en el que un selector es directamente dependiente de otro, donde, además, cuando esta se aplique también se encuentre directamente proporcional a su definición. Su formato es el siguiente:

```
selector1 + selector2 {
   Definición de estilos;
}
```

Ejemplo 12

Véase el siguiente código:

```
h1 + h2 {
   color: #0000ff;
   font-family: tahoma;
}
```

El script presenta la implementación del estilo adyacente a los selectores **h1** y **h2** con el tipo de letra Tahoma y el color azul. Para hacer la referencia a un documento web se presenta el siguiente código:

```
<h1> Uso de la etiqueta h1 </h1>
<h2> Uso de la etiqueta h2 </h2>
<h2> Uso nuevamente de la etiqueta h2 </h2>
```

En este caso, el estilo será aplicado solo a la segunda línea del código, ya que **h1** y **h2** se encuentran de forma adyacente y para que pueda aplicarse el estilo se debe usar las etiquetas **<h1>** y **<h2>** de forma seguida.

3.9 Agrupación de reglas CSS

Es un ordenamiento de código repetitivo que permite agrupar estilos similares entre los elementos web. Véase el formato de agrupación de estilos:

```
selector1, selector2 {
   Definición de estilos;
}
```

Ejemplo 13

Véase el siguiente script:

```
p {
   font-family: tahoma;
}
```

```
p {
    color: #ff0000;
}

p {
    text-align:center;
}
```

Lo anterior muestra que dentro de un documento CSS se puede referenciar varias veces a la misma etiqueta. Por ello, la mejor manera de tratar con dicho código es por medio de la agrupación:

```
p {
    font-family: tahoma;
    color: #ff0000;
    text-align:center;
}
```

Otra forma de agrupación podría realizarse a través de los distintos elementos web, como es el caso de las etiquetas **<p>** y **<h1>**, las cuales tienen propiedades comunes, por lo que es posible agrupar su definición de la siguiente manera:

```
p, h1 {
    font-family: tahoma;
    color: #ff0000;
    text-align:center;
}
```

3.10 Modelos de cajas

Inicialmente el término *cajas* está asociado a las tablas dentro de un documento web. Tómese en cuenta que cuando un documento presenta un diseño avanzado, las tablas no resultan adecuadas para su uso. Por lo tanto, se hace uso de las etiquetas de contenido como **<section>**, **<header>**, **<article>** o **<footer>** y un poco de código CSS para crear cajas en la pantalla y, de esta manera, los contenidos puedan posicionarse según como lo desee el diseñador.

Entonces, de acuerdo con lo especificado se puede decir que:

- Se pueden componer párrafos estrechos.
- Es posible colocar imágenes que se ajusten a diferentes bloques para que estas se acomoden correctamente en la pantalla.
- Los valores **width** y **height** normalmente son valores numéricos exactos o porcentajes.
- Cuando se asigne un valor cero, la unidad de medida es opcional.
- Cuando el valor es diferente de cero o no se especifica la unidad, la medida es completamente ignorada.

Ejemplo 14

Obsérvese el siguiente código CSS.

```
header{
    background-color: gray;
    border: 5px chartreuse;
    width: 770px;
    height: 200px;
    -khtml-border-radius: 10px;
    -moz-border-radius: 10px;
    -webkit-border-radius: 10px;
    border-radius: 10px;
}
```

El script presenta una definición sobre la caja de la etiqueta <header> y esta se presenta de la siguiente manera:



CSS3 divide las unidades de altura y anchura en dos grupos llamados absolutos y relativos.

3.10.1 Unidades relativas

Estas medidas definen su valor de acuerdo a otra; por lo tanto, la definición de las unidades relativas se basa en una especie de fórmula. Véase algunas características:

- Se adaptan fácilmente a los diferentes medios, es decir, su definición se acondiciona en el medio en que se visualice. Por ejemplo, un dispositivo móvil o una computadora personal.
- Su valor no está completamente definido, pues siempre se basará en la definición de otro.
- Son las unidades más usadas dentro de los documentos web.

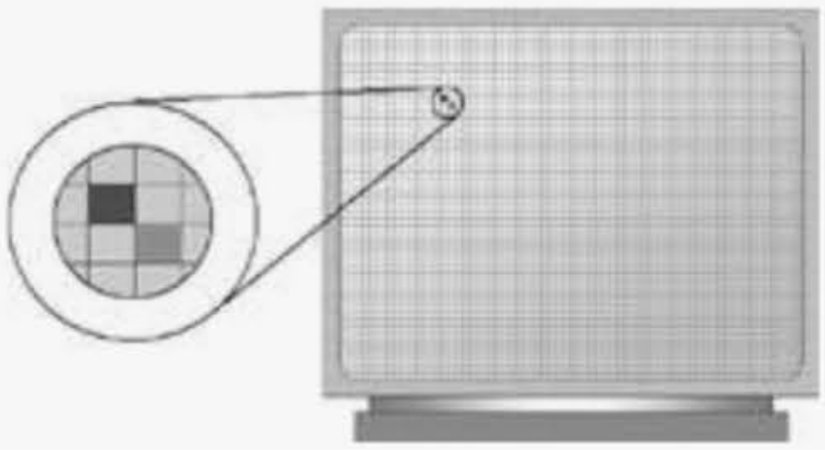
A continuación, se muestran las tres unidades de medida relativas definidas por CSS3 y la referencia que toma cada una para determinar su valor real:

Tabla 3.2 Tipos de unidades relativas de los elementos web.

TIPO	DESCRIPCIÓN

em	La unidad **em** o unidad escalable es igual al tamaño de la fuente actual. Por ejemplo, si la fuente del documento es 12pt (12 puntos), 1em es igual a 12pt; así, 2em sería igual a 24pt.
	`p {`
	`  font-size: 10px;`
	`  text-indent: 2em;`
	`}`
	La primera propiedad que se aplica al selector **p** es el tamaño de letra de 10px; la segunda línea, es la especificación de la tabulación, la cual duplica el valor definido por el tamaño de letra, para este caso, 20px.

ex	Aquí se puede especificar que el tamaño se determinará tantas veces sea la altura de la letra x minúscula. Por ejemplo, si la x en minúscula tiene un valor de 10mm como tamaño de fuente; entonces se puede decir que 1ex es 10mm, 1.5ex es 15mm y 2ex es 20mm. Por ejemplo:
	`p {`
	`  font-size: 10mm;`
	`  text-indent: 2ex;`
	`}`
	La tabulación especificada para los párrafos es 20mm.

px	Cuando se trabaja con píxeles, la especificación de valores en alto y ancho tienen más precisión, especialmente para la salida en pantalla de la computadora. Por ejemplo:
	`body{`
	`  font-size: 16px;`
	`}`
	Un pixel es la menor unidad homogénea en color que forma parte de una imagen digital, es a partir de ahí que forma elementos visibles para el usuario.
	 

Fuente: Elaboración propia.

3.10.2 *Unidades absolutas*

Las unidades absolutas definen las medidas de forma completa, ya que sus valores reales no se calculan ni tienen referencia a otras medidas, sino que son asignados directamente como valor.

A continuación, se muestran las unidades de medida absolutas definidas por CSS:

Tabla 3.3 Lista de unidades absolutas para los elementos web.

TIPO	DESCRIPCIÓN
in	Es una especificación de medida en pulgadas. Por ejemplo: `body { margin: 2.5in; }` Aquí se especifica el margen de arriba, abajo, izquierda y derecha del documento web.
cm	Se da en centímetros. Por ejemplo: `p { line-height: 1.5; font-size: 1em; }` La propiedad **line-height** define el espacio que puede tener una línea de otra en un párrafo.
mm	Esta especificación de medida se da en milímetros. Por ejemplo: `p { wordspacing: 4mm; }` La propiedad **word-spacing** permite definir el espacio entre palabras.
pt	Es una especificación de medidas en puntos. Este tipo es muy usado para textos. La unidad es equivalente a 0.35mm. Por ejemplo: `a { fontsize: 12pt }` Se define el tamaño del texto a 12 puntos para las etiquetas **<a>**.
pc	Especificación de medidas en picas, que es equivalente a 12puntos.

%	La medida porcentaje permite definir un tanto por ciento del valor definido en el selector padre. Por ejemplo:
	`p { font-size:16px; }`
	`a { font-size: 80%; }`
	Se ha definido el tamaño de párrafo a 16px y para el tamaño de texto que va como enlace un 80% del valor asignado al párrafo, es decir, 12.8px.

Fuente: Elaboración propia.

3.11 Márgenes y rellenos

Cuando se habla de márgenes y rellenos, se hace referencia a los espacios que se generan tanto dentro como fuera de una etiqueta web. Véase en la siguiente figura una imagen general de los espacios, bordes y márgenes de una etiqueta:

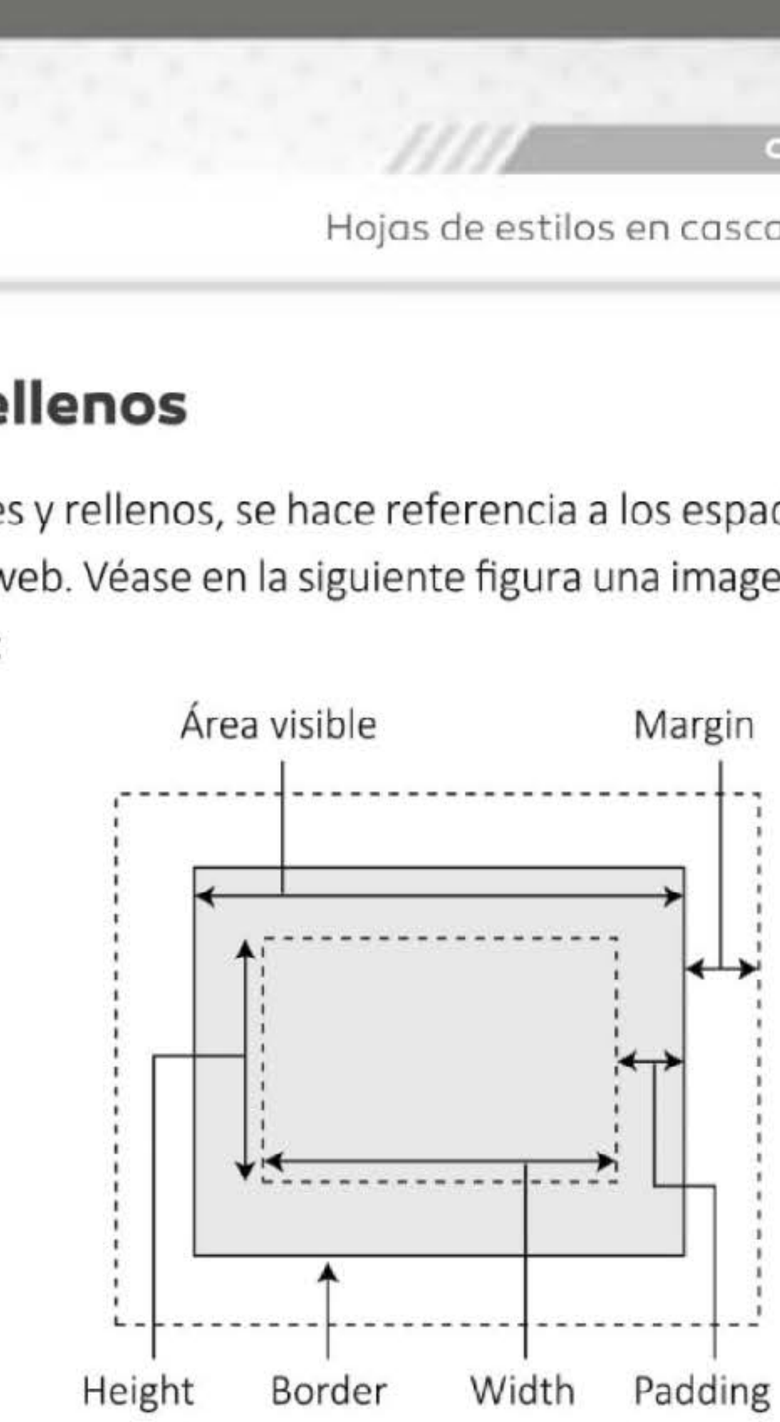


Figura 3.6 Márgenes y rellenos de un elemento web.
Fuente: Elaboración propia.

Entiéndase que los elementos usados en un documento web ocupan un lugar, los cuales pueden considerarse como elementos de bloque o elementos de línea. A continuación, su desarrollo.

3.11.1 *Elementos de bloque*

Se les llama así porque siempre ocupan una nueva línea y ocupan todo el espacio en forma horizontal del documento web. Las etiquetas HTML que son consideradas como elementos por bloque son:

address	center	div	h1, h2, h3, h4, h5, h6
hr	form	menu	p
pre	table	ul	dd, dt, td, th, tr

Ejemplo 15

Véase lo siguiente:

```
h1 {
   font-family: Verdana;
   color: #FFFFFF;
   text-decoration: none;
   border: none;
   font-size: 9px;
}
```

El script anterior define un estilo a la etiqueta <h1>. El estilo determina que se ocupará una línea completa dentro del documento web. Si junto a la definición de la etiqueta <h1> se encuentran otros elementos, estos bajarán de línea y se continuará con el diseño.

3.11.2 *Elementos de línea*

Se les llama así porque no comienzan en una nueva línea y solo ocupan el espacio necesario para mostrar su contenido. Las etiquetas HTML que son consideradas como elementos por línea son:

a	b	br	Cite
em	font	i	Small
strong	u	Input, select, textarea	

Ejemplo 16

Véase el siguiente código:

```
img {
    text-decoration: none;
    border: none;
}
```

El script elimina la decoración y los bordes a las imágenes. Para poder usarlo dentro de un HTML, se emplea el siguiente código:

```
<a href='http://www.facebook.com'><img src='images/face.jpg'></a>
```

3.12 Colores en CSS

CSS3 tiene soporte de colores RGB, HSL, HSV para la mayoría de los navegadores web. En ocasiones se puede colocar el nombre del color o el código de color. Véase una lista de los principales colores y sus valores:

› Colores rojos

NOMBRE DEL COLOR	RGB	HSL	HSV
LightCoral	F08080	0° 79% 72%	0° 47% 94%
Salmon	FA8072	6° 93% 71%	6° 55% 98%
IndianRed	CD5C5C	0° 53% 58%	0° 55% 80%
Red	FF0000	0° 100% 50%	0° 100% 100%
Crimson	DC143C	348° 83% 47%	348° 91% 86%
FireBrick	B22222	0° 68% 42%	0° 81% 71%
Brown	A52A2A	0° 75% 65%	0° 58% 91%
DarkRed	8B0000	0° 100% 27%	0° 100% 54%
Maroon	800000	0° 100% 25%	0° 100% 50%

› Colores amarillos

NOMBRE DEL COLOR	RGB	HSL	HSV
LightYellow	FFFFE0	60° 100% 94%	60° 12% 100%
Cornsilk	FFF8DC	48° 100% 93%	48° 14% 100%

NOMBRE DEL COLOR	RGB	HSL	HSV
LemonChiffon	FFFACD	54° 100% 90%	54° 20% 100%
LightGoldenrodYellow	FAFAD2	60° 80% 90%	60° 16% 98%
PaleGoldenrod	EEE8AA	55° 67% 80%	55° 29% 93%
Khaki	F0E68C	54° 77% 75%	54° 41% 94%
Yellow	FFFF00	60° 100% 50%	60° 100% 100%
Gold	FFD700	51° 100% 50%	51° 100% 100%
DarkKhaki	BDB76B	56° 38% 58%	56° 43% 74%

› Colores azules

NOMBRE DEL COLOR	RGB	HSL	HSV
Lavender	E6E6FA	240° 67% 94%	240° 8% 98%
BlueWeb	CEE7FF	209° 100% 90%	209° 20% 100%
PowderBlue	B0E0E6	187° 52% 80%	187° 23% 90%
LightBlue	ADD8E6	195° 53% 79%	195° 25% 90%
LightSkyBlue	87CEFA	203° 92% 75%	203° 47% 98%
SkyBlue	87CEEB	197° 71% 73%	197° 42% 92%
LightSteelBlue	B0C4DE	214° 41% 78%	214° 21% 87%
DeepSkyBlue	00BFFF	195° 100% 50%	195° 100% 100%
CornflowerBlue	6495ED	219° 79% 66%	219° 58% 93%
DodgerBlue	1E90FF	210° 88% 100%	210° 0% 100%
SteelBlue	4682B4	207° 61% 71%	207° 40% 89%
RoyalBlue	4169E1	225° 73% 57%	225° 71% 88%
Blue	0000FF	240° 100% 100%	240° 0% 100%
MediumBlue	0000CD	240° 100% 80%	240° 40% 100%
DarkBlue	00008B	240° 100% 55%	240° 90% 100%
Navy	000080	240° 100% 50%	240° 100% 100%
MidnightBlue	191970	240° 78% 44%	240° 88% 78%

› Colores grises

NOMBRE DEL COLOR	RGB	HSL	HSV
Gainsboro	DCDCDC	—° 0% 86%	—° 0% 86%
LightGrey	D3D3D3	—° 0% 83%	—° 0% 83%
Silver	C0C0C0	—° 0% 75%	—° 0% 75%
DarkGray	A9A9A9	—° 0% 66%	—° 0% 66%
LightSlateGray	778899	210° 22% 60%	210° 26% 69%
SlateGray	708090	210° 22% 56%	210° 29% 66%
Gray	808080	—° 0% 50%	—° 0% 50%
DimGray	696969	—° 0% 41%	—° 0% 41%
DarkSlateGray	2F4F4F	180° 25% 25%	180° 40% 31%
Black	000000	—° 0% 0%	—° 0% 0%

3.12.1 *Color de fondo*

Se usa especificando la propiedad **background-color** dentro de los selectores que hagan referencia a cajas.

Ejemplo 17

Véanse los siguientes scripts:

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

section{
   background-color:#FF0000;
}

section{
   background-color:rgb(255,0,0);
}

section{
   background-color:hsl(0,100%,100%);
}
```

Los códigos anteriores muestran las diferentes formas en las que se puede definir el color de fondo para el elemento **<section>** dentro del documento web.

3.12.2 *Color al texto*

Se emplea especificando la propiedad **color** dentro de los selectores que hagan referencia a textos.

Ejemplo 18

Véanse los siguientes códigos:

```
p{
   color:blue;
}
p{
   color:#0000FF;
}
p{
   color:rgb(0,0,255);
}
p{
   color:hsl(240,0%,100%);
}
```

Los scripts anteriores muestran las diferentes formas en las que se puede definir el color azul para el elemento **<p>** dentro del documento web.

3.12.3 *Color al borde*

Se usa especificando la propiedad **border-color** dentro de los selectores que hagan referencia a elementos web que tengan un marco como las secciones.

Ejemplo 19

Véanse los siguientes scripts:

```
section {
   border-color:black;
}

section {
   border-color:#000000;
}

section {
   border-color:rgb(0,0,0);
}

section {
   border-color:hsl(0,0%,0%);
}
```

Los códigos anteriores muestran las diferentes formas en las que se puede definir el color negro para el elemento **<section>** dentro del documento web.

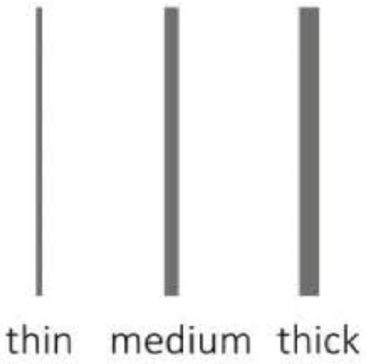
3.13 Bordes en CSS

Los bordes tienen una funcionalidad decorativa, también sirven para subrayar la separación entre dos elementos. Véanse las propiedades que presentan los bordes al ser aplicados en las etiquetas web:

3.13.1 *Ancho de borde*

El ancho de borde se define por medio de la propiedad **border-width**, la cual tiene los siguientes valores:

- **Preestablecidos**: thin, medium y thick.



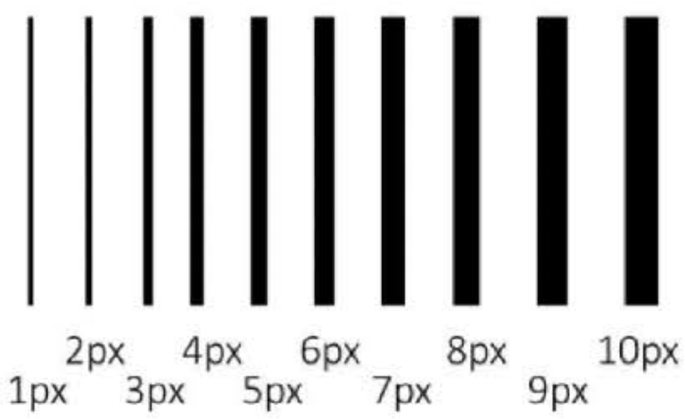
Ejemplo 20

Veáse lo siguiente:

```
border-width:thick;
border-style:solid;
```

Cuando se aplica un ancho de borde, también debe aplicarse el estilo del borde. Por lo tanto, ambos criterios de código deben ser considerados al momento de especificar el ancho.

› **Definidos por píxeles:** El cual aplica directamente un valor en píxeles.



Ejemplo 21

Veáse lo siguiente:

```
border-width: 10px;
border-style: solid;
```

Cuando se define el ancho de borde en píxeles, se puede especificar el grosor que se crea conveniente en el script.

3.13.2 Color del borde

Es la propiedad **border-color** que permite la definición de los colores sobre un borde. Los valores pueden tomar una notación hexadecimal o el nombre del color en inglés.

Ejemplo 22

Véase la implementación de borde para la etiqueta **<h1>** con un grosor de 10px y color de borde rojo:

```
h1 {
    border-width:10px;
    border-color:#f00;
    border-style:solid;
}
```

3.13.3 *Estilos de borde*

CSS3 presenta 8 tipos de estilos para los bordes por medio de la propiedad **border-style**. Así se tiene:

- **Dotted**: Borde en forma de líneas consecutivas.
- **Dashed**: Borde en forma de puntos consecutivos.
- **Solid**: Borde de línea firme.
- **Double**: Borde de doble línea firme.
- **Groove**: Borde biselado.
- **Ridge**: Borde biselado.
- **Inset**: Borde biselado interno.
- **Outset:** Borde biselado externo**.**

Ejemplo 23

Véanse los siguientes scripts:

- El siguiente script define el borde de la etiqueta **<section>** con un estilo de líneas consecutivas.

```
section {
   width: 550px;
   height: 150px;
   border-top-style: dotted;
   border-right-style: dotted;
   border-bottom-style: dotted;
   border-left-style: dotted;
}
```

También se puede definir de la siguiente manera:

```
section {
   width: 550px;
   height: 150px;
   border-style: dotted;
}
```

El resultado será:

- El código siguiente muestra la definición de un borde de color negro con un grosor de 5px y con un estilo de línea sólido.

```
section {
   width: 550px;
   height: 150px;
   border: 5px solid black;
}
```

El resultado será:



- El siguiente script muestra la definición de una sola línea del lado izquierdo, con un grosor de 6px, con estilo de línea sólida y con color azul.

```
section {
   width: 550px;
   height: 150px;
   border-left: 6px solid blue;
   background-color: beige;
}
```

El resultado será:



- El código siguiente muestra cómo redondear las esquinas del borde con un radio de 15px, con un grosor de 2px, y con un estilo de línea sólida de color negro.

```
section {
   width: 550px;
   height: 150px;
   border: 2px solid black;
   border-radius: 15px;
}
```

El resultado será:



3.14 Márgenes en CSS

Define un determinado espacio entre el elemento y su exterior. Debe tenerse en cuenta que el valor del margen comienza en el borde del elemento.

Ejemplo 24

Véase el siguiente código:

```
section{
    width: 385px;
    height: 430px;
    border: 2px solid black;
    border-radius: 10px;
}

img {
    width: 170px;
    height:194px;
    margin-top: 10px;
    margin-bottom: 10px;
    margin-right: 10px;
    margin-left: 10px;
}
```

El script anterior muestra la definición del margen de la etiqueta <img> para todos sus lados.



También se puede definir de la siguiente manera:

```
section{
    width: 385px;
    height: 430px;
    border: 2px solid black;
    border-radius: 10px;
}

img {
    width: 170px;
    height:194px;
    margin: 10px 10px 10px 10px;
}
```

Luego, si el margen es el mismo por todos los lados, entonces se puede definir de esta forma:

```
section{
    width: 385px;
    height: 430px;
    border: 2px solid black;
    border-radius: 10px;
}

img {
    width: 170px;
    height:194px;
    margin: 10px;
}
```

Ahora bien, si se necesita definir 20px al margen superior e inferior y 40px al margen izquierda y derecha, se escribe lo siguiente:

```
section{
    width: 385px;
    height: 430px;
    border: 2px solid black;
    border-radius: 10px;
}

img {
    width: 170px;
    height:194px;
    margin: 20px 40px;
}
```

3.15 Espaciado en CSS

Define un determinado espacio entre el elemento y su interior. Debe tenerse en cuenta que el valor del espaciado comienza en el borde del elemento.

Ejemplo 25

Véase el siguiente script:

```
section{
    width: 345px;
    height: 390px;
    border: 2px solid black;
    border-radius: 10px;
    padding-top: 20px;
    padding-bottom: 20px;
    padding-left: 20px;
    padding-right: 20px;
}

img {
    width: 170px;
    height:194px;
}
```

Lo anterior muestra la definición del margen aplicado a la etiqueta <section> para todos sus lados.



El script también puede definirse de la siguiente manera:

```
section{
    width: 345px;
    height: 390px;
    border: 2px solid black;
    border-radius: 10px;
    padding: 20px 20px 20px 20px;
}
```

```
img {
    width: 170px;
    height:194px;
}
```

Luego, si el margen es el mismo para todos los lados, entonces podría definirse de esta forma:

```
section{
    width: 345px;
    height: 390px;
    border: 2px solid black;
    border-radius: 10px;
    padding: 20px;
}

img {
    width: 170px;
    height:194px;
}
```

3.16 Textos en CSS

Define el formato que pueden tener los textos dentro de un documento web.

Ejemplo 26

Véanse los siguientes códigos:

- A continuación, se muestran los scripts de línea arriba, línea media o tachado, y subrayado, respectivamente:

```
#lineaUp {
    text-decoration: overline;
}

#lineaMid {
    text-decoration: line-through;
}

#lineaDown{
    text-decoration: underline;
}
```

Lo anterior sirve para definir la decoración de un texto y se muestra de la siguiente manera:

<u>Línea Arriba</u>

~~Línea Medio o Tachado~~

<u>Línea Abajo o Subrayado</u>

- También se pueden convertir los textos en mayúsculas, minúsculas o, en su caso, solo aplicar a la primera letra la mayúscula. Seguidamente, se muestran los códigos respectivos según el orden mencionado:

```
#mayuscula{
    text-transform: uppercase;
}

#minuscula {
    text-transform: lowercase;
}

#primeraLetra {
    text-transform: capitalize;
}
```

Los scripts anteriores se muestran en la página de la siguiente manera:

LETRA MAYÚSCULA
letra minúscula
Primera Letra

- El siguiente código muestra cómo aplicar espacios entre caracteres.

```
#espaciado {
    letter-spacing: 11px;
    font-size: 12px;
}
```

En el código HTML se escribe lo siguiente:

```
<section>
    <p>LIBRO HTML5 y CSS3</p>
    <p id='espaciado'>DISEÑO WEB</p>
</section>
```

Al asociar el CSS con el código HTML el resultado se muestra de la siguiente manera:

LIBRO HTML5 Y CSS3
D I S E Ñ O W E B

- Así también, se puede controlar el espacio entre líneas de párrafo. Véase el siguiente código:

```
#espaciado {
    letter-spacing: 9px;
    font-size: 12px;
}

section>p{
    line-height: 0.3;
}
```

En el código HTML se escribe lo siguiente:

```
<section>
    <p>LIBRO HTML5 y CSS3</p>
    <p id='espaciado'>DISEÑO WEB</p>
</section>
```

Al enlazar el CSS con el código HTML se obtiene lo siguiente:

LIBRO HTML5 Y CSS3

DISEÑO WEB

3.17 Alineación en CSS

Permite alinear elementos web dentro de un contenedor como la etiqueta <body>, <section> o <p>.

Ejemplo 27

› El script para alinear texto es el siguiente:

```
p{
    text-align: center;
    border: 3px solid gray;
    padding: 15px;
}
```

Lo anterior muestra la definición del centrado de un texto dentro de un contenedor. Para este caso, se emplea la etiqueta <p>, de modo que se muestre de la siguiente manera:

Libro HTML5 y CSS3

› También se puede centrar una imagen con respecto a un elemento de tipo <section> con el siguiente código:

```
section{
    width: 345px;
    height: 220px;
    padding: 5px;
    border: 2px solid black;
    border-radius: 10px;
}

img{
    display: block;
    margin-left: auto;
    margin-right: auto;
    width: 40%;
}
```

En el código HTML se escribe lo siguiente:

```
<section>
    <img src='images/comida.jpg'/>
</section>
```

Cuando se asocia el código CSS con el documento web debe mostrarse lo siguiente:



- Finalmente, véase el código para centrar una imagen respecto al tamaño total de un documento web:

```
table{
    margin-right:auto;
    margin-left:auto;
    border: 1px solid blue;
}

img{
    width: 450px;
    height: 310px;
}
```

En el código HTML se escribe lo siguiente:

```
<table>
    <tr>
        <td>
            <img src='images/foto.jpg'>
        </td>
    </tr>
</table>
```

Al enlazar el CSS con el código HTML se obtiene lo siguiente:



3.18 Enlaces en CSS

Define las características sobre el texto de enlace a otros documentos web.

Ejemplo 28

Véase lo siguiente:

```
a{
    font-family: tahoma;
    font-size: 14px;
    color: blue;
}
```

El script anterior define el color azul para el texto de enlace. Ahora bien, se pueden definir los colores de acuerdo al estado del enlace, por ejemplo:

```
body{
    font-family: tahoma;
    font-size: 14px;
}

/* Color inicial */
a:link {
    color: red;
}

/* Color del enlace visitado */
a:visited {
    color: green;
}
```

```
/* Color con el mouse encima del enlace */
a:hover {
    color: black;
}

/* Color del enlace seleccionado */
a:active {
    color: blue;
}
```

En el script se define el color rojo cuando el enlace se muestra en la web sin seleccionarse. Luego se aplica el color verde solo si ya se visitó dicho enlace. Seguidamente, se mostrará de color negro al posicionar el puntero del mouse por encima del enlace, y si el enlace se selecciona, se mostrará de color azul.

```
body{
    font-family: tahoma;
    font-size: 14px;
}
a:link {
    text-decoration: none;
}
a:visited {
    text-decoration: none;
}
a:hover {
    text-decoration: underline;
}
a:active {
    text-decoration: underline;
}
```

Lo anterior define un formato sin subrayado para los enlaces iniciales y visitados; asimismo, cuando se posiciona el puntero del mouse o se selecciona el enlace, se mostrará el subrayado.

```
body{
    font-family: tahoma;
    font-size: 14px;
}

a:link {
background-color: yellow;
}

a:visited {
    background-color: cyan;
}

a:hover {
    background-color: green;
}
```

```
a:active {
    background-color: black;
}
```

El script anterior define un color de relleno amarillo si el enlace se encuentra en el estado inicial; si se visita, se rellena de color cyan. En el caso de que el puntero del mouse se pase por el enlace, este se muestra de color verde, y si se selecciona, cambia a color negro.

```
body{
    font-family: tahoma;
    font-size: 14px;
}

a:link, a:visited {
    background-color: #FF0000;
    color: white;
    padding: 14px 25px;
    text-align: center;
    text-decoration: none;
    display: inline-block;
}

a:hover, a:active {
    background-color: green;
}
```

En este último script se define un color de fondo rojo para el documento web; cuando se posiciona el puntero del mouse en el fondo, este cambia a color verde.

3.19 Posicionamiento CSS

Cuando los elementos web, como las etiquetas HTML o el mismo código CSS, son agregados dentro de un documento, estos se procesan de una manera especial para que, finalmente, puedan ser entendidos por el navegador. Ahora bien, como cada selector genera una caja se debe tener en cuenta lo siguiente:

- Definición del alto y ancho de la caja, esté o no establecida.
- Identificación del tipo de etiqueta HTML, ya sea como elemento de bloque o de línea.
- Definición del posicionamiento de la caja.
- Establecimiento del tamaño del navegador.
- Establecimiento del tamaño de las imágenes.

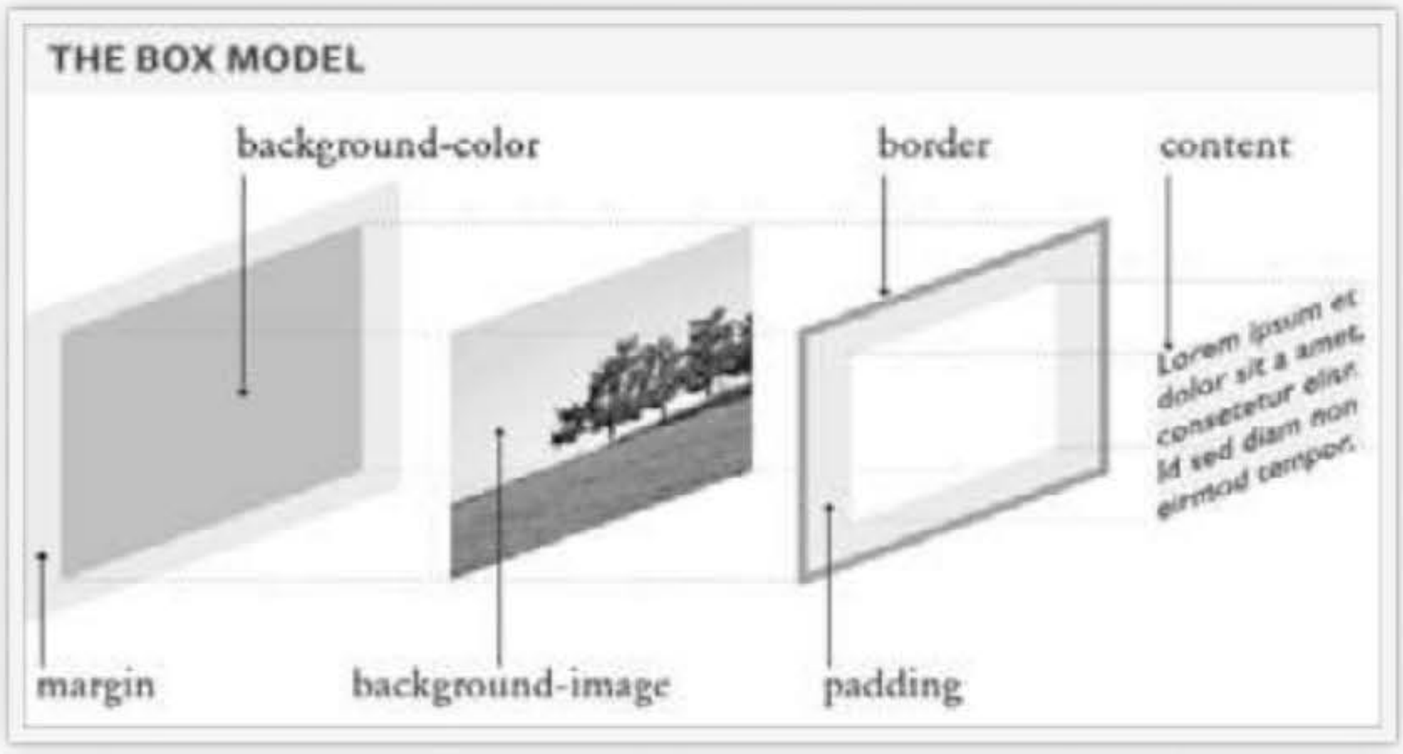


Figura 3.7 Posición de las etiquetas CSS.
Fuente: <http://respag.net/modelo-de-cajas-de-css.aspx>.

Hay que entender que un documento web tiene una dimensión inicial preparada por el diseñador web. Véase una distribución de píxeles de un documento:



Figura 3.8 Distribución de píxeles en un documento web.
Fuente: Elaboración propia.

En la figura anterior, se puede observar que la dimensión del documento es de 500px de ancho por 250px de alto. Este tamaño es variable y lo determina el diseñador web.

En vista de que los elementos HTML son colocados en una posición estándar dentro del documento web, es que se pueden modificar dichas posiciones según convenga, para lo cual se deben considerar las siguientes propiedades:

- Posicionamiento normal, posicionamiento relativo.
- Posicionamiento absoluto, posicionamiento fijo y posicionamiento flotante

3.19.1 *Posicionamiento normal*

Son llamados también estáticos. Es el más usado en todos los navegadores. El nombre estático no resulta el más adecuado pues, en realidad, los cuadros con **position:static** no se "posicionan" en el sentido que indica

el CSS, sino más bien en el orden en el que aparecen en el etiquetado. Además, ocupan todo el espacio que necesitan. Este también es el comportamiento por defecto que se da cuando no se aplica ningún CSS al HTML.

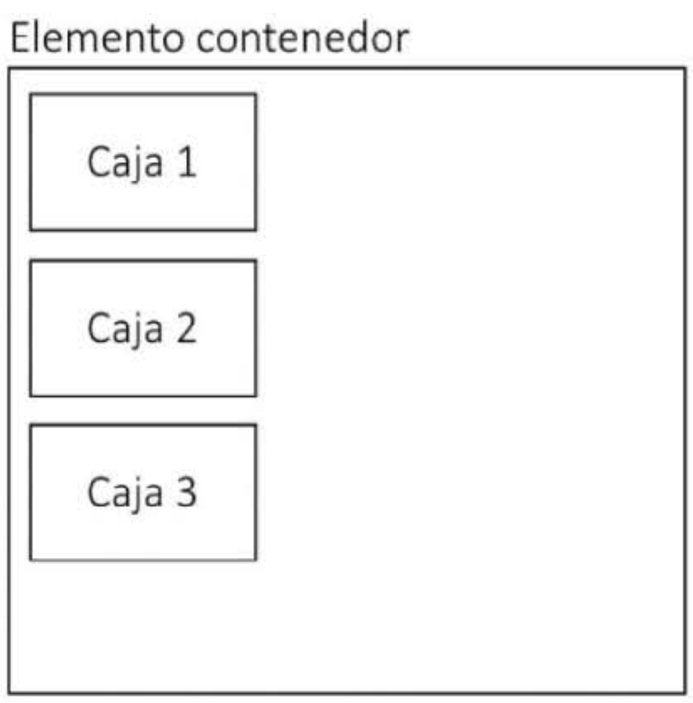


Figura 3.9 Posicionamiento de los elementos web.
Fuente: Elaboración propia.

Hay que tomar en cuenta que, al no definir un determinado posicionamiento, se estaría especificando el posicionamiento normal. Su formato es el siguiente:

```
selector {
   position: static;
}
```

3.19.2 *Posicionamiento relativo*

El posicionamiento relativo mantiene los elementos en el flujo normal, desplazándolos respecto a su posición por defecto. El control de las cajas se controla mediante las propiedades: **top**, **right**, **bottom** y **left**. Este tipo de posicionamiento no se usa habitualmente, ya que se puede conseguir el mismo resultado utilizando márgenes. Cabe tener en cuenta que los márgenes pueden tener un valor positivo o negativo, con los que, en la mayoría de casos, se obtendría un resultado igual.

Véase la implementación de la propiedad **position** para el selector **h1** en donde se le debe asignar -10px en la parte superior y 40 en la posición izquierda.

```
selector {
   position: relative;
   top: valortop;
   left: valorleft;
   right: valorright;
   bottom: valorbottom;
}
```

Donde:

- **selector**: Es el nombre del selector al que se le aplicarán las propiedades definidas en el CSS.
- **position:relative**: Indica la activación del posicionamiento relativo al selector seleccionado.

- **top**: Define la distancia superior hacia abajo.
- **left**: Determina la distancia desde la izquierda hacia la derecha.
- **right**: Define la distancia desde la derecha hacia la izquierda.
- **bottom**: Determina la distancia inferior hacia arriba.

3.19.3 *Posicionamiento absoluto*

El elemento que se posiciona de forma absoluta no ocupa espacio alguno en el documento. Esto significa que no deja un espacio vacío después de ser posicionado.

Para posicionar un elemento de forma absoluta, la propiedad **position** se establece como **absolute**. Posteriormente, se pueden usar las propiedades **left**, **right**, **top**, y **bottom** para colocar la caja. Su formato es:

```
selector{
    position:absolute;
    top: valortop;
    left: valorleft;
    right: valorright;
    bottom: valorbottom;
}
```

Donde:

- **selector:** Es el nombre del selector al que se le aplicarán las propiedades definidas en el CSS.
- **position:absolute**: Indica la activación del posicionamiento absoluto al selector seleccionado.
- **Top**: Define la distancia superior hacia abajo.
- **Left**: Determina la distancia desde la izquierda hacia la derecha.
- **Right**: Define la distancia desde la derecha hacia la izquierda.
- **Bottom**: Determina la distancia inferior hacia arriba.

3.19.4 *Posicionamiento fijo*

El posicionamiento fijo tiene muchos puntos en común con el posicionamiento absoluto, ya que al igual que este se indican las coordenadas en las que se desea posicionar el elemento.

La diferencia respecto con el posicionamiento absoluto es que, en el fijo, el bloque queda fijado en la página y no se mueve al utilizar la barra de desplazamiento. Es interesante su uso en ciertos tipos de menús, cabeceras o elementos que tengan que estar siempre visibles en la página. Presenta el siguiente formato:

```
selector{
    position: fixed;
    top: valortop;
    left: valorleft;
    right: valorright;
    bottom: valorbottom;
}
```

Donde:

- **selector**: Es el nombre del selector al que se le aplicarán las propiedades definidas en el CSS.
- **position:fixed**: Indica la activación del posicionamiento fijo al selector seleccionado.
- **top**: Define la distancia superior hacia abajo.
- **left**: Determina la distancia desde la izquierda hacia la derecha.
- **right**: Define la distancia desde la derecha hacia la izquierda.
- **bottom**: Determina la distancia inferior hacia arriba.

3.19.5 *Posicionamiento flotante*

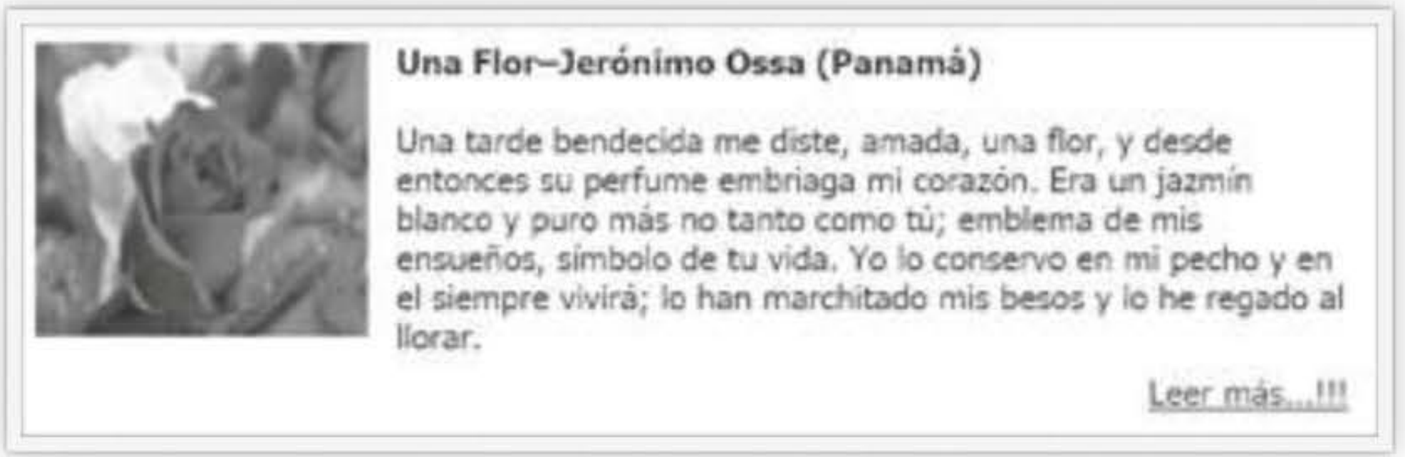
Cuando se habla de posición flotante, es algo complejo entenderlo; sin embargo, es el más utilizado. La mayoría de estructuras de páginas web complejas están diseñadas con el posicionamiento flotante, como se verá más adelante.

Cuando una caja se posiciona con el modelo de posicionamiento flotante, automáticamente se convierte en una caja flotante, lo que significa que se desplaza hasta la zona más a la izquierda o más a la derecha de la posición en la que originalmente se encontraba. Su formato es:

```
selector{
   float: posicion;
}
```

Caso desarrollado 1
Estilo de línea

Implemente el siguiente modelo de documento web usando estilos CSS.



Solución:

Para este caso, según el empleo de los estilos declarados en la misma línea de la etiqueta, solo se necesitará un solo archivo llamado **index.html**. A continuación, su código:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Estilo de Linea</title>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
</head>
<body>
   <table style='width: 500px; border:0; cellspacing:0; cellpadding:0'>
      <tr>
         <td style="font-family:Tahoma; font-size:12px; line-height:14px;
                  color:#2A2A2A' valign='top'>
            <section style="margin-left:20px; margin-top:20px;
                        margin-right:35px; margin-bottom:px'>
               <img src='images/foto.jpg' style="float: left; width: 119px;
                     height: 105px; margin-right:10px; margin-top:0px'>
               <strong>Una Flor-Jerónimo Ossa (Panamá)</strong><br>
               <br>
               Una tarde bendecida me diste, amada, una flor, y desde entonces
               su perfume embriaga mi corazón. Era un jazmín blanco y puro
               más no tanto como tú; emblema de mis ensueños, símbolo de tu
               vida. Yo lo conservo en mi pecho y en el siempre vivirá; lo
               han marchitado mis besos y lo he regado al llorar.<br>
               <footer style='float:right; margin-left:0px;margin-top:6px'>
                  <a href='#'>Leer más...!!!</a>
               </footer>
            </section>
         </td>
      </tr>
   </table>
</body>
</html>
```

Caso desarrollado 2
Estilo a nivel de página

Implemente el siguiente modelo de documento web usando estilos CSS.



FUEGO Y HIELO
(poema)
Robert Frost

Unos dicen que el mundo terminará en fuego, otros dicen que en hielo. Por lo que he gustado del deseo, estoy con los partidarios del fuego.

Pero si tuviera que sucumbir dos veces, creo saber bastante acerca del odio como para decir que, en la destrucción, el hielo también es poderoso. Y bastaría.

Leer más...!!

Solución:

Para este caso, al igual que el anterior y según el uso de los estilos declarados en la misma página, solo se necesitará un solo archivo llamado **index.html**. Obsérvese el código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset=UTF-8>
      <title>Manejo de Estilo de Página</title>
      <style>
         /*Estilos de Etiqueta*/
         table{
            border:0;
            cellspacing:0;
            cellpadding:0;
         }

         td {
```

```
        width: 250px;
        text-align: justify;
        font-family:tahoma;
        font-size:12px;
        color: black;
    }

    a {text-decoration:none;
        color:red;
    }

    header{
        text-align: center;
    }

    body {
        margin:0px;
        padding:0px;
    }

    section{
        margin:24px 18px 0px 21px;
    }

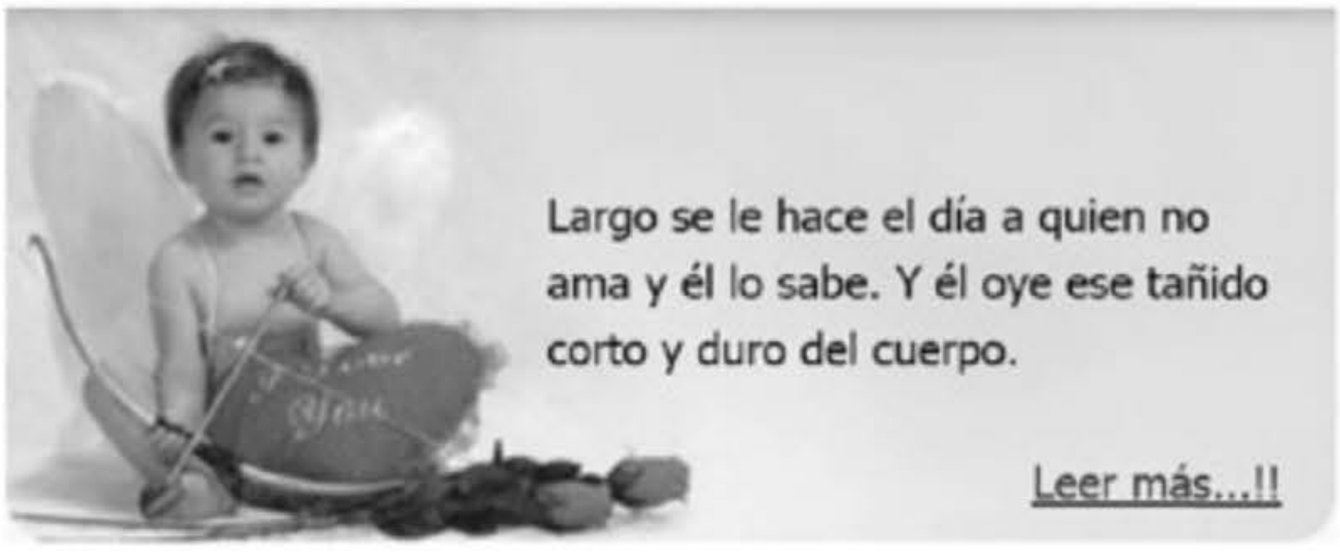
    /*Estilo de Identificación*/
    #imagen{
        width: 250px;
    }

    /*Estilo por clase*/
    .link {
        font-size:12px;
    }
    </style>
</head>
<body>
    <table >
        <tr>
            <td>
                <section>
                    <img src='images/foto.jpg' id='imagen'><br><br>
                    <header>
                        <strong>FUEGO Y HIELO</strong><br>
                        (poema)<br>
                        Robert Frost<br><br>
                    </header>
                    Unos dicen que el mundo terminará en fuego, otros dicen que en
                    hielo. Por lo que he gustado del deseo, estoy con los
                    partidarios del fuego. <br><br>
                    Pero si tuviera que sucumbir dos veces, creo saber bastante
```

```
                    acerca del odio como para decir que, en la destrucción, el hielo
                    también es poderoso. Y bastaría.<br>
                    <br>
                    <a href='#' class='link'>Leer más...!!</a>
                </section>
            </td>
        </tr>
    </table>
</body>
</html>
```

Caso desarrollado 3
Estilo externo

Implemente el siguiente modelo de documento web usando estilos CSS.



Consideración:

Para este caso, los estilos se guardarán en un archivo externo.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body {
    padding:0;
    margin:10;
    color:#000000;
    font-family:Tahoma;
    font-size:12px;
    line-height:18px;
}
```

```
td{
    padding: 0;
    background-image:url(images/foto.jpg);
    width:359px;
    height:144px;
    float: top;
}

.link{
    color:blue;
    float: right;
}

section{
    margin-left:146px;
    margin-top:40px;
    margin-right:15px;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Manejo Estilos Externos</title>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
<body>
        <table>
            <tr>
                <td>
                    <section>
                        Largo se le hace el día a quien no ama y él lo sabe. Y él oye
                        ese tañido corto y duro del cuerpo.<br>
                        <br>
                        <a href='#' class='link'>Leer más...!!</a>
                    </section>
                </td>
            </tr>
        </table>
    </body>
</html>
```

Caso desarrollado 4
Estilo importado

Implemente el siguiente modelo de documento web usando estilos CSS.



Consideración:

Para este caso, los estilos se guardarán en un archivo externo.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```css
/*Estilos de Etiqueta*/
body {
    margin: 10px;
    padding: 0px;
    text-align: left;
    font-family: tahoma;
    font-size: 12px;
    color: black;
    line-height: 14px;
}

section {
    width: 300px;
}

a {
    text-decoration:underline;
    color:#519FC3;
}

/*Estilos de Identificación*/
#marco1{
    background:url(images/bg5.jpg) bottom left no-repeat;
    margin:12px 10px 15px 19px;
}
```

```
#marco2{
    margin:14px 0px 0px 90px;
}

/*Estilos de Clase*/
.linea{
    width:250px;
    height:1px;
}

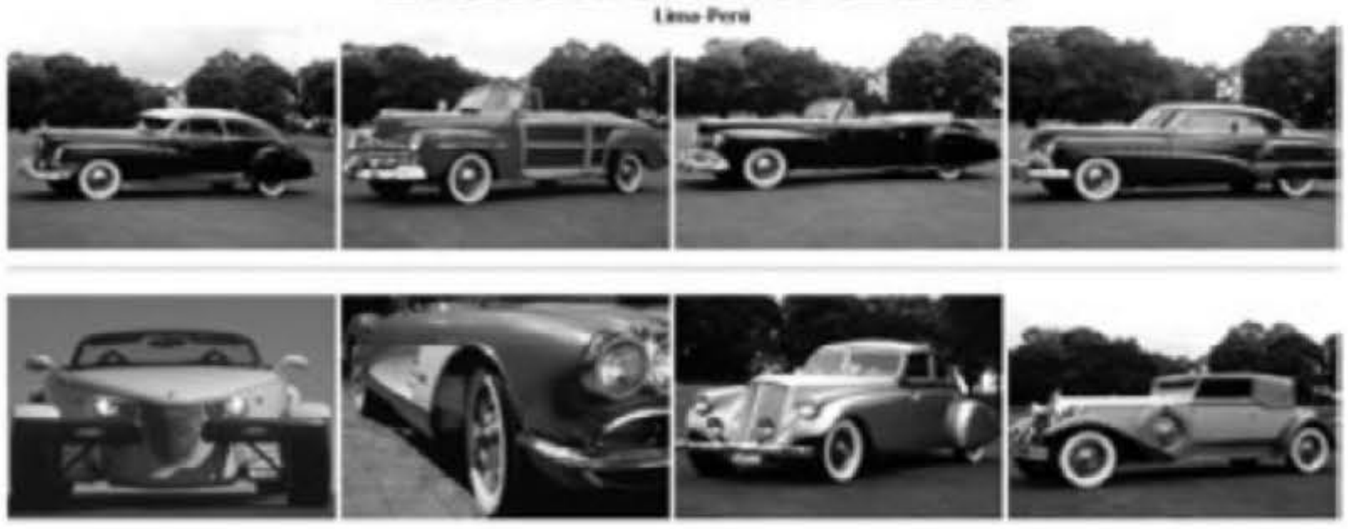
.link{
    float: right;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Manejo Estilo Importado</title>
        <meta charset='UTF-8'>
        <style type='text/css'>
            @import url('estilo.css');
        </style>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section id='marco1' >
            <section id='marco2'>
                <strong>La Amistad</strong><br>
                Los amigos son compañeros de viaje que nos ayudan a avanzar por el
                camino de una vida más feliz. <br>
                <img src='images/l.gif' class='linea'><br><br>
                <strong>Más que amistad</strong><br>
                La amistad es más difícil y más rara que el amor. Por eso, hay que
                salvarla como sea.<br><br>
                <strong><a href='#' class='link'>Leer más...!!</a></strong>
            </section>
        </section>
    </body>
</html>
```

Caso desarrollado 5
Posicionamiento normal

Implemente el siguiente modelo de documento web aplicando el posicionamiento normal del CSS.



Consideraciones:

Para este caso, se debe contar con ocho imágenes guardadas en la carpeta **images** del proyecto. Asimismo, se debe emplear el estilo externo; por lo tanto, se necesita un archivo de estilos llamado **estilo.css** y otro que contenga todo el script HTML5 llamado **index.html**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
h1{
    font-family: 'Tahoma';
    position: static;
    text-align: center;
    border: #663366 10px;
    margin: 0px;
    color:#000;
}

h4 {
    font-family: 'Tahoma';
    position:static;
    text-align:center;
    border: #663366 5px;
    color:#000;
    margin: 0px;
}
```

```
img {
    position:static;
    width: 286px;
    height:194px;
}

.linea{
    width:1155px;
    height:2px;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Posicionamiento Normal</title>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <h1>EXHIBICIÓN DE AUTOS CLÁSICOS</h1>
        <h4>Lima-Perú</h4>
        <img src='images/auto1.jpg'/>
        <img src='images/auto2.jpg'/>
        <img src='images/auto3.jpg'/>
        <img src='images/auto4.jpg'/>
        <img src='images/l.gif' class='linea'><br><br>
        <img src='images/auto5.jpg'/>
        <img src='images/auto6.jpg'/>
        <img src='images/auto7.jpg'/>
        <img src='images/auto8.jpg'/>
    </body>
</html>
```

Caso desarrollado 6
Posicionamiento relativo

Implemente el siguiente modelo de documento web aplicando el posicionamiento relativo del CSS.



Consideraciones:

Para este caso, se debe contar con cuatro imágenes guardadas en la carpeta **images** del proyecto. Asimismo, se debe utilizar el estilo externo; por ende, se necesita un archivo de estilos llamado **estilo.css** y otro que contenga todo el script HTML5 llamado **index.html**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body{
    font-family: 'Tahoma';
}

h1{
    position: relative;
    top: -30px;
    left: 0px;
}

h4 {
    position:relative;
    text-align: right;
    top: -50px;
    color:#000;
}
```

```
img {
    position:relative;
    top: -60px;
    width: 180px;
    height:194px;
}

section{
    width: 735px;
    height: 240px;
    background-color: gainsboro;
    padding: 15px;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Posicionamiento Relativo</title>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>CONSULTORIO VETERINARIO</h1>
            <h4>* LO MEJOR PARA SU MASCOTA *</h4>
            <img src='images/animal1.jpg'/>
            <img src='images/animal2.jpg'/>
            <img src='images/animal3.jpg'/>
            <img src='images/animal4.jpg'/>
        </section>
    </body>
</html>
```

Caso desarrollado 7
Posicionamiento absoluto

Implemente el siguiente modelo de documento web aplicando el posicionamiento absoluto del CSS.



Consideraciones:

Para este caso, se debe contar con cuatro imágenes guardadas en la carpeta **images** del proyecto. Asimismo, se debe utilizar el estilo externo; por ende, se necesita un archivo de estilos llamado **estilo.css** y otro que contenga todo el script HTML5 llamado **index.html**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body{
    font-family: 'Tahoma';
}
section{
    width: 870px;
    background-color: aliceblue;
    padding: 10px;
}
h1{
    text-align:center;
}
```

```
h3 {
    position: absolute;
    padding: 10px;
    background-color:#000;
    color:#fff;
    top: 242px;
    left: 300px;
    z-index:1;

}

img {
    position:relative;
    width: 170px;
    height:194px;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Posicionamiento Absoluto</title>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>COMIDA PERUANA - 5 TENEDORES</h1>
            <h3>ATENDEMOS TODOS LOS DÍAS</h3>
            <img src='images/comida1.jpg'/>
            <img src='images/comida2.jpg'/>
            <img src='images/comida3.jpg'/>
            <img src='images/comida4.jpg'/>
            <img src='images/comida5.jpg'/>
        </section>
    </body>
</html>
```

Caso desarrollado 8
Posicionamiento flotante

Implemente el siguiente modelo de documento web aplicando el posicionamiento flotante del CSS.



Consideraciones:

Para este caso, se debe contar con cuatro imágenes guardadas en la carpeta **images** del proyecto. Asimismo, se debe usar el estilo externo; por lo tanto, se necesita un archivo de estilos llamado **estilo.css** y otro que contenga todo el script HTML5 llamado **index.html**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 5px;
}

section{
    width: 600px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

Caso desarrollado 8
Posicionamiento flotante

Implemente el siguiente modelo de documento web aplicando el posicionamiento flotante del CSS.



Consideraciones:

Para este caso, se debe contar con cuatro imágenes guardadas en la carpeta **images** del proyecto. Asimismo, se debe usar el estilo externo; por lo tanto, se necesita un archivo de estilos llamado **estilo.css** y otro que contenga todo el script HTML5 llamado **index.html**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 5px;
}

section{
    width: 600px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

```
img {
    float:left;
    padding: 10px;
    width: 220px;
    height:180px;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Posicionamiento Flotante</title>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
            <img src='images/pisco.jpg'/>
            <p>
                La bebida nacional del Perú es el pisco. Se trata de un destilado o
                aguardiente de uva cuya producción se inició a fines del siglo XVI en
                la provincia de Pisco (actualmente ubicada en el departamento de Ica)
                y se extendió posteriormente a las haciendas de la costa peruana.
            </p>
            <p>
                <a href='#'>Leer más..!!</a>
            </p>
        </section>
    </body>
</html>
```



CAPÍTULO 4

Formateando un texto web

Competencias

- Implementa documentos web usando las etiquetas básicas texto y las etiquetas para fragmentos de textos.
- Elabora documentos web empleando las etiquetas de texto de modificación de significado y las etiquetas orientadas al código.

Contenido

4.1 Formato básico
4.2 Etiquetas para fragmentos de textos
4.3 Etiquetas de modificación de significado
4.4 Etiquetas orientadas a código
- Caso desarrollado 1 : Uso de la etiqueta <header> y <h1>...<h6>
- Caso desarrollado 2 : Uso de la etiqueta
- Caso desarrollado 3 : Uso de la etiqueta <pre>
- Caso desarrollado 4 : Uso de la etiqueta <pre> para listado
- Caso desarrollado 5 : Uso de la etiqueta
- Caso desarrollado 6 : Uso de la etiqueta
- Caso desarrollado 7 : Uso de la etiqueta <blockquote> y <cite>
- Caso desarrollado 8 : Uso de la etiqueta <dfn> y <abbr>
- Caso desarrollado 9 : Uso de la etiqueta <time>
- Caso desarrollado 10 : Uso de la etiqueta <code>
- Caso desarrollado 11 : Uso de la etiqueta <var>, <samp> y <kbd>
- Caso desarrollado 12 : Uso de la etiqueta <sup>
- Caso desarrollado 13 : Uso de la etiqueta <mark>
- Usando caracteres especiales

4.1 Formato básico

4.1.1 *Las etiquetas de cabecera <h1>...<h6>*

La etiqueta **<h>** resalta los textos como corresponde a los títulos de un determinado artículo, comentario, etc. Para implementar cabeceras dentro de un documento web se emplean las siguientes etiquetas: **<header>**, **<hgroup>**, **<h1>**, **<h2>**, **<h3>**, **<h4>**, **<h5>** y **<h6>**. Su formato es:

```
<h1> Contenido de la cabecera </h1>
<h2> Contenido de la cabecera </h2>
<h3> Contenido de la cabecera </h3>
<h4> Contenido de la cabecera </h4>
<h5> Contenido de la cabecera </h5>
<h6> Contenido de la cabecera </h6>
```

Cada etiqueta presenta un tamaño predeterminado iniciando con **<h1>**, como el tamaño más grande, hasta **<h6>**, como el tamaño más pequeño. Por otro lado, se puede agrupar las etiquetas de cabecera mediante la etiqueta **<header>** de la siguiente manera:

```
<header>
   <h1> Contenido de la cabecera </h1>
   <h2> Contenido de la cabecera </h2>
   <h3> Contenido de la cabecera </h3>
   <h4> Contenido de la cabecera </h4>
   <h5> Contenido de la cabecera </h5>
   <h6> Contenido de la cabecera </h6>
</header>
```

Ejemplo 1

Véase el script correspondiente para la siguiente imagen:

FORMATEANDO TEXTOS HTML

ETIQUETAS DE CABECERA

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
   font-family: Tahoma;
}

h1{
   position: relative;
   top: 0px;
   left: 0px;
}
```

```
h3 {
    position:relative;
    text-align: left;
    top: -20px;
    color:#000;
    letter-spacing: 12px;
}
```

En el archivo **index.html** se escribe el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <h1>FORMATEANDO TEXTOS HTML</h1>
        <h3>ETIQUETAS DE CABECERA</h3>
    </body>
</html>
```

También se podría agrupar dentro de la etiqueta **<header>** de la siguiente manera:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <h1>FORMATEANDO TEXTOS HTML</h1>
            <h3>ETIQUETAS DE CABECERA</h3>
        </header>
    </body>
</html>
```

Y si se integra con la etiqueta **<hgroup>**, el código sería de la siguiente manera:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
```

```
    <body>
        <header>
            <hgroup>
                <h1>FORMATEANDO TEXTOS HTML</h1>
                <h3>ETIQUETAS DE CABECERA</h3>
            </hgroup>
        </header>
    </body>
</html>
```

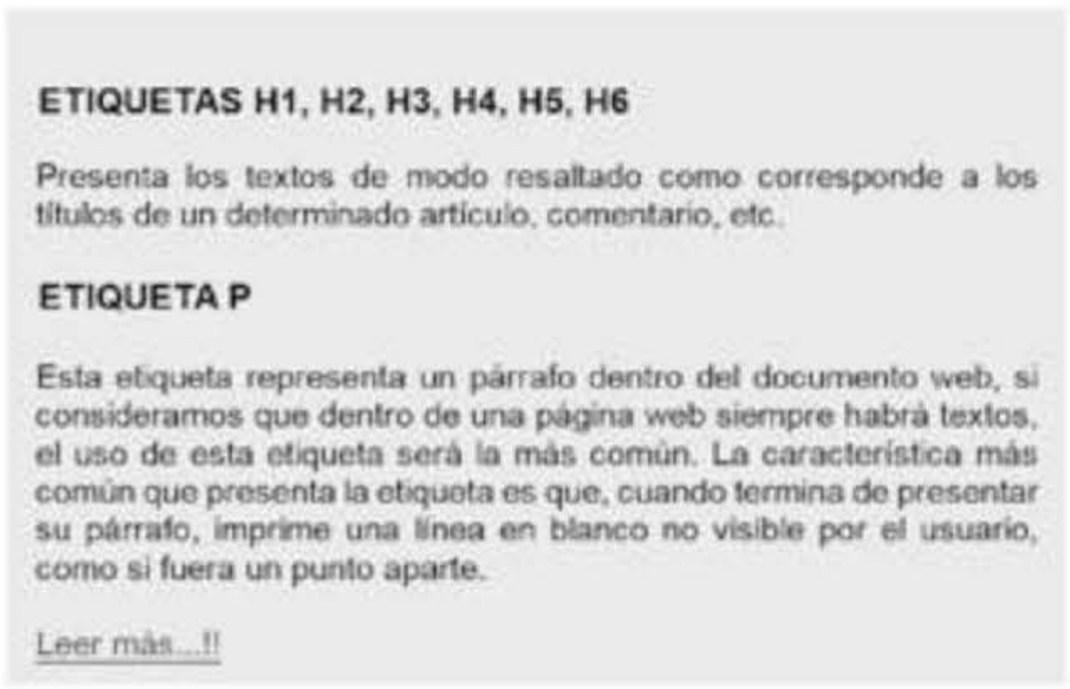
4.1.2 *La etiqueta de párrafo <p>*

Esta etiqueta presenta un párrafo dentro del documento web. Como se sabe, dentro de una página web siempre habrá textos, por ende, el uso de esta etiqueta será la más común.

La característica más frecuente que presenta la etiqueta `<p>` es que, cuando termina de presentar su párrafo, imprime una línea en blanco no visible por el usuario.

Ejemplo 2

› Véase el uso de la etiqueta `<p>` dentro de un `<section>`:



En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: Tahoma;
}

section {
    width: 500px;
    margin:5px;
    padding: 15px;
    background-color: bisque;
}
```

```
h3{
    text-transform: uppercase;
}

p{
    text-align: justify;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h3>Etiquetas h1, h2, h3, h4, h5, h6</h3>
            <p>Presenta los textos de modo resaltado como corresponde a los
            títulos de un determinado artículo, comentario, etc. </p>
            <h3>Etiqueta p</h3>
            <p>Esta etiqueta representa un párrafo dentro del documento web, si
            consideramos que dentro de una página web siempre habrá textos, el
            uso de esta etiqueta será la más común. La característica más
            común que presenta la etiqueta es que, cuando termina de presentar
            su párrafo, imprime una línea en blanco no visible por el usuario,
            como si fuera un punto aparte.</p>
            <a href='#' class='link'>Leer más...!!</a>
        </section>
    </body>
</html>
```

› Véase el uso de la etiqueta **<p>** integrado a las listas:

El uso de p debería limitarse, únicamente debería ser utilizado cuando no exista otro elemento más específico.

- El elemento p siempre debe tener una etiqueta de inicio, pero la etiqueta de cierre puede ser omitida si está seguido por un elemento address, article, aside, blockquote, dir, div, dl, fieldset, footer, form, h1, h2, h3, h4, h5, h6, header, hr, menu, nav, ol, p, pre, section, table, o ul, o si la etiqueta que la contiene termina su contenido y se cierra.
- El navegador realiza un ajuste automático de los párrafos al ancho de pantalla determinado por la ventana del programa navegador.

Fuente: https://www.w3schools.com

En el archivo **estilo.css** se inserta el siguiente código:

```
body{
    font-family: Tahoma;
}

section {
    width: 500px;
    margin:5px;
    padding: 15px;
    background-color: bisque;
}

p{
    text-align: justify;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <p>El uso de p debería limitarse, únicamente debería ser utilizado
            cuando no exista otro elemento más específico.</p>
            <ul>
                <li>
                    <p>El elemento p siempre debe tener una etiqueta de inicio, pero
                    la etiqueta de cierre puede ser omitida si está seguido por un
                    elemento address, article, aside, blockquote, dir, div, dl,
                    fieldset, footer, form, h1, h2, h3, h4, h5, h6, header, hr, menu,
                    nav, ol, p, pre, section, table, o ul, o si la etiqueta que la
                    contiene termina su contenido y se cierra.</p>
                </li>
                <li>
                    <p>El navegador realiza un ajuste automático de los párrafos
                    al ancho de pantalla determinado por la ventana del
                    programa navegador.</p>
                </li>
            </ul>
            <p>Fuente: https://www.w3schools.com</p>
        </section>
    </body>
</html>
```

4.1.3 *La etiqueta de cambio de línea
*

La etiqueta ** ** agrega un renglón en blanco y luego muestra el siguiente elemento web. Esto quiere decir que, hasta que no se encuentre la etiqueta ** ** de cierre, no habrá cambio de línea. Al usar la etiqueta **
** se puede forzar a un cambio de línea dentro o fuera de un párrafo. Por otro lado, los espacios en blanco adicionales o cambios de líneas forzados no serán interpretados por el navegador.

Ejemplo 3

Véase el uso de la etiqueta **
** dentro de un **<section>**.

Manuel Torres Remon
Consultor Informático

Direct Number: (051) 998-588547
Movil Number: (051) 985-458789
House: (051) 522-1123

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: Tahoma;
}

section {
    width: 250px;
    margin:5px;
    padding: 15px;
    background-color: bisque;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
</head>
<body>
    <section>
        <strong>Manuel Torres Remon<br>
        Consultor Informático</strong>
        <br>
        <br style='line-height:25px'>
```

```
            Direct Number: (051) 998-588547<br>
            Movil Number: (051) 985-458789<br>
            House: (051) 522-1123
        </section>
    </body>
</html>
```

4.1.4 *La etiqueta de preformateado de texto <pre>*

Cuando se usa la etiqueta **<p>**, los navegadores no interpretan los espacios en blanco ni el cambio de línea. La etiqueta **<pre>** determina que el navegador debe dejar los espacios en blanco intactos, mostrar el texto con una fuente de ancho fijo y deshabilitar los saltos de línea. Su formato es:

```
<pre>
    <!- contenido ->
</pre>
```

Para usar esta etiqueta primero hay que revisar qué entienden los navegadores:

- Los espacios en blanco permanecen intactos, esto quiere decir que los espacios consecutivos no deben ser eliminados por el navegador.
- El texto recibe el nombre de "texto preformateado" y se le asigna una fuente de ancho fijo para todos sus caracteres.
- El cambio de línea dentro de la página web es deshabilitado, es decir, un párrafo puede mostrarse fácilmente en una sola línea. El cambio de línea solo puede ser efectuado usando la etiqueta **
**.

Ejemplo 4

Véase el uso de la etiqueta **<pre>** dentro de un **<section>**:



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: Tahoma;
}

section {
    width: 330px;
    margin:2px;
    padding: 2px;
    background-color: bisque;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='UTF-8'>
      <link href='estilo.css' rel='stylesheet' type='text/css'/>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <section>
         <pre>&copy;2019 EditorialMacro &bull; HTML5 y CSS3</pre>
      </section>
   </body>
</html>
```

4.2 Etiquetas para fragmentos de textos

4.2.1 *La etiqueta *

Permite dar un énfasis fuerte sobre un contenido textual dentro del documento web, produciendo un efecto visual de negrita sobre dicho texto. Hay que considerar que **** separa la presentación del contenido, puesto que la primera dependerá del navegador o de la hoja de estilo. Esto se menciona porque existe también la etiqueta **<b>**, que también aplica un énfasis fuerte (negrita), pero lo realiza como una presentación visual directa.

El uso de la etiqueta **<b>** es permitida en HTML5, pero no es totalmente recomendada. Se dice que **<b>** puede usarse como último recurso. La etiqueta **** se usa principalmente para definir un texto importante en el documento. Su formato es:

```
<strong>
   <!– contenido resaltado –>
</strong>
```

El equivalente para negrita en el estilo CSS es:

```
#resaltado{
   font-weight:800;
}
```

O también:

```
#resaltado{
   font-weight:bold;
}
```

Ejemplo 5

Véase el uso de la etiqueta **** dentro de un **<section>**:



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body{
    font-family: 'Tahoma';
    margin: 5px;
}

section{
    width: 600px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
            <p>
                La bebida nacional del <strong>Perú</strong> es el pisco. Se trata de
                un destilado o aguardiente de uva cuya producción se inició a fines del
```

```
            <strong>siglo XVI</strong> en la provincia de <strong>Pisco</strong>
            (actualmente ubicada en el departamento de <strong>Ica</strong>) y se
            extendió posteriormente a las haciendas de la costa peruana.
        </p>
        <p>
            <strong><a href='#'>Leer mas..!!</a></strong>
        </p>
    </section>
</body>
</html>
```

4.2.2 *La etiqueta *

El uso de esta etiqueta se debió a la necesidad de poder aplicar un formato especial para hacer más llamativo un determinado párrafo. La etiqueta `<em>` permite realizar algo parecido aplicando un énfasis sobre un determinado párrafo. Este énfasis tiene la característica de mostrar el texto en cursiva. Su formato es:

```
<em>
    <!- contenido de párrafo ->
</em>
```

Otra de las formas de aplicar este énfasis es usando estilos CSS de la siguiente manera:

```
#cursiva{
    font-style: italic;
}
```

Ejemplo 6

Véase el uso de la etiqueta `<em>` dentro de un `<section>`:

PISCO SOUR

La bebida nacional del *Perú* es el pisco. Se trata de un destilado o aguardiente de uva cuya producción se inició a fines del *siglo XVI* en la provincia de *Pisco* (actualmente ubicada en el departamento de *Ica*) y se extendió posteriormente a las haciendas de la costa peruana.

Leer más..!!

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 5px;
}

section{
    width: 600px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
            <p>
                La bebida nacional del <em>Perú</em> es el pisco. Se
                trata de un destilado o aguardiente de uva cuya producción se inició a
                fines del <em>siglo XVI</em> en la provincia de <em>Pisco</em>
                (actualmente ubicada en el departamento de <em> Ica</em>) y se extendió
                posteriormente a las haciendas de la costa peruana.
            </p>
            <p>
                <em><a href='#'>Leer más..!!</a></em>
            </p>
        </section>
    </body>
</html>
```

4.2.3 *La etiqueta <mark>*

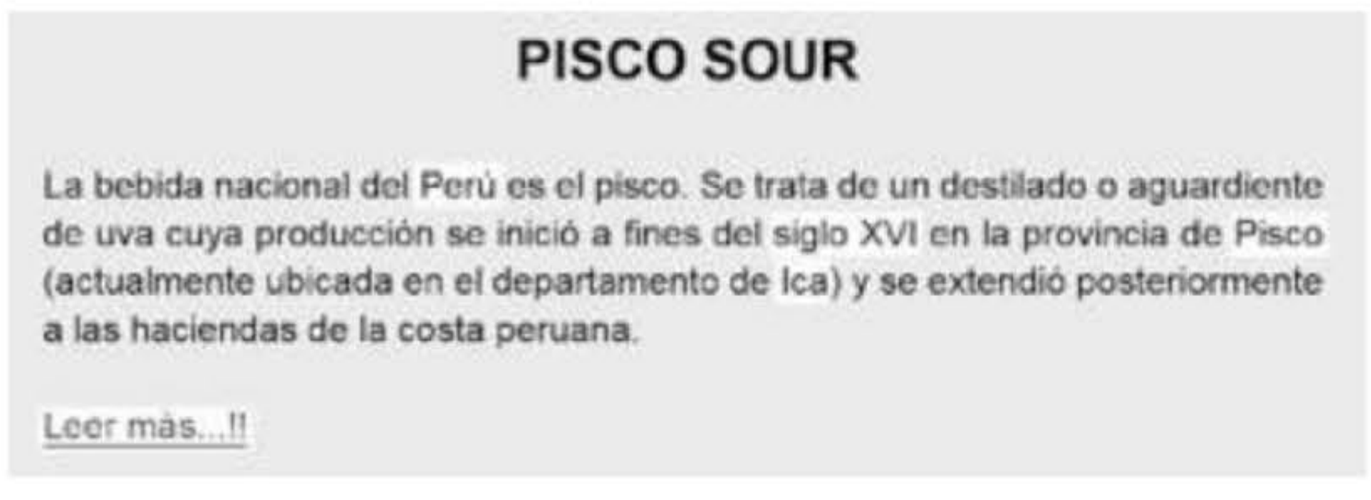
Permite resaltar un texto importante dentro de un párrafo para diferenciarlo de los demás, mostrando un color parecido al de un plumón resaltador. Su formato es:

```
<mark>
    <!- contenido del texto resaltado ->
</mark>
```

A diferencia de la etiqueta **** y ****, que permiten dar importancia al texto aplicando y negrita y cursiva, respectivamente, la etiqueta **<mark>** resalta el texto de color amarillo.

Ejemplo 7

Véase el uso de la etiqueta **<mark>** dentro de un **<section>**.



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 5px;
}

section{
    width: 600px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
            <p>
                La bebida nacional del <mark>Perú</mark> es el pisco. Se trata de
                un destilado o aguardiente de uva cuya producción se inició a
                fines del <mark>siglo XVI</mark> en la provincia de
                <mark>Pisco</mark> (actualmente ubicada en el departamento de
                <mark>Ica</mark>) y se extendió posteriormente a las haciendas de
                la costa peruana.
            </p>
            <p>
                <mark><a href='#'>Leer mas..!!</a></mark>
            </p>
        </section>
    </body>
</html>
```

4.2.4 *La etiqueta <i>*

Se utiliza para dar énfasis a un texto dentro de un documento web. Este tipo de énfasis hace que el texto se muestre en cursiva. Al igual que la etiqueta **<b>**, se recomienda no emplearla y usar de preferencia los estilos para aplicar cursiva en HTML5.

En HTML5 se sugiere usar la etiqueta **<i>** como último recurso y en el caso de que ninguna otra etiqueta sea la más apropiada, ya que podría usarse la etiqueta ****.

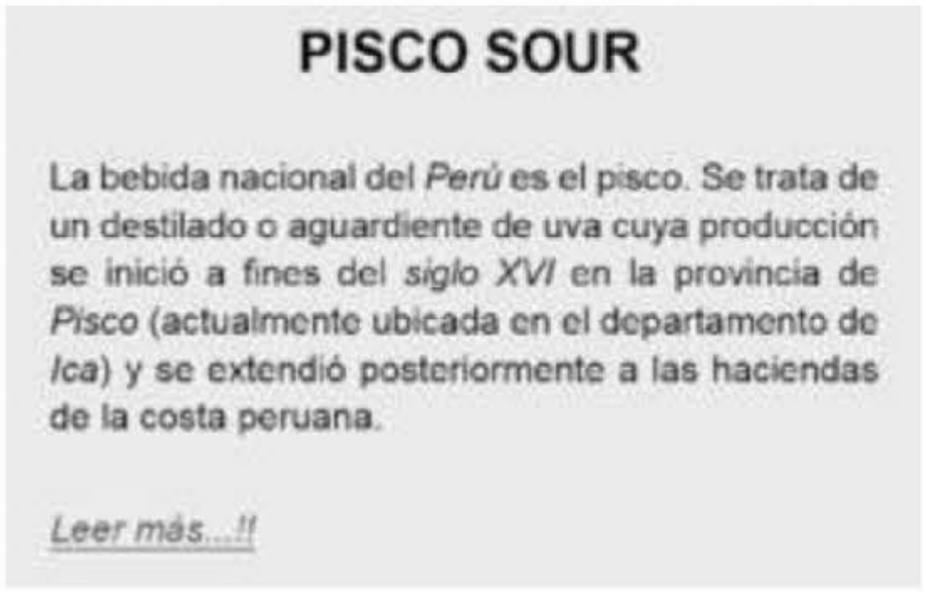
```
<i>
    <!– contenido en cursiva –>
</i>
```

Su equivalente en estilo CSS sería:

```
#cursiva{
    font-style:italic;
}
```

Ejemplo 8

Véase el uso de la etiqueta **<i>** dentro de un **<section>**:



Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 15px;
}

section{
    width: 400px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
```

```
<p>
   La bebida nacional del <i>Perú</i> es el pisco. Se
   trata de un destilado o aguardiente de uva cuya producción se
   inició a fines del <i>siglo XVI</i> en la provincia de
   <i>Pisco</i> (actualmente ubicada en el departamento de
   <i>Ica</i>) y se extendió posteriormente a las
   haciendas de la costa peruana.
</p>
<p>
   <i><a href='#'>Leer más..!!</a></i>
</p>
</section>
</body>
</html>
```

4.2.5 *La etiqueta <b>*

Se emplea para resaltar un texto importante en un documento web aplicando el estilo negrita. En HTML5 se recomienda usar los estilos para aplicar este formato.

Se sugiere usar la etiqueta `<b>` como último recurso y en el caso de que ninguna otra etiqueta sea la más apropiada. Para definir la apariencia de los títulos se recomienda que se utilicen las etiquetas `<h1>...<h6>`, mientras que para destacar el texto es preferible utilizar etiquetas como `<strong>`, `<mark>` o `<em>`. Su formato es:

```
<b>
   <!- contenido en negrita ->
</b>
```

El equivalente en estilo CSS para negrita sería:

```
#negrita{
   font-weight:bold;
}
```

Ejemplo 9

Véase el uso de la etiqueta `<b>` dentro de un `<section>`:

PISCO SOUR

La bebida nacional del **Perú** es el pisco. Se trata de un destilado o aguardiente de uva cuya producción se inició a fines del **siglo XVI** en la provincia de **Pisco** (actualmente ubicada en el departamento de **Ica**) y se extendió posteriormente a las haciendas de la costa peruana.

Leer más...!!

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 15px;
}

section{
    width: 400px;
    background-color: lightblue;
}

h1{
    text-align:center;
}

p{
    padding: 10px;
    text-align:justify;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>PISCO SOUR</h1>
            <p>
                La bebida nacional del <b>Perú</b> es el pisco. Se
                trata de un destilado o aguardiente de uva cuya producción se
                inició a fines del <b>siglo XVI</b> en la provincia de
                <b>Pisco</b> (actualmente ubicada en el departamento de
                <b>Ica</b>) y se extendió posteriormente a las
                haciendas de la costa peruana.
            </p>
            <p>
                <b><a href='#'>Leer más..!!</a></b>
            </p>
        </section>
    </body>
</html>
```

4.2.6 *La etiqueta *

Es una etiqueta sin valor semántico, pues permite asignar un formato a un determinado texto. Esta etiqueta brindará un nombre a un conjunto de elementos en nuestro documento web. Cuando el navegador interprete el contenido del ****, verificará si tiene asignado un estilo y lo aplicará solo al contenido del ****. Su formato es:

```
<span>
   <!- contenido del span->
</span>
```

Hay que tener en cuenta que colocar un **** no afecta la semántica del párrafo; por lo tanto, se debe crear un estilo que haga referencia desde el **** para aplicar un estilo particular a dicho bloque. Su formato es el siguiente:

```
<span id='nombre del estilo'>
   <!- contenido del span->
</span>
```

Ejemplo 10

Véase el uso de la etiqueta **** dentro de un **<section>**:

FUEGO Y HIELO
(poema)
Robert Frost

Unos dicen que el mundo terminará en fuego, otros dicen que en hielo.
Por lo que he gustado del deseo, estoy con los partidarios del fuego.
Pero si tuviera que sucumbir dos veces, creo saber bastante acerca del odio como para decir que en la destrucción el hielo también es poderoso. Y bastaría.

Leer más...!!

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
   margin:0px;
   padding:0px;
   font-family: Tahoma;
   font-size: 14px;
}
```

```
section{
    margin:25px;
    width: 250px;
    text-align: justify;
}

span.titulo{
    color: red;
    font-weight:bold;
    display: flex ;
    justify-content: center;
}

span.subtitulo{
    font-weight:bold;
    display: flex ;
    justify-content: center;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
</head>
<body>
    <section>
        <span class='titulo'>FUEGO Y HIELO</span>
        <span class='subtitulo'>(poema)</span>
        <span class='subtitulo'>Robert Frost</span><br>
        <span>Unos dicen que el mundo terminará en fuego, otros dicen que en
        hielo. Por lo que he gustado del deseo, estoy con los partidarios
        del fuego.</span> <br><br>
        <span>Pero si tuviera que sucumbir dos veces, creo saber bastante
        acerca del odio como para decir que en la destrucción el hielo
        también es poderoso. Y bastaría.<br></span>
        <br>
        <a href='#' class='link'>Leer más...!!</a>
    </section>
</body>
</html>
```

4.2.7 *La etiqueta <cite>*

Hoy en día existen infinidad de mensajes en Internet y la credibilidad de estos depende tanto de la reputación que tenga el autor, así como de las fuentes citadas y su calidad. Las citas se referencian mediante la etiqueta **<cite>**, el cual es interpretado y mostrado por el navegador en forma de letra cursiva.

Las características más importantes son:

- Sirve para marcar el origen de la cita, pudiendo ser una persona, una obra literaria, etc.
- Sirve únicamente para mencionar títulos de obras mencionadas en el texto.
- Muestra el texto en cursiva.
- No permite saltos de línea ni anterior ni posterior.
- El contenido de la cita permanece en la misma línea que el texto anterior y posterior.

Su formato es:

```
<cite>
   <!– contenido de la cita –>
</cite>
```

Tómese en cuenta que la citas largas o extensas se implementan mediante la etiqueta **<blockquote>** y cuenta con las siguientes características:

- Permite hacer saltos de línea tanto anteriores como posteriores a la cita.
- Permite establecer un **padding** y un **margin** al contenido de la cita larga.

Su formato es:

```
<blockquote>
   <!– contenido de la cita larga –>
</blockquote>
```

Ejemplo 11

Véase el uso de la etiqueta **<cite>** y **<blockquote>** dentro de un **<section>**:

BLOCKQUOTE Y CITE

Permite a los autores insertar citas en forma de bloques de contenido, usualmente compuestas por un párrafo, un grupo de párrafos o un conjunto de muchos otros elementos incluyendo imágenes (img), tablas (table) y artículos (article), entre otros.

> Esta es la principal diferencia entre este elemento y q, que está diseñado para citar únicamente líneas de texto.

Aunque no es requerido por el estándar, los navegadores pueden formatear el texto en la cita de alguna manera. La forma más común de representar a los elementos blockquote es con una sangría. *Tomado de: http://www.htmlquick.com/es*

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 15px;
}

section{
    width: 400px;
    padding: 15px;
    background-color: lightblue;
    text-align:justify;
}

h1{
    text-align:center;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>BLOCKQUOTE y CITE</h1>
            Permite a los autores insertar citas en forma de bloques de
            contenido, usualmente compuestas por un párrafo, un grupo de párrafos
            o un conjunto de muchos otros elementos incluyendo imágenes (img),
            tablas (table) y artículos (article), entre otros.
            <blockquote>
                Esta es la principal diferencia entre este elemento y q,
                que está diseñado para citar únicamente líneas de texto.
            </blockquote>
            Aunque no es requerido por el estándar, los navegadores pueden
            formatear el texto en la cita de alguna manera. La forma más común de
            representar a los elementos blockquote es con una sangría.
            <cite>Tomado de: http://www.htmlquick.com/es</cite>
        </section>
    </body>
</html>
```

4.3 Etiquetas de modificación de significado

4.3.1 *La etiqueta <sup>*

Se utiliza para aplicar superíndice al texto, mayormente se emplea para las fórmulas. Su formato es:

```
<sup>
    <!– contenido del superíndice –>
</sup>
```

Ejemplo 12

Véase el uso de la etiqueta `<sup>` dentro de un `<section>`:

Fórmula cuadrática

Una ecuación de segundo grado o ecuación cuadrática de una variable es una ecuación que tiene la forma de una suma algebraica de términos cuyo grado máximo es dos. La expresión canónica general de una ecuación cuadrática de una variable es:

$$ax^2 + bx + c = 0$$

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body{
    font-family: 'Tahoma';
    margin: 15px;
}

section{
    width: 400px;
    padding: 15px;
    background-color: lightblue;
    text-align:justify;
}

h1{
    text-align:center;
}

#ecuacion{
    text-align: center;
    font-family: Consolas;
    font-size: 22px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset='UTF-8'>
    <link href='estilo.css' rel='stylesheet' type='text/css'/>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>Fórmula Cuadrática</h1>
            <p>Una ecuación de segundo grado o ecuación cuadrática de una variable
            es una ecuación que tiene la forma de una suma algebraica de términos
            cuyo grado máximo es dos. La expresión canónica general de una
            ecuación cuadrática de una variable es:</p>
            <p id='ecuacion'>
                ax<sup>2</sup> + bx + c = 0
            </p>
        </section>
    </body>
</html>
```

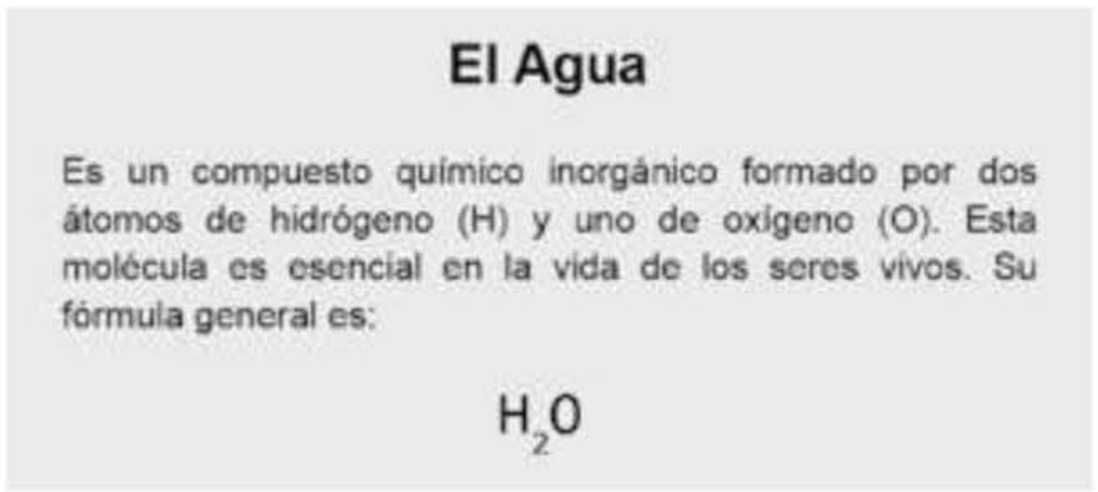
4.3.2 *La etiqueta <sub>*

Se utiliza para aplicar subíndice al texto, mayormente se utiliza para las fórmulas. Su formato es:

```
<sub>
    <!– contenido del subíndice –>
</sub>
```

Ejemplo 13

Véase el uso de la etiqueta **<sub>** dentro de un **<section>**:



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: 'Tahoma';
    margin: 15px;
}

section{
    width: 400px;
    padding: 15px;
    background-color: lightblue;
    text-align:justify;
}

h1{
    text-align:center;
}

#formula{
    text-align: center;
    font-family: Consolas;
    font-size: 22px;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
        <h1>El Agua</h1>
            Es un compuesto químico inorgánico formado por dos átomos de
            hidrógeno (H) y uno de oxígeno (O). Esta molécula es esencial en la
            vida de los seres vivos. Su fórmula general es:
            <p id='formula'>
                H<sub>2</sub>O
            </p>
        </section>
    </body>
</html>
```

4.3.3 *La etiqueta <small>*

Esta etiqueta permite que un texto sea representado con un tamaño de fuente inferior a lo normal. En HTML5 no se recomienda su uso, ya que también existe una alternativa adecuada con los estilos CSS. Su formato es:

```
<small>
   <!- contenido de texto pequeño->
</small>
```

Su equivalente en estilo CSS sería:

```
#textopequeño{
   font-size:small;
}
```

También podría representarse de la siguiente forma:

```
#textopequeño{
   font-size:smaller;
}
```

4.3.4 *La etiqueta <q>*

Es una etiqueta de citas cortas. Se puede implementar dentro de un párrafo o elemento de lista. Hay que considerar que es un elemento desarrollado para una o dos líneas. Tiene las siguientes características:

- No realiza saltos de línea, ya que es una cita de línea.
- El contenido de la cita permanece en una sola línea.
- Automáticamente se encierra el contenido de la cita entre comillas dobles para diferenciarlo de los demás textos.

Su formato es:

```
<q>
   <!- contenido de la cita corta ->
</q>
```

4.3.5 *La etiqueta <dfn>*

Permite definir un término que es descrito dentro de un párrafo. Esta definición es mostrada en letra cursiva y su uso normalmente es para términos que se usarán muchas veces dentro de un mismo documento web. Su formato es:

```
<dfn>
   <!- contenido de la definición ->
</dfn>
```

4.3.6 *La etiqueta <abbr>*

HTML5 usa mucho la semántica así es que solo faltaba una etiqueta para representar las abreviaciones que se presentan en los párrafos como el nombre de una organización, las siglas de una entidad, etc. Se recomienda usar la propiedad **Title** para ayudar a la definición de las abreviaciones. Su formato es:

```
<abbr>
   <!- contenido de las iniciales ->
</abbr>
```

4.4 Etiquetas orientadas al código

4.4.1 *La etiqueta <time>*

Permite describir el formato adecuado de la fecha u hora para que el navegador lo entienda correctamente. A veces se genera un problema únicamente con respecto a las fechas pues, en ocasiones, se trabaja con el formato corto (15/02/2015) o el formato largo (Domingo, 15 de Febrero 2015). La propiedad **datetime** de la etiqueta **<time>** permite aplicar formato a los elementos del tiempo de la mejor manera que se crea conveniente. Su formato es:

```
<time>
   <!- contenido de la fecha y hora ->
</time>
```

Véase un ejemplo básico de impresión de fecha. Para este ejercicio hay que tener en cuenta el formato de la región establecida en el sistema operativo, se empleará la fecha con el formato más usado en la web AAAA-MM-DD:

```
<time>2019-02-15</time>
```

O también podría ser especificado como fecha larga de la siguiente forma:

```
<time datetime-'2015-02-19'>
   Domingo, 15 de Febrero del 2019
</time>
```

Otros ejemplos:

```
<time datetime-'20:00'>
   8pm.
</time>
<time datetime-'2019-02-15T20:15'>
   15 de febrero del 2019 a las 8:15pm.
</time>
```

4.4.2 *La etiqueta <code>*

Esta etiqueta permite especificar que una porción de texto es un código o script de algún lenguaje de programación. Esto servirá para diferenciarlo de los demás textos. Para lograr esto, al texto se le asigna el tipo de letra monospace de forma automática. Su formato es:

```
<code>
    <!- contenido del script o código ->
</code>
```

Ejemplo 14

Véase el uso de la etiqueta **<code>** dentro de un **<section>**:

Código CSS

```
body{ font-family:"Tahoma"; margin: 15px; }
section{ width: 400px; padding: 15px; background-color:
lightblue; text-align:justify; }
h1{ text-align:center; }
```

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: 'Tahoma ';
    margin: 15px;
}

section{
    width: 400px;
    padding: 15px;
    background-color: lightblue;
    text-align:justify;
}

h3{
    text-align:center;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
```

```
<body>
    <section>
        <h3>Código CSS</h3>
        <code>
            body{
                font-family: 'Tahoma';
                margin: 15px;
            }<br>
            section{
                width: 400px;
                padding: 15px;
                background-color: lightblue;
                text-align:justify;
            }<br>
            h1{
                text-align:center;
            }
        </code>
    </section>
</body>
</html>
```

4.4.3 *La etiqueta <var>*

Permite representar de una manera particular las variables usadas en un documento web. Dicha forma particular empleará el estilo de letra cursiva. Su implementación se utiliza particularmente cuando se integra código HTML5 con algún lenguaje de programación como Java, PHP o ASP. Su formato es:

```
<var>
    <!– especificación de variables –>
</var>
```

4.4.4 *La etiqueta <samp>*

Permite representar las variables de salida de una aplicación. Estas normalmente representan al valor resultante de una aplicación en un lenguaje de programación. Se podría decir que es el resultado que mostrará la aplicación. Su formato es:

```
<samp>
    <!– especificación de variables de salida –>
</samp>
```

4.4.5 *La etiqueta <kbd>*

Permite representar las variables de entrada de una aplicación en un documento web. Esta etiqueta definirá la forma que tendrán las variables de entrada de una aplicación. Su formato es:

```
<kbd>
    <!– especificación de valores de entrada–>
</kbd>
```

Caso desarrollado 1

Uso de la etiqueta <header> y <h1>...<h6>

Implemente el siguiente artículo dentro de un documento web usando las etiquetas **<header>** y **<h1>...<h6>**.

Cisco y AGT forman alianza estratégica global para ciudades inteligentes

El objetivo es transformar la manera en que las ciudades son administradas y aseguradas, cumpliendo así la promesa de la internet de todo.

Cisco y AGT International, proveedor global de soluciones para la ciudad, anunciaron hoy una alianza estratégica global de ciudades inteligentes (smart city global strategic alliance) que entregará la promesa de la internet de todo (ioe) a través de análisis de conexiones y dispositivos y tecnologías de nube que cambian dramáticamente la manera en que las ciudades son administradas y cuidadas.

Consideraciones:

- Cree un proyecto web llamado **Caso_Header**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```
/*Estilo para todo el documento*/
body{
    font-family:Tahoma;
    font-size:12px;
}

/*Estilo para la sección*/
section{
    padding: 10px;
    width:550px;
    border-color:#666;
```

```
    border-style:solid;
}

/*Estilo para el encabezado H1*/
h1{
    color:#09F;
}

/*Estilo para el parrafo*/
p{
    color:#666;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de Header</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h1>
                    Cisco y AGT forman alianza estratégica global para
                    ciudades inteligentes
                </h1>
                <h2>
                    El objetivo es transformar la manera en que las ciudades
                    son administradas y aseguradas, cumpliendo así la promesa
                    de la internet de todo.
                </h2>
            </header>
            <p>
                Cisco y AGT International, proveedor global de soluciones
                para la ciudad, anunciaron hoy una alianza estratégica global
                de ciudades inteligentes (smart city global strategic
                alliance) que entregará la promesa de la internet de todo
                (ioe) a través de análisis de conexiones y dispositivos y
                tecnologías de nube que cambian dramáticamente la manera en
                que las ciudades son administradas y cuidadas.
            </p>
        </section>
    </body>
</html>
```

Caso desarrollado 2
Uso de la etiqueta

Agregue el siguiente artículo dentro de un documento web usando la etiqueta **
**.

> **Video a demanda: 20 años en movimiento**
>
> Además, el segmento de broadcasting también se ha visto complejizado con estos avances tecnológicos, ya que al principio sólo existían 3 actores: proveedores de contenido (estudios de cine o canales de tv), operadores de red y el consumidor final que poseía sólo un dispositivo en el hogar. en la actualidad, encontramos mayor cantidad de actores. estos son:
>
> Proveedores de contenido.
> Integradores u ott (over the top por sus siglas en inglés), tal como Netflix.
> Operadores de red y operadores virtuales que no necesariamente tienen red propia, sino que combinan y arman una solución agregando un valor de integración o tecnológico.
> Los distintos dispositivos que puede tener hoy una familia tipo: 1 televisor, 2 notebooks, 1 tablet, 4 smartphones.
>
> **Fuente:** www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_br**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```css
/*Estilo para todo el documento*/
body{
    font-family:Tahoma;
    font-size:14px;
    text-align: justify;
}

/*Estilo para la sección*/
section{
    padding: 20px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

/*Estilo para el encabezado H1*/
h1{
    color:#09F;
```

```
    text-align: center;
}

/*Estilo para el parrafo*/
p{
    color:black;
}

#derecha{
    text-align: right;
    font-size:12px;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de BR</title>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>Video a demanda: 20 años en movimiento</h1>
            <p>
            Además, el segmento de broadcasting también se ha visto complejizado
            con estos avances tecnológicos, ya que al principio sólo existían 3
            actores: proveedores de contenido (estudios de cine o canales de tv),
            operadores de red y el consumidor final que poseía sólo un
            dispositivo en el hogar. en la actualidad, encontramos mayor cantidad
            de actores. estos son:
            <br/>
            <br/>
            Proveedores de contenido.<br/>
            Integradores u ott (over the top por sus siglas en inglés), tal como
            Netflix.<br/>
            Operadores de red y operadores virtuales que no necesariamente tienen
            red propia, sino que combinan y arman una solución agregando un valor
            de integración o tecnológico.<br/>
            Los distintos dispositivos que puede tener hoy una familia tipo: 1
            televisor, 2 notebooks, 1 tablet, 4 smartphones.
            </p>
            <p id='derecha'>Fuente: www.diarioti.com </p>
        </section>
    </body>
</html>
```

Hay que tomar en cuenta que HTML5 ignora todos los espacios en blanco sobrantes dentro de un párrafo, es decir, no se podrá dejar dos espacios después de cada palabra ni cambiar de línea, ya que el navegador no lo interpretará correctamente y, por lo tanto, mostrará párrafos juntos. Véase el siguiente script errado del caso propuesto:

```
<!DOCTYPE html>
<html>
   <head>
      <title>Uso de BR</title>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <section>
         <h1>Video a demanda: 20 años en movimiento</h1>
         <p>
         Además, el segmento de broadcasting también se ha visto complejizado
         con estos avances tecnológicos, ya que al principio sólo existían 3
         actores: proveedores de contenido (estudios de cine o canales de tv),
         operadores de red y el consumidor final que poseía sólo un
         dispositivo en el hogar. en la actualidad, encontramos mayor cantidad
         de actores. estos son:
         Proveedores de contenido.
         Integradores u ott (over the top por sus siglas en inglés), tal como
         Netflix.
         Operadores de red y operadores virtuales que no necesariamente tienen
         red propia, sino que combinan y arman una solución agregando un valor
         de integración o tecnológico.
         Los distintos dispositivos que puede tener hoy una familia tipo: 1
         televisor, 2 notebooks, 1 tablet, 4 smartphones.
         </p>
         <p id='derecha'>Fuente: www.diarioti.com </p>
      </section>
   </body>
</html>
```

El resultado del código anterior es:

Video a demanda: 20 años en movimiento

Además, el segmento de broadcasting también se ha visto complejizado con estos avances tecnológicos, ya que al principio sólo existían 3 actores: proveedores de contenido (estudios de cine o canales de tv), operadores de red y el consumidor final que poseía sólo un dispositivo en el hogar. en la actualidad, encontramos mayor cantidad de actores. estos son: Proveedores de contenido. Integradores u ott (over the top por sus siglas en inglés), tal como Netflix. Operadores de red y operadores virtuales que no necesariamente tienen red propia, sino que combinan y arman una solución agregando un valor de integración o tecnológico. Los distintos dispositivos que puede tener hoy una familia tipo: 1 televisor, 2 notebooks, 1 tablet, 4 smartphones.

Fuente: www.diarioti.com

Como se observa, a pesar de haber dejado suficiente espacio entre las líneas del mismo párrafo, el navegador no lo interpreta y muestra todos los párrafos juntos, por lo que esa presentación no corresponde al modelo del caso expuesto.

Caso desarrollado 3
Uso de la etiqueta <pre>

Implemente el siguiente artículo dentro de un documento web usando la etiqueta **<pre>**.

Video a demanda: 20 años en movimiento

Además, el segmento de broadcasting también se ha visto complejizado con estos avances tecnológicos, ya que al principio sólo existían 3 actores: proveedores de contenido (estudios de cine o canales de tv), operadores de red y el consumidor final que poseía sólo un dispositivo en el hogar. en la actualidad, encontramos mayor cantidad de actores. estos son:

- Proveedores de contenido.
- Integradores u ott (over the top por sus siglas en inglés), tal como Netflix.
- Operadores de red y operadores virtuales que no necesariamente tienen red propia, sino que combinan y arman una solución agregando un valor de integración o tecnológico.
- Los distintos dispositivos que puede tener hoy una familia tipo: 1 televisor, 2 notebooks, 1 tablet, 4 smartphones.

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Pre**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** añada el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:12px;
    margin: 20px;
}

section{
    padding: 10px;
    width:720px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}

pre{
    color:#666;
    font-size: 14px;
}

p{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de PRE</title>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
<section>
    <h1>Video a demanda: 20 años en movimiento</h1>
    <pre>
        Además, el segmento de broadcasting también se ha visto complejizado
        con estos avances tecnológicos, ya que al principio sólo existían 3
        actores: proveedores de contenido (estudios de cine o canales de TV),
        operadores de red y el consumidor final que poseía sólo un dispositivo
        en el hogar. En la actualidad, encontramos mayor cantidad de actores.
        Estos son:
        -  Proveedores de contenido.
        -  Integradores u OTT (Over The Top por sus siglas en inglés), tal
```

```
            como Netflix.
        - Operadores de red y operadores virtuales que no necesariamente
          tienen red propia, sino que combinan y arman una solución
          agregando un valor de integración o tecnológico.
        - Los distintos dispositivos que puede tener hoy una familia tipo:
          1 televisor, 2 notebooks, 1 tablet, 4 smartphones.
    </pre>
    <p>Fuente: www.diarioti.com </p>
</section>
</body>
</html>
```

Caso desarrollado 4
Uso de la etiqueta <pre> para listado

Agregue el siguiente artículo dentro de un documento web usando la etiqueta `<pre>`.

Listado de Navegadores web más usados

Autor	Fecha de la primera	Última versión
Amaya	14 de noviembre de 1996	11.3.1
Chrome Google	2 de septiembre de 2008	24.0.1312.52
Internet Explorer	27 de marzo de 2000	5.2.3
Mozilla Foundation	7 de diciembre de 1998	1.7.13
Netscape	14 de noviembre de 2000	9.0.0.6
Opera	8 de diciembre de 1996	15.0

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Pre2**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** añada el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:12px;
}
```

```
section{
    padding: 10px;
    width:350px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}

pre{
    font-family:Tahoma;
    color:#666;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <title>Uso de PRE</title>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>Listado de Navegadores web más usados</h1>
            <pre>
                <strong>Autor          Fecha de la primera       Última versión</strong>
                Amaya              14 de noviembre de 1996    11.3.1
                Chrome Google       2 de septiembre de 2008   24.0.1312.52
                Internet Explorer  27 de marzo de 2000        5.2.3
                Mozilla Foundation  7 de diciembre de 1998    1.7.13
                Netscape           14 de noviembre de 2000    9.0.0.6
                Opera               8 de diciembre de 1996    15.0
            </pre>
            <P>Fuente: www.diarioti.com </P>
        </section>
    </body>
</html>
```

Caso desarrollado 5
Uso de la etiqueta

Implemente el siguiente artículo dentro de un documento web usando la etiqueta ****.

App gratuita convierte disco duro o llave USB en nube privada

D-Link recalca que la app mydlink SharePort proporciona una alternativa segura al almacenamiento público en la nube.

Muchas personas utilizan habitualmente los servicios públicos en la nube para **almacenar, compartir o descargarse archivos en su smartphone o tablet**. Una vez subidos a la nube, los datos están disponibles en prácticamente cualquier dispositivo. Pese a que es muy útil, este servicio de almacenamiento público tiene su lado negativo: por un lado, **el almacenamiento es limitado y el espacio extra tiene coste**, y, por otro, en la mayoría de los casos, la empresa en la que se almacenan estos datos tiene derecho **a acceder a los archivos del usuario y no garantiza su disponibilidad**.

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Span**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** inserte el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:14px;
    margin: 15px;
    text-align: center;
}

section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}
```

```
p{
    text-align: justify;
    color:#666;
}

/*Estilo para los SPAN*/
#resaltado{
    font-weight:800; /*Negrita*/
    color:#000; /*Color Negro*/
    text-decoration:underline;  /*Subrayado*/
}

footer{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de Span</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>
                    App gratuita convierte cualquier disco duro o llave USB en
                    nube privada
                </h1>
                <h3>
                    D-Link recalca que la app mydlink SharePort proporciona una
                    alternativa segura al almacenamiento público en la nube.
                </h3>
            </hgroup>
            <p>
                Muchas personas utilizan habitualmente los servicios públicos en la
                nube para <span id='resaltado'>almacenar, compartir o descargarse
                archivos en su smartphone o tablet</span>. Una vez subidos a la nube,
                los datos están disponibles en prácticamente cualquier dispositivo.
                Pese a que es muy útil, este servicio de almacenamiento público tiene
                su lado negativo: por un lado,<span id='resaltado'> el almacenamiento
                es limitado y el espacio extra tiene coste</span>, y, por otro, en la
                mayoría de los casos, la empresa en la que se almacenan estos datos
```

```
            tiene derecho <span id='resaltado'>a acceder a los archivos del
            usuario y no garantiza su disponibilidad</span>.
        </p>
        <footer>Fuente: www.diarioti.com </footer>
    </section>
</body>
</html>
```

Caso desarrollado 6
Uso de la etiqueta

Agregue el siguiente artículo dentro de un documento web usando la etiqueta ****.

App gratuita convierte disco duro o llave USB en nube privada

D-Link recalca que la app mydlink SharePort proporciona una alternativa segura al almacenamiento público en la nube.

Muchas personas utilizan habitualmente los servicios públicos en la nube para *almacenar, compartir o descargarse archivos en su smartphone o tablet*. Una vez subidos a la nube, los datos están disponibles en prácticamente cualquier dispositivo. Pese a que es muy útil, este servicio de almacenamiento público tiene su lado negativo: por un lado, *el almacenamiento es limitado y el espacio extra tiene coste*, y, por otro, en la mayoría de los casos, la empresa en la que se almacenan estos datos tiene derecho *a acceder a los archivos del usuario y no garantiza su disponibilidad*.

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Em**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
    text-align: justify;
    margin: 15px;
}
```

```
section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}

p{
    font-family:Tahoma, Geneva, sans-serif;
    font-size:14px;
    color:#666;
}

/*Estilo del texto énfasis aplicando Negrita*/
#enfasis{
    font-weight:800;
}

footer{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de EM</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>
                    App gratuita convierte cualquier disco duro o llave USB en nube
                    privada
                </h1>
                <h3>
                    D-Link recalca que la app mydlink SharePort proporciona una
                    alternativa segura al almacenamiento público en la nube.
                </h3>
            </hgroup>
```

```
<p>
    Muchas personas utilizan habitualmente los servicios públicos en la
    nube para <EM ID='ENFASIS'>almacenar, compartir o descargarse
    archivos en su smartphone o tablet</EM>. Una vez subidos a la nube,
    los datos están disponibles en prácticamente cualquier dispositivo.
    Pese a que es muy útil, este servicio de almacenamiento público tiene
    su lado negativo: por un lado,<EM ID='ENFASIS'> el almacenamiento es
    limitado y el espacio extra tiene coste</EM>, y, por otro, en la
    mayoría de los casos, la empresa en la que se almacenan estos datos
    tiene derecho <EM ID='ENFASIS'>a acceder a los archivos del usuario y
    no garantiza su disponibilidad</EM>.
  </p>
  <footer>Fuente: www.diarioti.com </footer>
 </section>
</body>
</html>
```

Caso desarrollado 7
Uso de la etiqueta <blockquote> y <cite>

Implemente el siguiente artículo dentro de un documento web usando la etiqueta **<blockquote>** y **<cite>**.

IBM desarrolla chip que se autodestruye

Agencia de investigación científica militar DARPA ha encargado a IBM crear un chip CMOS que se convierte en polvo de silicio al recibir un comando específico por ondas de radio.

> "Es casi imposible detectar y recuperar artículos [electrónicos en el campo de batalla], lo que resulta en su acumulación no intencional en el medio ambiente, y el potencial uso no autorizado, y consiguiente riesgo para la propiedad intelectual y ventajas tecnológicas".

Escribe ***DARPA*** en su sitio web, al referirse al programa VARP.

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_blockquote**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** inserte el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:12px;
}

section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}

p{
    font-family:Tahoma;
    font-size:14px;
    color:#666;
}
cite{
    font-weight:bold;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de Blockquote y Cite</title>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>IBM desarrolla chip que se autodestruye</h1>
                <h3>
                    Agencia de investigación científica militar DARPA ha encargado
                    a IBM crear un chip CMOS que se convierte en polvo de silicio
                    al recibir un comando específico por ondas de radio.
                </h3>
            </hgroup>
```

```
<p>
    <blockquote>'Es casi imposible detectar y recuperar artículos
    [electrónicos en el campo de batalla], lo que resulta en su
    acumulación no intencional en el medio ambiente, y el
    potencial uso no autorizado, y consiguiente riesgo para la
    propiedad intelectual y ventajas tecnológicas'.</blockquote>
    Escribe <cite>DARPA</cite> en su sitio web, al referirse al
    programa VAPR.
</p>
<footer>Fuente: www.diarioti.com </footer>
</section>
</body>
</html>
```

Caso desarrollado 8
Uso de la etiqueta <dfn> y <abbr>

Agregue el siguiente artículo dentro de un documento web usando la etiqueta `<dfn>` y `<abbr>`.

Check Point descubre una vulnerabilidad crítica en Wikipedia.org y otros sitios "Wiki"

Una investigación realizada por el fabricante revela una vulnerabilidad en la popular plataforma web Open Source MediaWiki.

Check Point Software Technologies Ltd. **(NASDAQ: CHKP)**, ha revelado el hallazgo de una vulnerabilidad crítica en la conocida plataforma web de código abierto MediaWiki, utilizada para crear y mantener sitios "Wiki". La plataforma MediaWiki es actualmente la sexta más visitada del mundo, con más de 94 millones de visitantes únicos por mes.

Los investigadores de ***Check Point*** descubrieron que esta vulnerabilidad crítica dejó expuesto a MediaWiki (Versión 1.8 en adelante) a la ejecución remota de código **RCE**, de modo que un atacante podría obtener el control completo del servidor web. De forma inmediata tras el hallazgo, ***Check Point*** alertó a la Fundación WikiMedia acerca de esta vulnerabilidad y, tras su verificación, la propia Fundación publicó una actualización y parches para el software MediaWiki.

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Dfn_Abbr**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** añada el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:12px;
}

section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}

p{
    font-family:Tahoma;
    font-size:14px;
    color:#666;
}

dfn,abbr{
    font-weight:bold;
}

footer{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de DFN-ABBR</title>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>
                    Check Point descubre una vulnerabilidad crítica en Wikipedia.org
                    y otros sitios 'Wiki'
                </h1>
```

```
        <h3>
            Una investigación realizada por el fabricante revela una
            vulnerabilidad en la popular plataforma web Open Source MediaWiki.
        </h3>
    </hgroup>
    <p>
        <dfn>Check Point Software Technologies Ltd.</dfn> <abbr
        title='Proveedor de soluciones de seguridad para Internet'>(NASDAQ:
        CHKP)</abbr>, ha revelado el hallazgo de una vulnerabilidad crítica
        en la conocida plataforma web de código abierto MediaWiki, utilizada
        para crear y mantener sitios 'Wiki'. La plataforma MediaWiki es
        actualmente la sexta más visitada del mundo, con más de 94 millones
        de visitantes únicos por mes.
    </p>
    <p>
        Los investigadores de <dfn>Check Point</dfn> descubrieron que esta
        vulnerabilidad crítica dejó expuesto a MediaWiki (Versión 1.8 en
        adelante) a la ejecución remota de código <abbr>RCE</abbr>, de modo
        que un atacante podría obtener el control completo del servidor web.
        De forma inmediata tras el hallazgo, <dfn>Check Point</dfn> alertó a
        la Fundación WikiMedia acerca de esta vulnerabilidad y, tras su
        verificación, la propia Fundación publicó una actualización y parches
        para el software MediaWiki.
    </p>
    <footer>Fuente: www.diarioti.com </footer>
  </section>
 </body>
</html>
```

Caso desarrollado 9
Uso de la etiqueta <time>

Implemente el siguiente artículo dentro de un documento web usando la etiqueta **<time>**.

Facebook cumple 10 años, ¿estará vigente en 2024?

Con motivo del 10º aniversario del lanzamiento de Facebook el 4 de febrero, el catedrático de sistemas de información de Warwick Business School, Dr. Markos Zachariadis, presenta sus pronósticos para la mayor red social del mundo.

En palabras del *Dr. Markos Zachariadis*, Facebook lleva un largo camino recorrido desde sus inicios, y actualmente es mucho más que "un sitio para subir fotos". Sin embargo, aunque Facebook se ha convertido en la mayor red social del mundo, muchos cuestionan su futuro a largo plazo. ¿Seguirá existiendo, con igual relevancia que ahora, en diez años más?.

> Según el experto, Facebook está bien posicionada para expandirse y seguir creciendo, creando un "efecto de red" y añadiendo nuevos servicios que mantengan el interés de los usuarios. La reciente inversión de Facebook en aplicaciones móviles, que incluyen la nueva aplicación 'Paper' en Estados Unidos y la adquisición de Instagram, sumada a sus esfuerzos por entender mejor, e intensificar su negocio de publicidad móvil, puede asegurar su permanencia durante otros 10 años, e incluso más. A mi juicio, la empresa va viento en popa. Por ejemplo, los pronósticos para el acceso de la población mundial a Internet desde dispositivos móviles es de impresionantes 4 mil millones para 2017.

Fecha de reproducción: 14/02/14 4:06pm

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Time**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
}

section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}
```

```
h1{
    color:#09F;
}

h3{
    text-align: justify;
}

p{
    font-family:Tahoma, Geneva, sans-serif;
    text-align: justify;
    font-size:14px;
    color:#666;
}

blockquote{
    font-family:Tahoma;
    text-align: justify;
    font-size:14px;
    color:#000;
}

footer>p{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de Time</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>Facebook cumple 10 años, ¿estará vigente en 2024?</h1>
                <h3>
                    Con motivo del 10º aniversario del lanzamiento de Facebook el 4
                    de febrero, el catedrático de sistemas de información de Warwick
                    Business School, Dr. Markos Zachariadis, presenta sus pronósticos
                    para la mayor red social del mundo.
                </h3>
            </hgroup>
```

```
<p>
    En palabras del <cite>Dr. Markos Zachariadis</cite>, Facebook lleva
    un largo camino recorrido desde sus inicios, y actualmente es mucho
    más que <q>un sitio para subir fotos</q>. Sin embargo, aunque
    Facebook se ha convertido en la mayor red social del mundo, muchos
    cuestionan su futuro a largo plazo. ¿Seguirá existiendo, con igual
    relevancia que ahora, en diez años más?.
    <blockquote>Según el experto, Facebook está bien posicionada para
    expandirse y seguir creciendo, creando un 'efecto de red' y añadiendo
    nuevos servicios que mantengan el interés de los usuarios. La
    reciente inversión de Facebook en aplicaciones móviles, que incluyen
    la nueva aplicación 'Paper' en Estados Unidos y la adquisición de
    Instagram, sumada a sus esfuerzos por entender mejor, e intensificar
    su negocio de publicidad móvil, puede asegurar su permanencia durante
    otros 10 años, e incluso más. A mi juicio, la empresa va viento en
    popa. Por ejemplo, los pronósticos para el acceso de la población
    mundial a Internet desde dispositivos móviles es de impresionantes 4
    mil millones para 2017. </blockquote>
</p>
<footer>
    <p>Fecha de reproducción:
        <TIME DATETIME='2014-02-14T16:06'>14/02/14 4:06pm</TIME>
    </p>
    <p>Fuente: www.diarioti.com</p>
</footer>
</section>
</body>
</html>
```

Caso desarrollado 10
Uso de la etiqueta <code>

Agregue el siguiente artículo dentro de un documento web usando la etiqueta `<code>`.

Nuevo engaño en Facebook promete hackear las cuentas de otros usuarios

Los ciberdelincuentes están celebrando, a su manera, el décimo aniversario de Facebook. Sophos advierte sobre un nuevo engaño.

En el décimo aniversario de la red social, los engaños siguen apareciendo regularmente, apelando a la curiosidad de los usuarios para realizar acciones maliciosas. En este caso se le promete al usuario la posibilidad de acceder de forma ilícita a la cuenta de otro usuario, mostrando el siguiente mensaje: "¿Quieren "hackear" – tener el password de la cuenta de alguien en Facebook? Aprovechen esta vulnerabilidad antes de que se den cuenta y la quiten, está funcionando AHORITA!

```
. Necesitan utilizar Google Chrome
. Se meten al perfil de la persona
. Aprietan la tecla F12
. Van a la seccion/pestaña que dice "Consola"
. Copian y pegan el siguiente código , presionan y listo!!!
. Se abrirá un pop-up con el password de esa persona!!!"
```

Fuente: www.diarioti.com

Consideraciones:

- Cree un proyecto web llamado **Caso_Code**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
}

section{
    padding: 10px;
    width:550px;
    border-color:#666;
    border-style:solid;
}

h1{
    color:#09F;
}
```

```
p{
    font-family:Tahoma, Geneva, sans-serif;
    font-size:14px;
    color:#666;
}

footer{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <hgroup>
                <h1>
                    Nuevo engaño en Facebook promete hackear las cuentas de
                    otros usuarios
                </h1>
                <h3>
                    Los ciberdelincuentes están celebrando, a su manera, el décimo
                    aniversario de Facebook. Sophos advierte sobre un nuevo engaño.
                </h3>
            </hgroup>
            <p>
                En el décimo aniversario de la red social, los engaños siguen
                apareciendo regularmente, apelando a la curiosidad de los usuarios
                para realizar acciones maliciosas. En este caso se le promete al
                usuario la posibilidad de acceder de forma ilícita a la cuenta de
                otro usuario, mostrando el siguiente mensaje:
                '¿Quieren 'hackear' - tener el password de la cuenta de alguien en
                Facebook? Aprovechen esta vulnerabilidad antes de que se den cuenta y
                la quiten, está funcionando AHORITA!
            </p>
            <pre>
                <CODE>
                    . Necesitan utilizar Google Chrome
                    . Se meten al perfil de la persona
                    . Aprietan la tecla F12
                    . Van a la seccion/pestaña que dice 'Consola'
```

```
                . Copian y pegan el siguiente código , presionan y listo!!!
                . Se abrirá un pop-up con el password de esa persona!!!'
            </CODE>
        </pre>
        <footer>
            Fuente: www.diarioti.com
        </footer>
    </section>
</body>
</html>
```

Caso desarrollado 11
Uso de la etiqueta <var>, <samp> y <kbd>

Implemente el siguiente artículo dentro de un documento web empleando las etiquetas **<var>**, **<samp>** y **<kbd>**.

Algoritmo Java

Diseñe un algoritmo que determine la clasificación de un ángulo dado en grados, minutos y segundos. Asuma que el ángulo está en el intervalo de 0º a 360º.

```
public void actionPerformed( ActionEvent e ){
 if( e.getSource() == btnProcesar ){
    int grados, minutos, segundos;   double beta;   String tipo;

  grados=Integer.parseInt(txtGrados.getText());
  minutos=Integer.parseInt(txtMinutos.getText());
  segundos=Integer.parseInt(txtSegundos.getText());

 beta = grados + minutos/60.0 + segundos/3600.0;
 if( beta == 0 )
 tipo = "Nulo";
  else if( beta < 90 )
   tipo = "Agudo";
  else if( beta == 90 )
  tipo = "Recto";
  else if( beta < 180 )
  tipo = "Obtuso";
  else if( beta == 180 )
  tipo = "Llano";
  else if( beta < 360 )
  tipo = "Cóncavo";
  else
  tipo = "Completo";

  txtS.setText("El ángulo se clasifica como:"+tipo);
}
```

Código Java

Consideraciones:

- Cree un proyecto web llamado **Caso_var_samp_kbd**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** coloque el siguiente código:

```
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
}

section{
    width:500px;
    border-color:#666;
    border-style:solid;
}

h1{
    padding: 10px;
    color:#09F;
}

p{
    padding: 0 0 0 10px;
    font-family:Tahoma, Geneva, sans-serif;
    font-size:14px;
    color:#666;
}

#codigo{
    color:#000;
}

footer{
    text-align: right;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Uso de Var</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
```

```
<body>
   <section>
      <h1>Algoritmo Java</h1>
      <p>
         Diseñe un algoritmo que determine la clasificación de un ángulo dado
         en grados, minutos y segundos. Asuma que el ángulo está en el
         intervalo de 0º a 360º.
      </p>
      <p id='codigo'>
         public void actionPerformed( ActionEvent e ){<br/>
          if( e.getSource() == btnProcesar ){ <br/>
           
         <VAR>
              int grados, minutos, segundos;
              double beta;
              String tipo;<br/><br/>
         </VAR>
         <KBD>
              grados=Integer.parseInt(txtGrados.getText());<br/>
              minutos=Integer.parseInt(txtMinutos.getText());<br/>
              segundos=Integer.parseInt(txtSegundos.getText());
            <br/><br/>
         </KBD>
           beta = grados + minutos/60.0 + segundos/3600.0;<br/>
           if( beta == 0 )<br/>
           tipo = 'Nulo';<br/>
            else if( beta < 90 )<br/>
             tipo = 'Agudo';<br/>
            else if( beta == 90 )<br/>
            tipo = 'Recto';<br/>
            else if( beta < 180 )<br/>
            tipo = 'Obtuso';<br/>
            else if( beta == 180 )<br/>
            tipo = 'Llano';<br/>
            else if( beta < 360 )<br/>
            tipo = 'Cóncavo';<br/>
            else<br/>
            tipo = 'Completo';<br/><br/>
         <SAMP>
              txtS.setText('El ángulo se clasifica como:'+tipo);
            <br/>
         </SAMP>
         }
      </p>
```

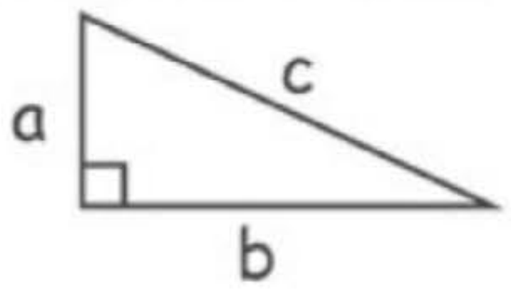
```
            <footer>
                <p>Código Java</p>
            </footer>
        </section>
    </body>
</html>
```

Caso desarrollado 12
Uso de la etiqueta <sup>

Agregue el siguiente artículo dentro de un documento web usando la etiqueta **<sup>**.

Teorema de Pitágoras

El teorema de Pitágoras establece que en todo triángulo rectángulo, el cuadrado de la hipotenusa (el lado de mayor longitud del triángulo rectángulo) es igual a la suma de los cuadrados de los catetos (los dos lados menores del triángulo, los que conforman el ángulo recto).



Cada uno de los sumandos, representa el área de un cuadrado de lado, a, b, c. Con lo que la expresión anterior, en términos de áreas se expresa en la forma siguiente: $a^2+b^2=c^2$

Consideraciones:

- Cree un proyecto web llamado **Caso_sup**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** inserte el siguiente código:

```
/*Estilo para todo el documento*/
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
}

/*Estilo para la sección*/
section{
    padding: 15px;
    width:500px;
    height:250px;
```

```
    border-color:#666;
    border-style:solid;
    text-align: justify;
}

/*Estilo para el encabezado H1*/
h1{
    color:#09F;
}

/*Estilo para el párrafo*/
p{
    color:#666;
}

/*Estilo para las columnas*/
#columna {
    padding:0 30px 0 20px;
    float:left;
    width: 40%;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h1> Teorema de Pitágoras </h1>
            </header>
            <p>
                El teorema de Pitágoras establece que en todo triángulo rectángulo,
                el cuadrado de la hipotenusa (el lado de mayor longitud del triángulo
                rectángulo) es igual a la suma de los cuadrados de los catetos (los
                dos lados menores del triángulo, los que conforman el ángulo recto).
            </p>
            <article id='columna'>
                <figure>
                    <img src='img/triangulo.jpg'>
                </figure>
            </article>
```

```
        <article id='columna'>
            <p>
            Cada uno de los sumandos, representa el área de un cuadrado de
            lado, a, b, c. Con lo que la expresión anterior, en términos de
            áreas se expresa en la forma siguiente:
            a<sup>2</sup>+b<sup>2</sup>-c<sup>2</sup>
            </p>
        </article>
    </section>
  </body>
</html>
```

Caso desarrollado 13
Uso de la etiqueta <mark>

Implemente el siguiente artículo dentro de un documento web usando la etiqueta <mark>.

Identificadas las expectativas de los "estilos de vida conectados"

Ericsson ConsumerLab identifica criterios de éxito clave para que los servicios del mañana respondan adecuadamente a las expectativas.

El mercado de los servicios digitales está ofreciendo continuamente nuevas aplicaciones y soluciones que permiten a los usuarios comunicarse, socializar, encontrar información y realizar compras en línea. Ericsson ConsumerLab ha evaluado estas crecientes necesidades de los consumidores, así como una serie de conceptos de servicios, diseñados para satisfacerlas. Al entrevistar a los usuarios de teléfonos inteligentes en tres mercados: Brasil, Indonesia y Estados Unidos, ConsumerLab ha identificado siete necesidades fundamentales, que son cruciales para el éxito de nuevos servicios. Estas son:

- Acceso inmediato en todas partes: en función a su estilo de vida "sobre la marcha", los usuarios esperan tener acceso a una gama de servicios bajo demanda y en tiempo real.
- Personalizar mi mundo: las personas están exigiendo cada vez más, alertas y servicios personalizados que se adapten a su estilo de vida.
- Desconexión: los consumidores utilizarán la tecnología para equilibrar la mentalidad de "siempre en línea", al activamente tomar tiempo para "desconectarse", relajarse y descansar. Los consumidores expresaron su interés en los servicios que les ayuden a hacer esto.
- Asistencia a la productividad: la gente está interesada en los servicios que gestionan y coordinan sus agendas.
- Mejorar la Convivencia: con vidas muy ocupadas, amigos y familia constantemente confiarán en la tecnología para mantenerse en contacto, sin importar el lugar en el cual se encuentren.
- Control a los consumidores: la gente quiere mantener control sobre sus gastos, los datos empleados y las actualizaciones de información, lo que les permite dedicar más tiempo a las cosas importantes de la vida.
- Ubicación basada en mi: los consumidores están expresando la necesidad de obtener información relevante en relación a la ubicación geográfica en donde se encuentran, y quieren que dicha información esté disponible en cualquier momento.

ConsumerLab probó varios conceptos de nuevos servicios en – grupos focales en todos los mercados estudiados. Un común denominador sobre estos conceptos es que ayudan a los consumidores a reducir la complejidad, ahorrar tiempo y disminuir el estrés a través de la gestión centralizada, proporcionando un mejor control del gasto y el uso.

Consideraciones:

- Cree un proyecto web llamado **Caso_Mark**.
- Cree una carpeta con el nombre **css** dentro del proyecto.
- Dentro de la carpeta **css**, agregue un archivo de tipo hoja de estilo y nómbrelo **estilo**.

Solución:

1. En el archivo **estilo.css** agregue el siguiente código:

```
/*Estilo para todo el documento*/
body{
    font-family:Verdana, Geneva, sans-serif;
    font-size:12px;
}

/*Estilo para la sección*/
section{
    padding: 10px;
    width: 500px;
    border-color: #666;
    border-style: solid;
}

/*Estilo para el encabezado H1*/
h1{
    color:#09F;
}

/*Estilo para el párrafo*/
p{
    color:#666;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <HEADER>
                <hgroup>
                    <H1>
                        Identificadas las expectativas de los 'estilos de vida conectados'
                    </H1>
```

```
            <H3>
                Ericsson ConsumerLab identifica criterios de éxito clave para
                que los servicios del mañana respondan adecuadamente a las
                expectativas.
            </H3>
        </hgroup>
    </HEADER>
    <p>
        El mercado de los servicios digitales está ofreciendo continuamente
        nuevas aplicaciones y soluciones que permiten a los usuarios
        comunicarse, socializar, encontrar información y realizar compras en
        línea. Ericsson ConsumerLab ha evaluado estas crecientes necesidades
        de los consumidores, así como una serie de conceptos de servicios,
        diseñados para satisfacerlas. Al entrevistar a los usuarios de
        teléfonos inteligentes en tres mercados: Brasil, Indonesia y Estados
        Unidos, ConsumerLab ha identificado siete necesidades fundamentales,
        que son cruciales para el éxito de nuevos servicios. Estas son:
    </p>
    <ul>
        <li>
            <MARK>Acceso inmediato en todas partes:</MARK> en función a su
            estilo de vida 'sobre la marcha', los usuarios esperan tener
            acceso a una gama de servicios bajo demanda y en tiempo real.
        </li>
        <li>
            <MARK>Personalizar mi mundo: </MARK>las personas están exigiendo
            cada vez más, alertas y servicios personalizados que se adapten a
            su estilo de vida.
        </li>
        <li>
            <MARK>Desconexión:</MARK> los consumidores utilizarán la
            tecnología para equilibrar la mentalidad de 'siempre en línea',
            al activamente tomar tiempo para 'desconectarse', relajarse y
            descansar. Los consumidores expresaron su interés en los
            servicios que les ayuden a hacer esto.
        </li>
        <li>
            <MARK>Asistencia a la productividad:</MARK> la gente está
            interesada en los servicios que gestionan y coordinan sus
            agendas.
        </li>
        <li>
            <MARK>Mejorar la Convivencia: </MARK>con vidas muy ocupadas,
            amigos y familia constantemente confiarán en la tecnología para
            mantenerse en contacto, sin importar el lugar en el cual se
            encuentren.
        </li>
```

```
            <li>
                <MARK>Control a los consumidores: </MARK>la gente quiere mantener
                control sobre sus gastos, los datos empleados y las
                actualizaciones de información, lo que les permite dedicar más
                tiempo a las cosas importantes de la vida.
            </li>
            <li>
                <MARK>Ubicación basada en mi: </MARK>los consumidores están
                expresando la necesidad de obtener información relevante en
                relación a la ubicación geográfica en donde se encuentran, y
                quieren que dicha información esté disponible en cualquier
                momento.
            </li>
        </ul>
        <p>
            ConsumerLab probó varios conceptos de nuevos servicios en - grupos
            focales en todos los mercados estudiados. Un común denominador sobre
            estos conceptos es que ayudan a los consumidores a reducir la
            complejidad, ahorrar tiempo y disminuir el estrés a través de la
            gestión centralizada, proporcionando un mejor control del gasto y el
            uso.
        </p>
    </section>
</body>
</html>
```

Usando caracteres especiales

A lo largo de este cuarto capítulo se han usado algunos caracteres especiales para incorporar algunos símbolos especiales como, por ejemplo, los espacios en blanco o la tabulación de espacios dentro de un párrafo. Por tal motivo, a continuación se hará una lista de los códigos que se deben incorporar en HTML5 para mostrar un carácter especial:

Código para caracteres especiales:

CARÁCTER	CÓDIGO
<	< (menor que)
>	> (mayor que)
'	' (apóstrofo)
"	" (comillas)
&	& (ampersand)
$	$$ (dólar)
	(espacio)

Código para letras:

LETRA	CÓDIGO	LETRA	CÓDIGO	LETRA	CÓDIGO	LETRA	CÓDIGO	LETRA	CÓDIGO
A	A	a	a	À	À	Û	Û	õ	õ
B	B	b	b	Â	Â	Ü	Ü	ö	ö
C	C	c	c	Ã	Ã	Ý	Ý	÷	÷
D	D	d	d	Ä	Ä	Þ	Þ	ø	ø
E	E	e	e	Å	Å	ß	ß	ù	ù
F	F	f	f	Æ	Æ	à	à	ú	ú
G	G	g	g	Ç	Ç	á	á	û	û
H	H	h	h	È	È	â	â	ü	ü
I	I	i	i	É	É	ã	ã	ý	ý
J	J	j	j	Ê	Ê	ä	ä	þ	þ
K	K	k	k	Ë	Ë	å	å	ÿ	ÿ
L	L	l	l	Ì	Ì	æ	æ	M	M
m	m	Í	Í	ç	ç	N	N	n	n
Î	Î	è	è	O	O	o	o	Ï	Ï
é	é	P	P	p	p	Đ	Ð	ê	ê
Q	Q	q	q	Ñ	Ñ	ë	ë	R	R
r	r	Ò	Ò	ì	ì	S	S	s	s
Ó	Ó	í	í	T	T	t	t	Ô	Ô
î	î	U	U	u	u	Õ	Õ	ï	ï
V	V	v	v	Ö	Ö	ð	ð	W	W
w	w	×	×	ñ	ñ	X	X	y	x
Ø	Ø	ò	ò	Y	Y	y	y	Ù	Ù
ó	ó	Z	Z	z	z	Ú	Ú	ô	ô

Código para números:

NÚMERO	CÓDIGO
0	0
1	1
2	2
3	3
4	4
5	5
6	6

NÚMERO	CÓDIGO
7	7
8	8
9	9
º	º
¹	¹
²	²
³	³

Código para signos:

SIGNO	CÓDIGO	SIGNO	CÓDIGO	SIGNO	CÓDIGO
¿	¿	&	&	¤	¤
?	?	'	'	¥	¥
¡	¡	*	*	¦	¦
!	!	+	+	§	§
(	(	=	=	¨	¨
)	)	,	,	©	©
[	[	-	-	ª	ª
]	]	.	.	¬	¬
"	«	:	:		
"	»	;	;	®	®
{	{	@	@	¯	¯
}	}	^	^	°	°
>	>	_	_	±	±
<	<	`	`	´	´
/	/	~	~	µ	µ
\	\		&#127;	¶	¶
"	"				·	·
#	#	¢	¢	¸	¸
%	%	£	£		

Finalmente, las tildes y la letra ñ

SIGNO	ESCRITURA
á	á
Á	Á
é	é
É	É
í	í
Í	Í
ó	ó
Ó	Ó
ú	ú
Ú	Ú
ñ	ñ
Ñ	Ñ

CAPÍTULO 5

Listas



Competencias

- Implementa listas desordenadas en un documento web.
- Elabora listas ordenadas como una herramienta de diseño profesional.

Contenido

5.1 Trabajando con listas
5.2 La etiqueta <ul> - lista desordenada
5.3 La etiqueta <ol> - lista ordenada
- Caso desarrollado 1: Listas desordenadas
- Caso desarrollado 2: Listas desordenadas con viñeta cuadrada
- Caso desarrollado 3: Listas desordenadas con viñeta circular hueca
- Caso desarrollado 4: Listas desordenadas sin viñeta
- Caso desarrollado 5: Listas desordenadas con imagen
- Caso desarrollado 6: Listas ordenadas con numeración decimal
- Caso desarrollado 7: Listas ordenadas con numeración romana
- Caso desarrollado 8: Listas ordenadas con numeración romana descendente

5.1 Trabajando con listas

Las listas llegan a ser una parte importante de los documentos web. HTML5 propone la etiqueta `<nav>` como menú de enlaces, esta etiqueta incluye una lista de opciones que emplean a su vez etiquetas de listas en HTML5. Es importante aclarar que las etiquetas de listas también pueden mostrar listas comunes que no necesariamente sean enlaces web.

¿Qué se puede hacer dentro de las listas?

- Organizar varios elementos en una lista desordenada.
- Organizar varios elementos en una lista ordenada.
- Distinguir listas ordenadas y desordenadas.
- Describir las similitudes entre listas ordenadas y desordenadas.
- Crear un hipervínculo a una sección dentro de su documento.
- Enlazar el documento al contenido externo.
- Resumir el propósito de los hipervínculos (etiquetas delimitadoras).
- Crear un diagrama que represente documentos con hipervínculos.

A continuación se presentan algunos modelos de listas:

- Lista de enlaces

  - Volutpat arcu nisl imperdiet mceleeli
  - Cuisque ac risus congue tpis mollis

- Lista sin viñetas

Tel.:

1-800-113-0088

1-800-233-0088

Address:

Cum sociis natoque penatt

magnis dis parturient.

E-mail:

- Lista de navegación web

Home Page | About Us | Services | Our Lines | Contact Us

Los tipos de listas para HTML5 son:

- Listas desordenadas (*unordered*)
- Listas ordenadas (*ordered*)
- Listas definidas (*definition*)

5.2 La etiqueta <ul> - lista desordenada

Se le puede llamar también "lista común", pues es la más usada al momento de presentar una lista que no tiene una correlación específica. Se caracteriza por presentar un símbolo o viñeta en el lado izquierdo.



Figura 4.1 Definición de UL.
Fuente: Elaboración propia.

El formato para la implementación de listas desordenadas es:

```
<ul>
   <!- elementos de la lista ->
</ul>
```

Las características más importantes de este tipo de listas son:

- Establece un nivel de anidamiento específico, es decir, puede contener una lista dentro de otra.
- Los atributos **type**, **start** y **compact** de la etiqueta **<ul>** han quedado obsoletas para el HTML5, se recomienda el uso de CSS para la implementación de las mismas.

La etiqueta **<ul>** define listas, pero no contenidos; por lo tanto, se le debe dar más especificaciones. Para realizar un listado no ordenado se debe usar la etiqueta **<li>** para cada elemento de la lista. Su formato es:

```
<ul>
   <li> opcion 1 </li>
   <li> opcion 2 </li>
   <li> opcion 3 </li>
</ul>
```

Ejemplo

Véase el uso de la etiqueta **<ul>** dentro de un **<section>**:

- Fútbol
- Vóley
- Básquet

En el archivo **estilo.css** se agrega el siguiente código:

```
body{
    font-family: Tahoma;
}

section{
    width: 200px;
    background-color: aliceblue;
    padding: 5px;
    margin: 10px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <ul>
                <li>Fútbol</li>
                <li>Vóley</li>
                <li>Básquet</li>
            </ul>
        </section>
    </body>
</html>
```

Si se añade el siguiente código al archivo **estilo.css**:

```
ul{
    list-style: disc;
}
```

Se obtendrá un diseño igual, es decir, no varía; esto significa que el diseño predeterminado de la viñeta es en forma de disco.

Ahora bien, si se necesita que el diseño muestre viñetas circulares huecas como la siguiente imagen:



El script del archivo **estilo.css** debe modificarse de la siguiente manera:

```
ul{
    list-style: circle;
}
```

Si se desea que el diseño muestre viñetas cuadradas como en la siguiente imagen:



El script del archivo **estilo.css** debe cambiarse de la siguiente manera:

```
ul{
    list-style: square;
}
```

Si se necesita ocultar las viñetas, pero mantener la forma de la lista como se observa en la siguiente imagen:



El script del archivo **estilo.css** debe modificarse de la siguiente manera:

```
ul{
    list-style: none;
}
```

5.3 La etiqueta <ol> - lista ordenada

Las listas ordenadas presentan su viñeta como números (1, 2, 3...), letras (a, b, c...), números romanos (I, II, III...) y todo lo que pueda representar una correlación de valores. Dichos valores se generarán automáticamente según el número de los elementos de la lista.

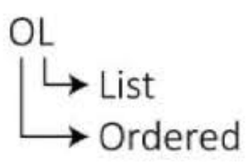


Figura 4.2 Definición de OL.
Fuente: Elaboración propia.

Su formato es:

```
<ol>
    <!- elementos de la lista ->
</ol>
```

Ejemplo

Véase el uso de la etiqueta **<ol>** dentro de un **<section>**:

1. Fútbol
2. Vóley
3. Básquet

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: Tahoma;
}

section{
    width: 200px;
    background-color: aliceblue;
    padding: 5px;
    margin: 10px;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <ol>
                <li>Fútbol</li>
                <li>Vóley</li>
                <li>Básquet</li>
            </ol>
        </section>
    </body>
</html>
```

Si se añade la siguiente definición al archivo **estilo.css**:

```
ol{
   list-style-type: decimal;
}
```

Se conseguirá el mismo diseño; esto quiere decir que el diseño predeterminado de la viñeta es la forma con números.

Ahora bien, si se necesita que el diseño muestre números con ceros a la izquierda como en la siguiente imagen:

01. Fútbol
02. Vóley
03. Básquet

El script del archivo **estilo.css** debe modificarse de la siguiente manera:

```
ol{
   list-style-type: decimal-leading-zero;
}
```

Si se quiere que el diseño muestre el abecedario en minúsculas como en la siguiente imagen:

a. Fútbol
b. Vóley
c. Básquet

El script del archivo **estilo.css** debe modificarse de la siguiente manera:

```
ol{
   list-style-type: lower-alpha;
}
```

Si se desea que el diseño muestre números romanos en minúscula como en la siguiente imagen:

i. Fútbol
ii. Vóley
iii. Básquet

El script del archivo **estilo.css** debe cambiarse de la siguiente manera:

```
ol{
   list-style-type: lower-roman;
}
```

Si se necesita que el diseño muestre el abecedario en mayúsculas como en la siguiente imagen:



El script del archivo **estilo.css** debe modificarse de la siguiente manera:

```
ol{
   list-style-type: upper-alpha;
}
```

Si se quiere que el diseño muestre números romanos en mayúsculas como en la siguiente imagen:



El script del archivo **estilo.css** debe cambiarse de la siguiente manera:

```
ol{
   list-style-type: upper-roman;
}
```

Si se necesita empezar la numeración con otro valor como en la siguiente imagen:



La variación debe realizarse en el archivo **index.html**.

```
<section>
   <ol start='5'>
      <li>Fútbol</li>
      <li>Vóley</li>
      <li>Básquet</li>
   </ol>
</section>
```

La variación anterior también podría implementarse de la siguiente manera:

```
<section>
   <ol>
      <li value='5'>Fútbol</li>
      <li>Vóley</li>
      <li>Básquet</li>
   </ol>
</section>
```

Si necesita que el diseño muestre los números de forma descendente como en la siguiente imagen:

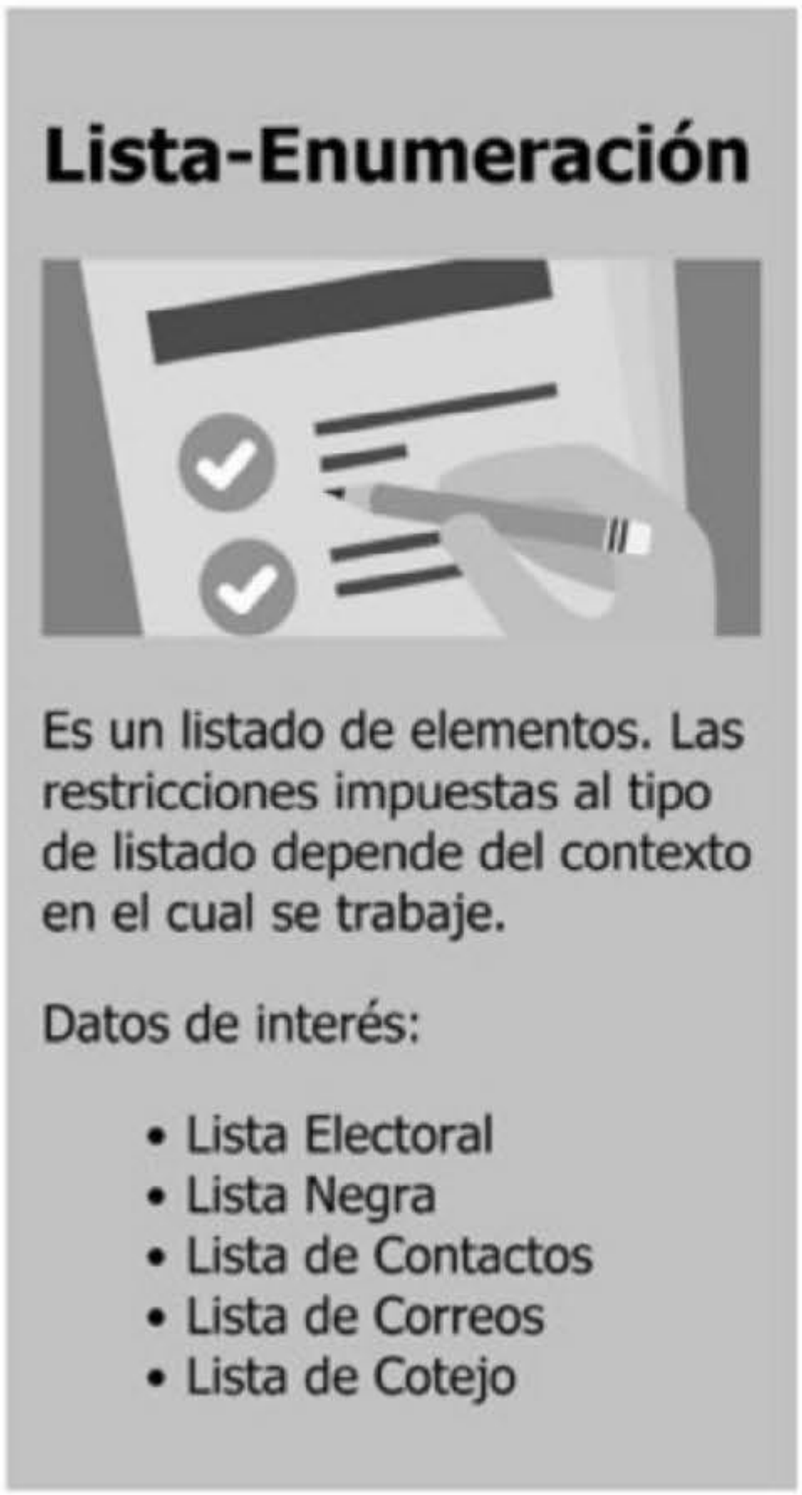
3. Fútbol
2. Vóley
1. Básquet

La modificación debe realizarse en el archivo **index.html**.

```
<section>
   <ol reversed>
      <li>Fútbol</li>
      <li>Vóley</li>
      <li>Básquet</li>
   </ol>
</section>
```

Caso desarrollado 1
Listas desordenadas

Implemente el siguiente diseño dentro de un documento web usando listas desordenadas.



Consideraciones:

- Cree un proyecto web llamado **Caso_Desordenadas**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}
```

```css
/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
            <p>
                Es un listado de elementos. Las restricciones impuestas al tipo de
                listado depende del contexto en el cual se trabaje.
            </p>
            <p>Datos de interés:</p>
            <ul>
                <li>Lista Electoral</li>
                <li>Lista Negra</li>
                <li>Lista de Contactos</li>
                <li>Lista de Correos</li>
                <li>Lista de Cotejo</li>
            </ul>
        </section>
    </body>
</html>
```

Caso desarrollado 2
Listas desordenadas con viñeta cuadrada

Agregue el siguiente diseño dentro de un documento web usando listas desordenadas con viñetas cuadradas.

Lista-Enumeración



Es un listado de elementos. Las restricciones impuestas al tipo de listado depende del contexto en el cual se trabaje.

Datos de interés

- Lista Electoral
- Lista Negra
- Lista de Contactos
- Lista de Correos
- Lista de Cotejo

Consideraciones:

- Cree un proyecto web llamado **Caso_Viñeta_Cuadrada**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}

/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}

ul{
    list-style: square;
}
```

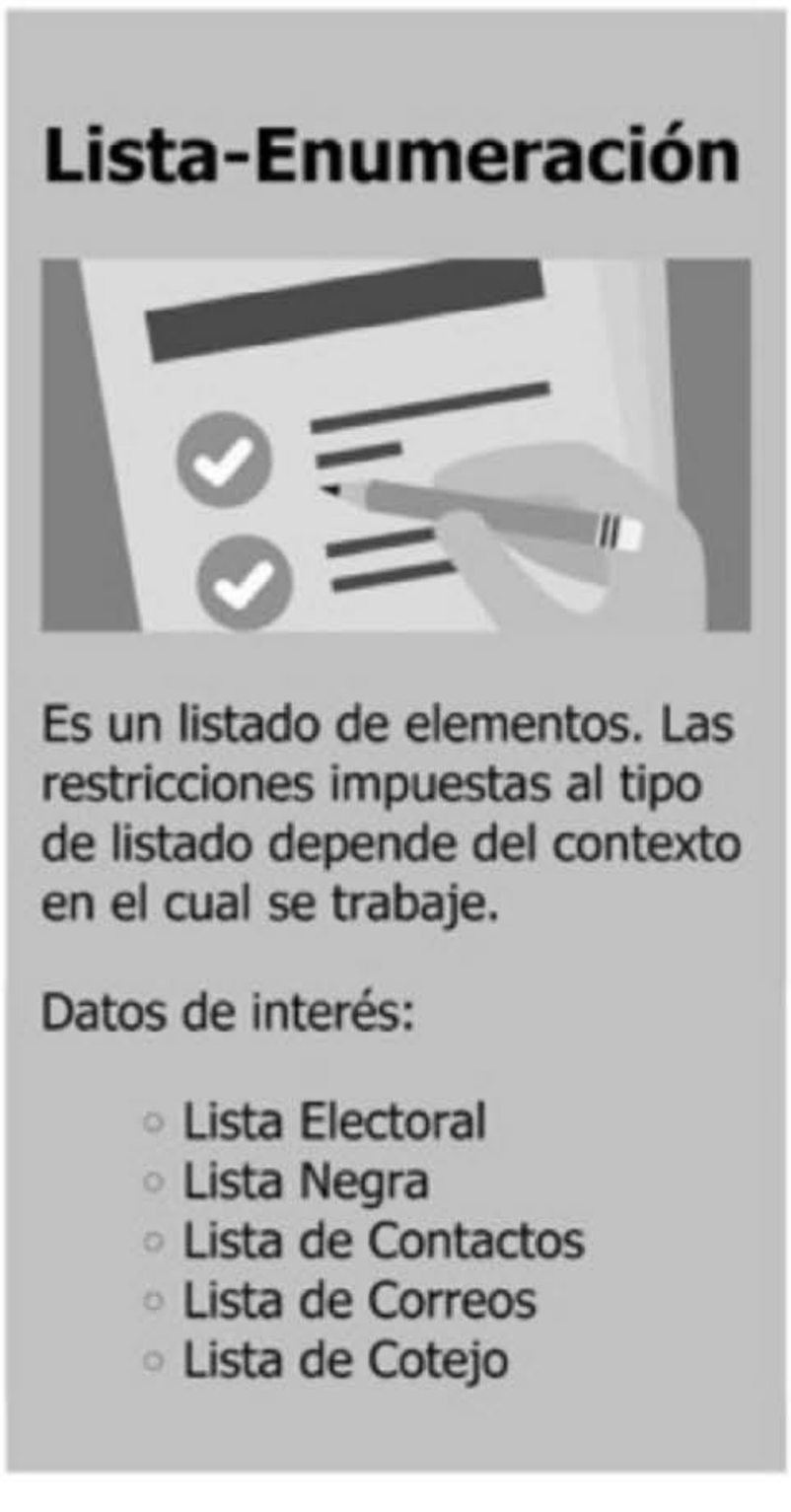
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
            <p>
                Es un listado de elementos. Las restricciones impuestas al tipo de
                listado depende del contexto en el cual se trabaje.
            </p>
            <p>Datos de interés:</p>
            <ul>
                <li>Lista Electoral</li>
                <li>Lista Negra</li>
                <li>Lista de Contactos</li>
```

```
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ul>
    </section>
</body>
</html>
```

Caso desarrollado 3
Listas desordenadas con viñeta circular hueca

Implemente el siguiente diseño dentro de un documento web usando listas desordenadas con viñetas circulares huecas.



Consideraciones:

- Cree un proyecto web llamado **Caso_Viñeta_Hueca**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}

/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}

ul{
    list-style: circle;
}
```

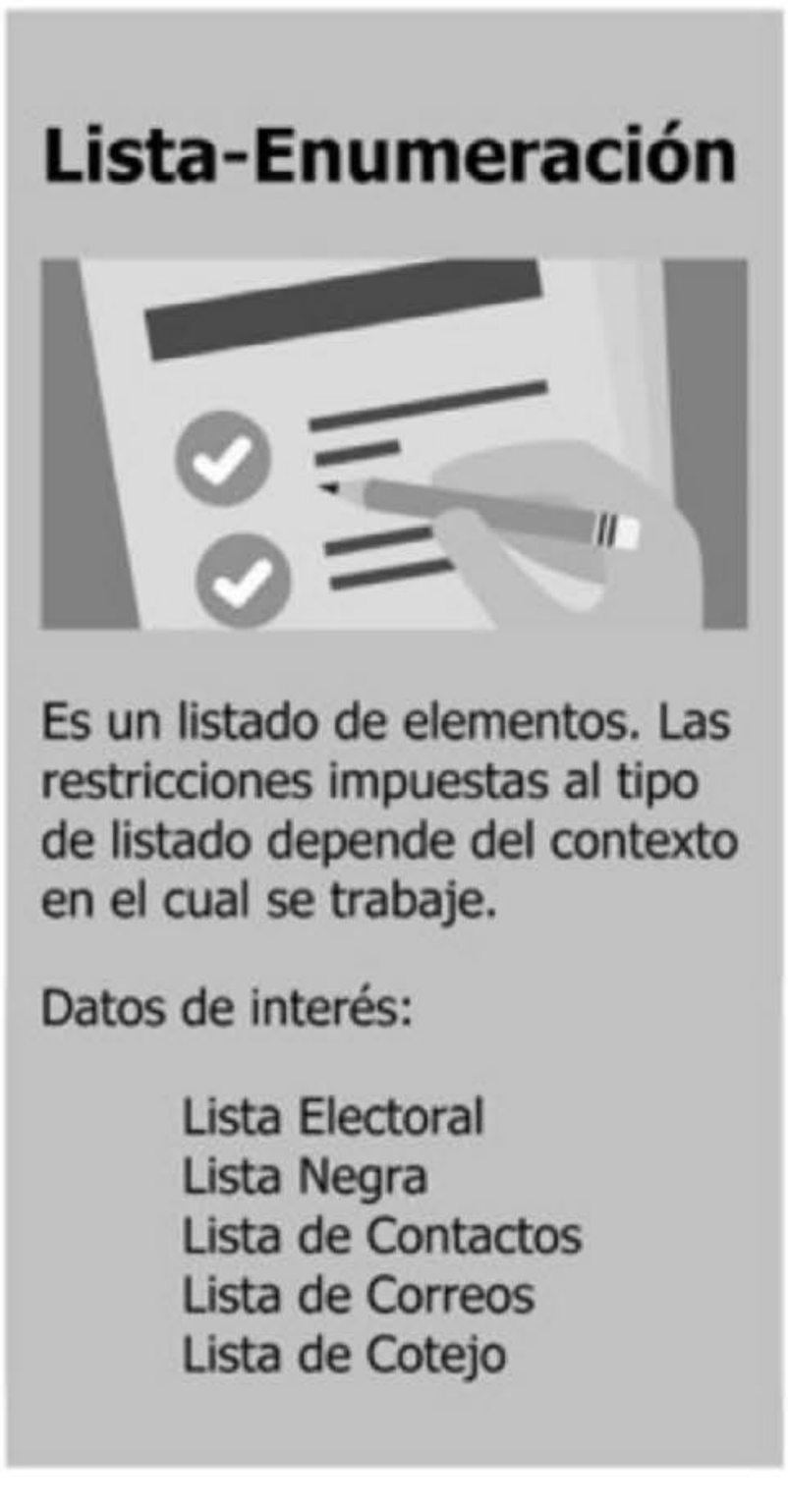
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
            <p>
                Es un listado de elementos. Las restricciones impuestas al tipo de
                listado depende del contexto en el cual se trabaje.
            </p>
            <p>Datos de interés:</p>
            <ul>
                <li>Lista Electoral</li>
                <li>Lista Negra</li>
                <li>Lista de Contactos</li>
```

```
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ul>
    </section>
</body>
</html>
```

Caso desarrollado 4
Listas desordenadas sin viñeta

Agregue el siguiente diseño dentro de un documento web usando listas desordenadas sin viñetas.



Consideraciones:

- Crear un proyecto web llamado **Caso_Sin_Viñeta**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}

/*Estilo para la seccion*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}

ul{
    list-style: none;
}
```

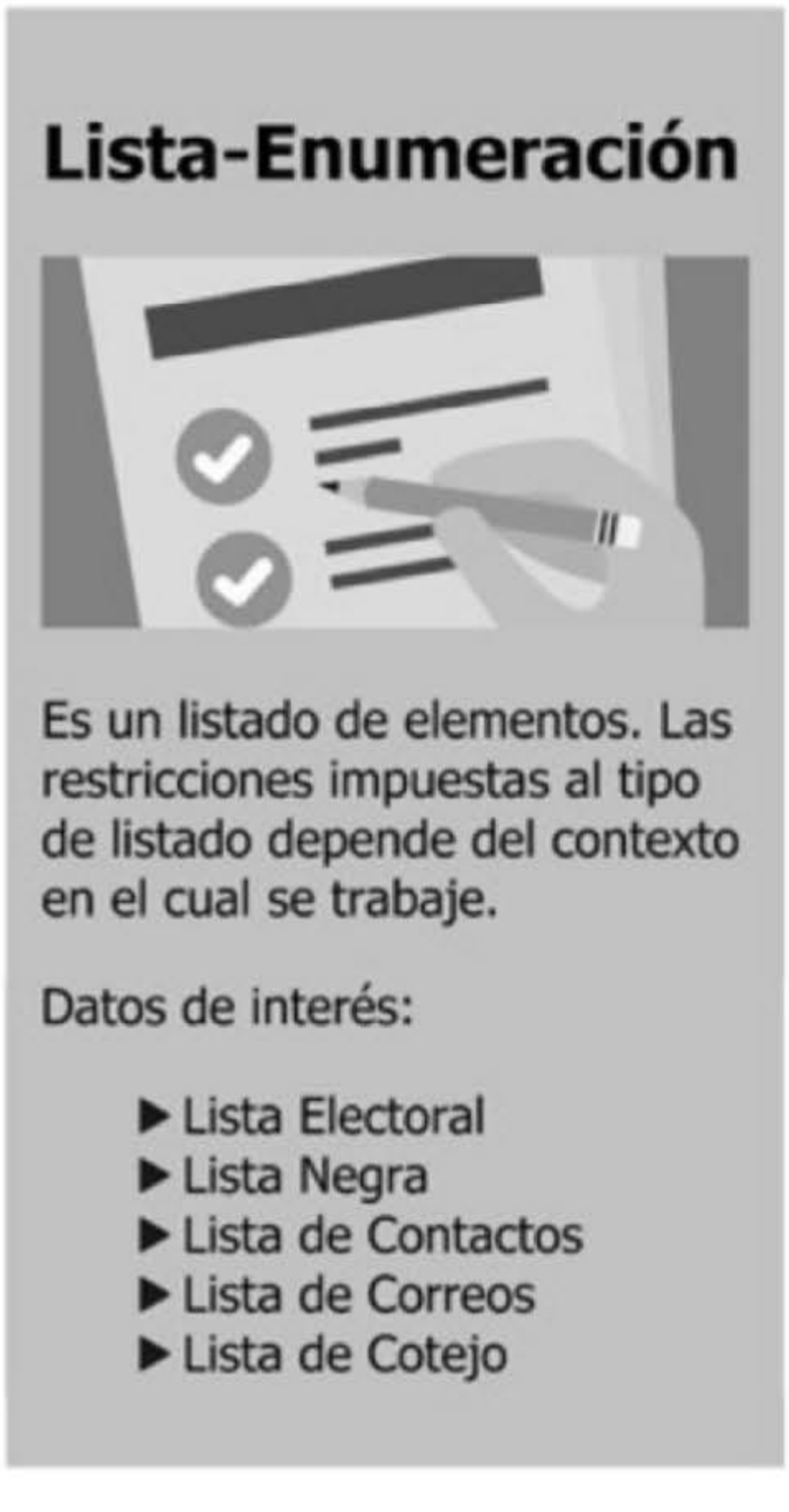
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
            <p>
                Es un listado de elementos. Las restricciones impuestas al tipo de
                listado depende del contexto en el cual se trabaje.
            </p>
            <p>Datos de interés:</p>
            <ul>
                <li>Lista Electoral</li>
                <li>Lista Negra</li>
                <li>Lista de Contactos</li>
```

```
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ul>
    </section>
</body>
</html>
```

Caso desarrollado 5
Listas desordenadas con imagen

Implemente el siguiente diseño dentro de un documento web usando listas desordenadas con imágenes.



Consideraciones:

- Cree un proyecto web llamado **Caso_Viñeta_Imagen**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño y, además, el ícono que se mostrará en la viñeta de la lista.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}

/*Estilo para la seccion*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}

li{
    list-style-type: none;
    list-style-image: url(../img/icono.png);
    margin-left:10px;
    font:14px Tahoma;
}
```

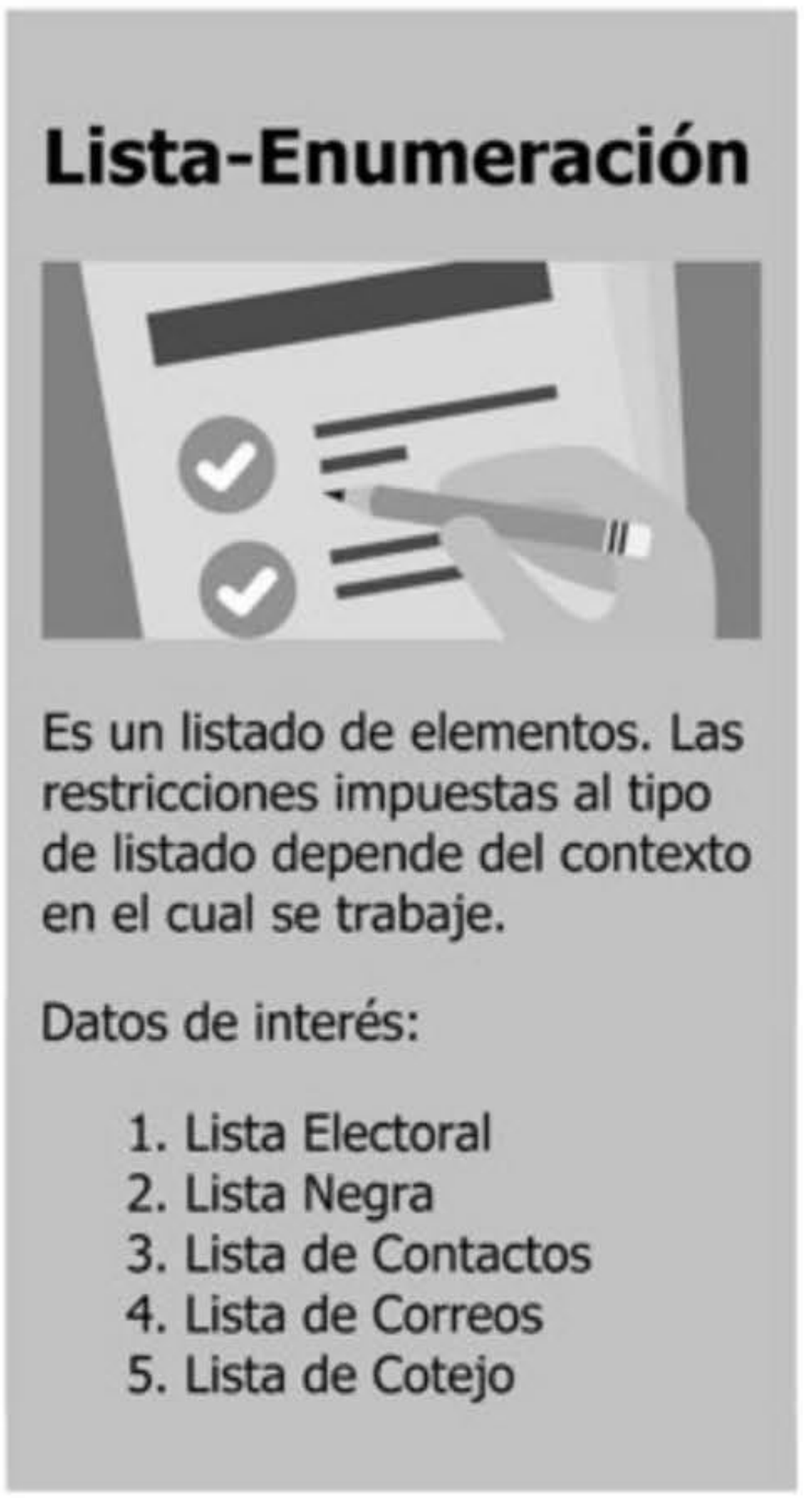
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
            <p>
                Es un listado de elementos. Las restricciones impuestas al tipo de
                listado depende del contexto en el cual se trabaje.
            </p>
```

```
        <p>Datos de interés:</p>
        <ul>
            <li>Lista Electoral</li>
            <li>Lista Negra</li>
            <li>Lista de Contactos</li>
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ul>
    </section>
</body>
</html>
```

Caso desarrollado 6
Listas ordenadas con numeración decimal

Agregue el siguiente diseño dentro de un documento web usando listas ordenadas con numeración decimal.



Consideraciones:

- Cree un proyecto web llamado **Caso_Ordenada_Decimal**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}
/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}
/*Estilo para la imagen*/
img {
    width: 200px;
}
ol{
    list-style-type: decimal;
}
```

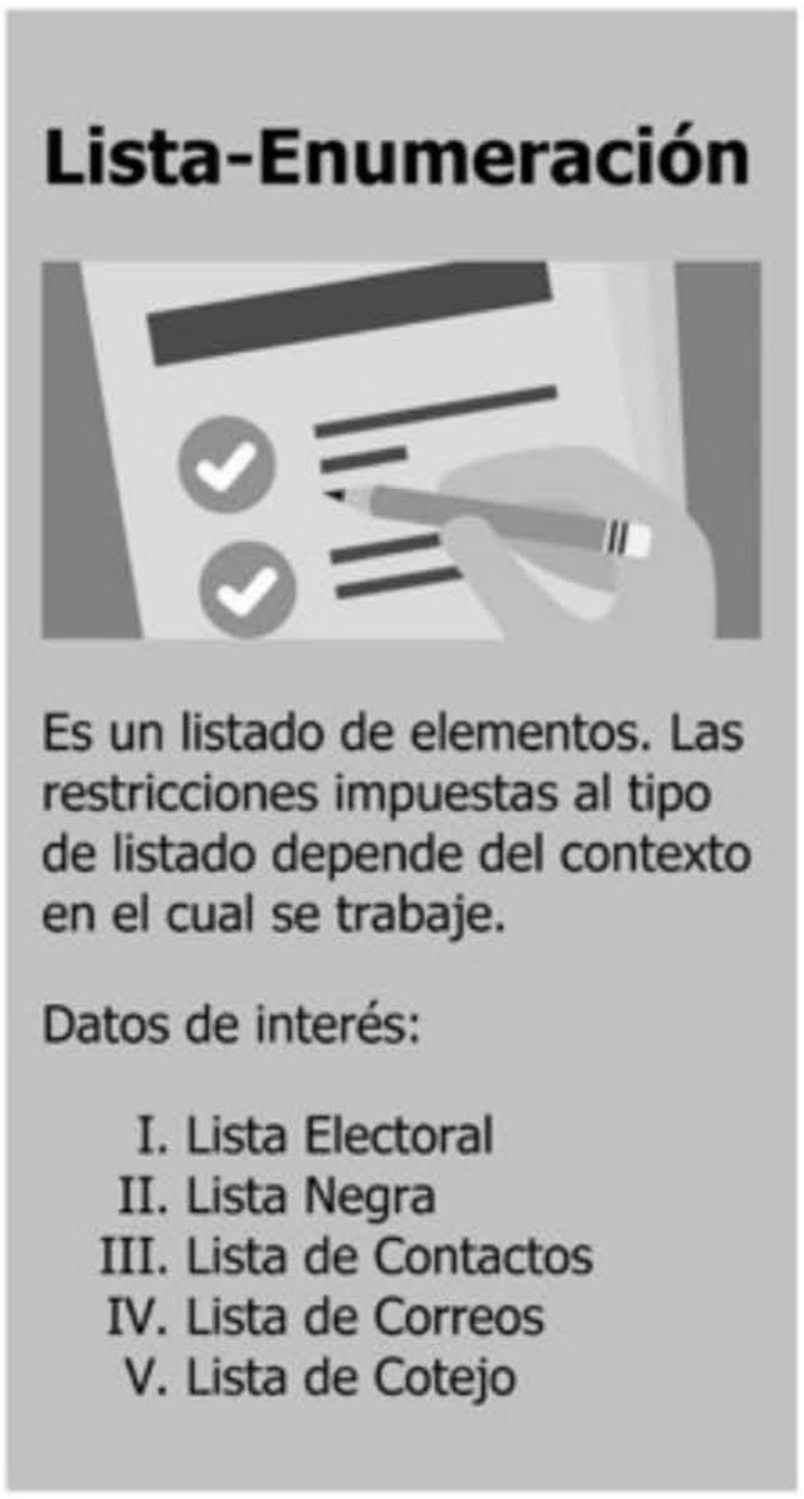
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <title>Listas Desordenadas</title>
    <meta charset='UTF-8'>
    <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
<body>
    <section>
        <header>
            <h2>Lista-Enumeración</h2>
            <img src='img/foto.png' alt=""/>
        </header>
        <p>
            Es un listado de elementos. Las restricciones impuestas al tipo de
            listado depende del contexto en el cual se trabaje.
        </p>
```

```
        <p>Datos de interés:</p>
        <ol>
            <li>Lista Electoral</li>
            <li>Lista Negra</li>
            <li>Lista de Contactos</li>
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ol>
    </section>
</body>
</html>
```

Caso desarrollado 7
Listas ordenadas con numeración romana

Implemente el siguiente diseño dentro de un documento web usando listas ordenadas con numeración romana.



Consideraciones:

- Cree un proyecto web llamado **Caso_Ordenada_Romana**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}
/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}
/*Estilo para la imagen*/
img {
    width: 200px;
}

ol{
    list-style-type: upper-roman;
}
```

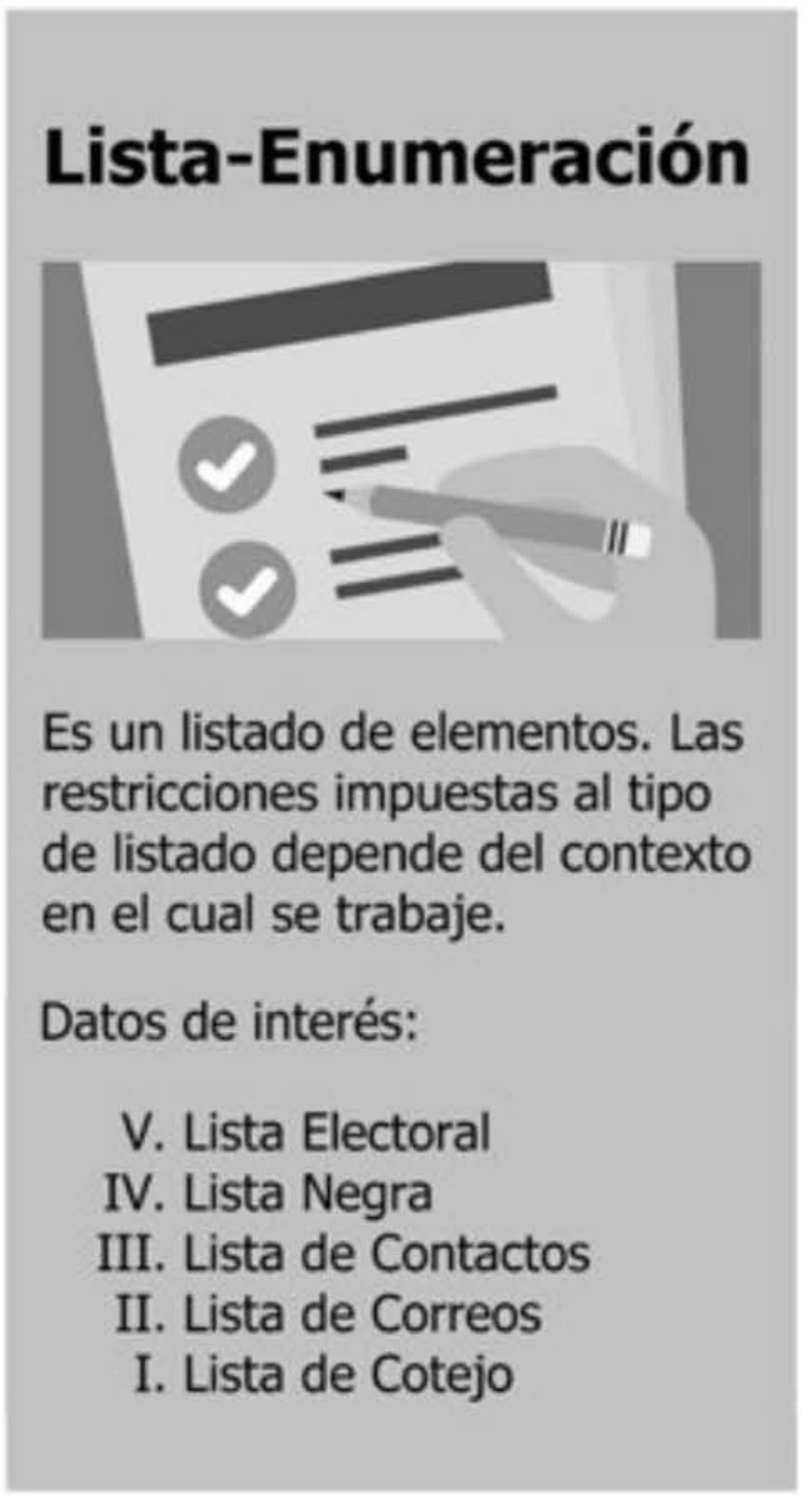
2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <title>Listas Desordenadas</title>
    <meta charset='UTF-8'>
    <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
<body>
    <section>
        <header>
            <h2>Lista-Enumeración</h2>
            <img src='img/foto.png' alt=""/>
        </header>
        <p>
            Es un listado de elementos. Las restricciones impuestas al tipo de
            listado depende del contexto en el cual se trabaje.
        </p>
```

```
        <p>Datos de interés:</p>
        <ol>
            <li>Lista Electoral</li>
            <li>Lista Negra</li>
            <li>Lista de Contactos</li>
            <li>Lista de Correos</li>
            <li>Lista de Cotejo</li>
        </ol>
    </section>
</body>
</html>
```

Caso desarrollado 8
Listas ordenadas con numeración romana descendente

Agregue el siguiente diseño dentro de un documento web usando listas ordenadas de forma descendente y con numeración romana.



Consideraciones:

- Cree un proyecto web llamado **Caso_Romana_Descendente**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar la imagen del diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento web*/
body{
    font-family:Tahoma;
    font-size:14px;
}

/*Estilo para la sección*/
section{
    background:#CCC;
    max-width:200px;
    padding: 10px;
}

/*Estilo para la imagen*/
img {
    width: 200px;
}

ol{
    list-style-type: upper-roman;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Listas Desordenadas</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <header>
                <h2>Lista-Enumeración</h2>
                <img src='img/foto.png' alt=""/>
            </header>
```

```
<p>
   Es un listado de elementos. Las restricciones impuestas al tipo de
   listado depende del contexto en el cual se trabaje.
</p>
<p>Datos de interés:</p>
<ol reversed>
   <li>Lista Electoral</li>
   <li>Lista Negra</li>
   <li>Lista de Contactos</li>
   <li>Lista de Correos</li>
   <li>Lista de Cotejo</li>
</ol>
</section>
</body>
</html>
```

```
<p>
   Es un listado de elementos. Las restricciones impuestas al tipo de
   listado depende del contexto en el cual se trabaje.
</p>
<p>Datos de interés:</p>
<ol reversed>
   <li>Lista Electoral</li>
   <li>Lista Negra</li>
   <li>Lista de Contactos</li>
   <li>Lista de Correos</li>
   <li>Lista de Cotejo</li>
</ol>
</section>
</body>
</html>
```

CAPÍTULO 6

Enlaces y navegación



Competencias

- Implementa un documento web usando enlaces básicos y diferenciando la URL absoluta y relativa.
- Desarrolla una implementación web utilizando la etiqueta <base>, los atributos de la etiqueta <a> y los enlaces por ancla.

Contenido

6.1 Introducción
6.2 Estructura de la URL
6.3 Enlaces básicos
6.4 Tipos de URL
6.5 La etiqueta <base>
6.6 Atributos de la etiqueta <a>
6.7 Implementación de enlaces por anclas
- Caso desarrollado 1: Uso de la etiqueta <a> y <nav>
- Caso desarrollado 2: Uso de la etiqueta <base>
- Caso desarrollado 3: Proyecto web - Agencia de empleos

6.1 Introducción

Se define como un enlace web a la conexión de un elemento con otro mediante una palabra o conjunto de palabras, algún símbolo, una simple imagen, etc. Este último actualmente es el más usado. Todo esto dependerá de la naturaleza de los enlaces previstos.

Generalmente para enlazar algo hacia otra página web, mostrar un documento, visualizar un video o escuchar música se pueden usar diferentes elementos como un texto o una imagen para asociarlos.

Un ejemplo claro de enlaces se puede observar al ingresar a la URL <www.youtube.com>, en la cual se puede visualizar un video desde un enlace web.



Figura 6.1 Logo de Youtube.
Fuente: <https://es.gizmodo.com/youtube-cambia-de-logotipo-por-primera-vez-en-su-histor-1798536812>.

Si se habla de un portal web, los enlaces podrían alojarse en el mismo documento. Esto se puede observar en la siguiente imagen que muestra el mapa de un sitio web.

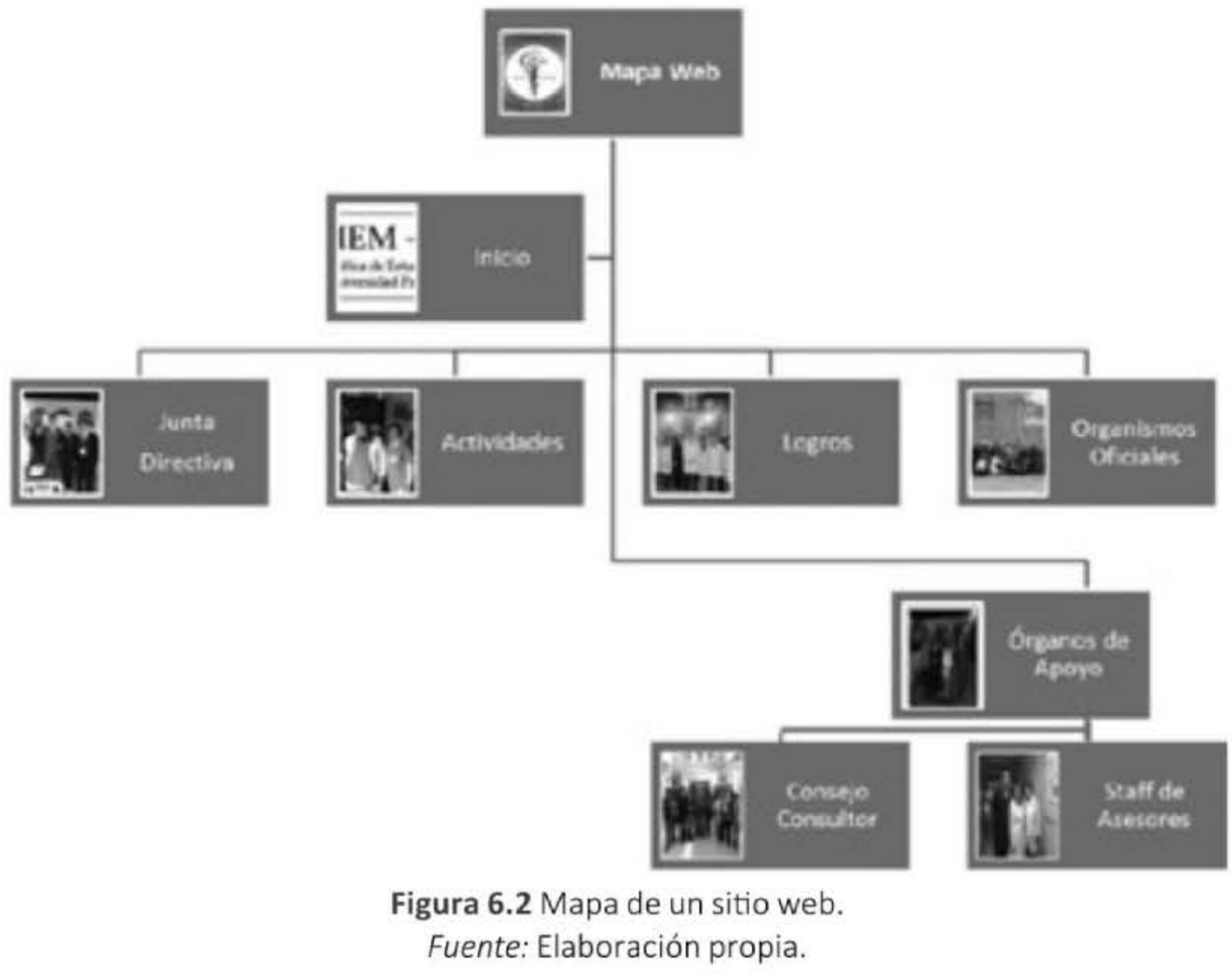


Figura 6.2 Mapa de un sitio web.
Fuente: Elaboración propia.

Los enlaces web se pueden dividir en dos grupos dependiendo del control y la distribución del mapa web realizado por el administrador web. Entonces, se tienen:

- **Control interno**: Se trata de tener el control de los enlaces mediante palabras claves que muestran páginas web que se encuentran alojadas dentro del mismo sitio web.
- **Control externo**: Se trata del control de páginas que no son administradas por el sitio web, sino que dependen de otros servidores web. Aquí se tiene una clara dependencia externa, pues al ser

una página web enfocada en ventas, no debería permitirse que el cliente ingrese a un documento web externo.

Finalmente, el objetivo de administrar enlaces web hace que se tenga un comportamiento especial de los algoritmos que se encuentran en los buscadores, particularmente, el algoritmo PageRank1 de Google. Este último asigna un valor numérico que representa la importancia que tiene una página web tiene en Internet. La idea de Google es que cuando una página realiza un enlace a otra web, esto significa un voto asignado a la página visitada.



Figura 6.3 Modelo de enlace externo.
Fuente: <https://www.humanlevel.com/diccionario-marketing-online/pagerank-google>.

6.2 Estructura de la URL

El manejo de las URL es algo cotidiano cuando nos movemos por Internet, ya que un porcentaje muy alto del tiempo nos lo dedicamos a escribir en el navegador direcciones URL.

La estructura general de una URL es la siguiente:

protocolo://servidor/directorio/página

Donde:

- **Protocolo:** Aquí se define la forma de comunicarse entre las computadoras participantes de la red. Es decir, se le indica a la computadora qué tipo de protocolo debe usarse para realizar la conexión. Los protocolos más conocidos son:
  - **http–HiperText Transfer Protocol**: Es el protocolo más usado en la web. Su puerto por defecto es el 80.
  - **ftp–File Transfer Protocol**: Es el protocolo que se usa para transferir archivos a un servidor. Su puerto por defecto es el 21.
  - **mailto**: Es el puerto que se usa para indicar una dirección de correo electrónico a la cual se desea acceder.
- **Servidor:** Aquí se define el nombre de la máquina destino, también llamada "servidor", puesto que proveerá de información. Estos se componen de tres partes:

RED	NOMBRE DEL SERVIDOR	TIPO Y PAÍS
www	google	com.pe

» **Red:** También es conocido como *host* el cual hace referencia a cualquier tipo de computadora conectada en la red, el más usado es www.

» **Nombre del servidor:** Es el nombre que se le asigna a la máquina destino de la cual se desea obtener la información solicitada. El nombre asignado es comercialmente distribuido y suele ser el patrón representativo de una organización. Por ejemplo: Google, Youtube y Facebook.

» **Tipo y país:** Representan la ocupación del servidor, es decir, el rubro al cual se accede como, por ejemplo, **.com** (comercial), **.net** (relacionado a la red) y **.org** (organizaciones), **.gob** (organizaciones del gobierno), **.mil** (organizaciones militares), **.edu** (organizaciones educativas). Al final también se incluyen zonas geográficas como, por ejemplo, **.pe** (Peru), **.es** (España), etc.

› **Directorio:** Su trabajo es similar a los directorios del sistema operativo. Quiere decir, organizará las páginas que se tienen en el servidor. Por ejemplo, **<http://www.editorialmacro/autor/pagina1.html>**.

Aquí pude observarse el directorio **autor** que representa el repositorio de las páginas de los autores de la editorial.

› **Página:** Representa la página web a la cual se desea acceder. Hay que tener en cuenta que la página **index.html** se abre por defecto sin necesidad de digitarla.

6.3 Enlaces básicos

Los enlaces básicos hacen referencia a las URL conocidas usando la etiqueta **<a>**. Hay que considerar que, si la URL es externa, se tiene que detallar toda la URL; mientras que, si la página web se encuentra dentro del mismo sitio web, solo se hará referencia al nombre de archivo y su extensión. El formato general es:

```
<a>Texto de enlace</a>
```

6.3.1 *Enlazando a otra página web*

La idea de realizar los enlaces es poder navegar sobre un sitio web de la mejor manera posible. Por lo tanto, se usará el atributo **href** de la etiqueta **<a>** de la siguiente manera:

```
<a href='Ruta del archivo y extensión'>Texto de Enlace</a>
```

Por ejemplo, en <**a href**="http://www.editorialmacro.com">Visita Editorial Macro</**a**>,

se puede observar que el atributo **href** hace referencia a la dirección URL completa, pasando por el protocolo hasta el nombre del documento web que, para este caso, hace referencia al valor predeterminado llamado **index.html**.

Por otro lado, si se va a realizar una referencia a un documento web que se encuentre dentro del mismo sitio o proyecto web, se tendría el siguiente formato:

```
<a href='contactos.html'>Contáctenos</a>
```

Se puede observar que el atributo **href** hace referencia solo al documento web, esto se debe a que todos los archivos se encuentran dentro de la misma carpeta y no es necesario su especificación completa.

6.3.2 *Enlazando a un* e-mail

Los enlaces pueden referenciar a un email, pero debemos considerar que en su computadora deberá estar previamente configurado su correo y un servidor correcto ya que al enlazar un email apertura el Outlook como medio de comunicación. Por ejemplo:

```
<a href='mailto:servicioalcliente@empleos.com'>Escribanos</a>
```

Ejemplo 1

Véase la implementación de un menú de opciones, haciendo uso de los enlaces básicos.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
/*Definición del marco*/
section{
    width: 350px;
    margin: 10px;
    padding: 4px;
    background-color: blue;
}

/*Quitar la viñeta de la lista y definir tipo de letra*/
#lista ul {
    list-style:none;
    font-family: Tahoma;
    font-size:16px;
}

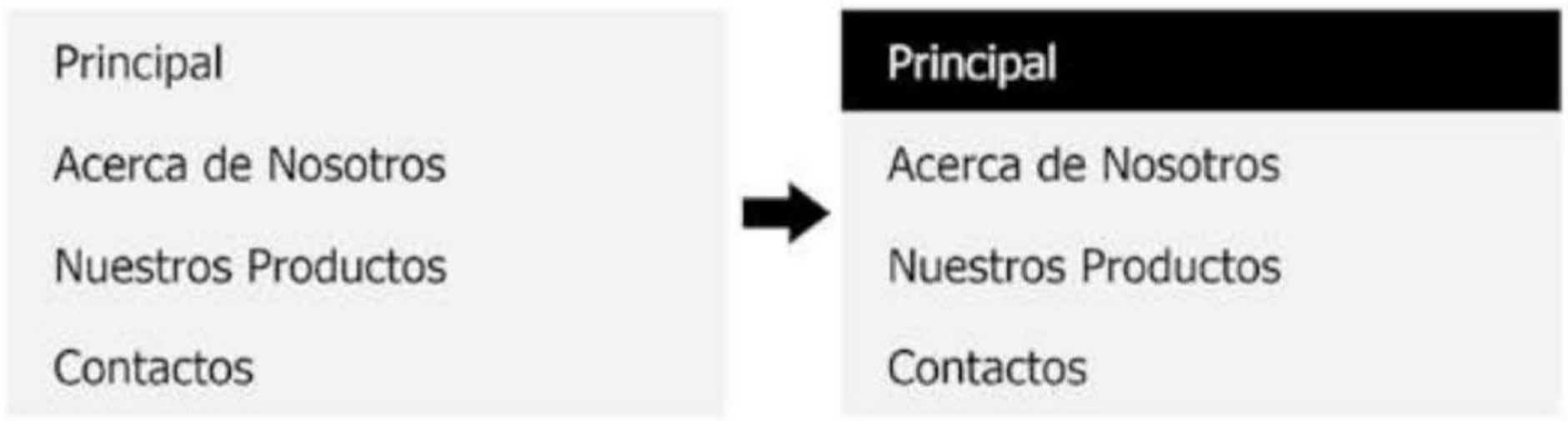
/*Definicion de color y quitar el subrayado*/
#lista ul li a {
    text-decoration:none;
    color:white;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <link href='estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
   <body>
      <section id='lista'>
         <ul>
            <li><a href='principal.html'>Principal</a></li>
            <li><a href='acerca.html'>Acerca de Nosotros</a></li>
            <li><a href='productos.html'>Nuestros Productos</a></li>
            <li><a href='mailto:atencion@tienda.com'>Contactos</a></li>
         </ul>
      </section>
   </body>
</html>
```

Ejemplo 2

Véase la implementación de un menú de opciones de forma vertical, donde al posicionar el puntero del mouse sobre una opción, se muestra un color de fondo en el enlace. En esta implementación se hará uso de los enlaces básicos.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
/*Definición del tipo de letra*/
body{
   font-family: Tahoma;
}
```

```css
/*Definición del ancho de la sección*/
section{
    width: 250px;
    background-color: #f1f1f1;
}

/*Ocultando Viñeta y anulando su espacio*/
ul {
    padding: 0px;
    list-style-type: none;
}

/*Definición de los enlaces*/
li a {
    display: block;
    color: #000;
    padding: 8px 0 8px 16px;
    text-decoration: none;
}

/*Modificar el color de fondo al pasar el puntero del mouse*/
li a:hover {
    background-color: black;
    color: white;
}
```

En el archivo **index.html** se coloca el siguiente código:

```html
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <ul>
                <li><a href='principal.html'>Principal</a></li>
                <li><a href='acerca.html'>Acerca de Nosotros</a></li>
                <li><a href='productos.html'>Nuestros Productos</a></li>
                <li><a href='mailto:atencion@tienda.com'>Contactos</a></li>
            </ul>
        </section>
    </body>
</html>
```

Ejemplo 3

Véase la implementación de un menú de opciones de forma horizontal, donde al posicionar el puntero del mouse sobre una opción, se muestra un color de fondo en el enlace. En esta implementación se hará uso de los enlaces básicos.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
/*Definición del tipo de letra*/
body{
    font-family: Tahoma;
}

/*Definición del ancho de la sección*/
section{
    width: 600px;
    height: 30px;
    text-align: center;
    background-color: #f1f1f1;
}

/*Definición de lista horizontal*/
li {
    display: inline;
}

/*Ocultando viñeta y anulando su espacio*/
ul {
    padding: 5px;
    list-style-type: none;
}

/*Definición de los enlaces*/
li a {
    color: #000;
    padding: 10px;
    text-decoration: none;
}
```

```
/*Modificar el color de fondo al pasar el puntero del mouse*/
li a:hover {
    background-color: black;
    color: white;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <ul>
                <li><a href='principal.html'>Principal</a></li>
                <li><a href='acerca.html'>Acerca de Nosotros</a></li>
                <li><a href='productos.html'>Nuestros Productos</a></li>
                <li><a href='mailto:atencion@tienda.com'>Contactos</a></li>
            </ul>
        </section>
    </body>
</html>
```

6.4 Tipos de URL

Debe mencionarse que, a veces, se deja de lado muchas cosas importantes dentro de la definición de los enlaces web. Una de ellas es el tipo de URL con el que se cuenta, ya que conocer esto podría ahorrar tiempo y espacio en un documento web. Véase los dos tipos de URL más conocidos:

6.4.1 *URL absoluto*

Se le llama así cuando se hace referencia a un elemento por su URL completa (se incluye el protocolo http). Normalmente, es usado para enlaces externos, es decir, de una web a otra. Por ejemplo:

- Enlace URL externo:

```
<a href='http://www.agencia.com'>Agencia de empleos</a>
```

- Enlace a una imagen desde una URL externa:

```
<a href='http://www.agencia.com/images/Foto1.jpg'>Foto1</a>
```

  Este tipo de URL es la que se usa cuando se necesita una imagen, video o algún elemento que se encuentre en otro servidor o dominio.

6.4.2 *URL relativo*

Se le llama así cuando se hace referencia a elementos que se encuentran dentro del mismo sitio web. Esto quiere decir que se debe conocer exactamente como está compuesto el sitio web para poder hacer la referencia correcta. Para realizar una referencia a un elemento contenido dentro del mismo sitio web, se debe especificar la ruta a través de una jerarquía de carpetas que va desde la carpeta actual hasta el elemento vinculado. Véase lo siguiente:

Enlace de una imagen:

```
<a href='imágenes/foto1.jpg'>Foto1</a>
```

Mapa del sitio:

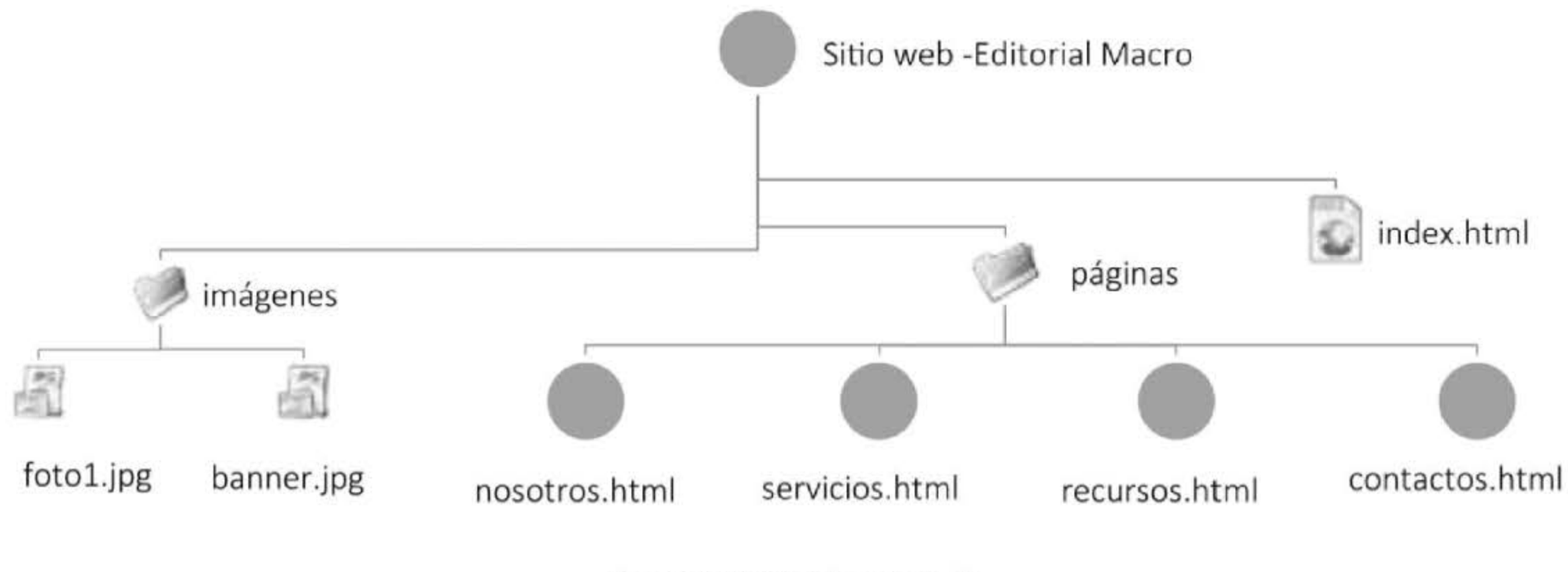


Figura 6.4 Mapa de sitio web.
Fuente: Elaboración propia.

6.4.3 *Diferencias entre URL absoluta y relativa*

- **Si el destino del enlace se encuentra en un nivel inferior al origen:**
  En el siguiente ejemplo se asume que la URL de origen es la carpeta principal y se desea hacer referencia a la página **nosotros.html** que se encuentra en un nivel inferior.

  Origen: <http://www. editorial.com/index.html>
  URL absoluta: <http://www.editorial.com/páginas/nosotros.html>
  URL relativa: <páginas/nosotros.html>

  Lo recomendado en este caso es usar la referencia que muestra la URL Relativa, pues desde el origen se ingresa a la carpeta **páginas** y se muestra el documento **nosotros.html**; en cambio, en la URL absoluta se debe especificar toda la dirección del elemento.

- **Si el destino del enlace se encuentra en el mismo nivel del origen:**
  En el siguiente ejemplo se asume que la URL de origen es la carpeta **páginas**, que se encuentra dentro de la carpeta principal, y se desea hacer referencia a la página **servicios.html**, que se encuentra en la misma carpeta.

Origen: <http://www.editorial.com/páginas/nosotros.html>
URL absoluta: <http://www.editorial.com/páginas/servicios.html>
URL relativa: <servicios.html>

Lo recomendado en este caso es usar la referencia que muestra la URL relativa, pues tanto el archivo de origen como el de destino se encuentran en la misma carpeta.

- **Si el destino del enlace se encuentra en un nivel superior al origen:**

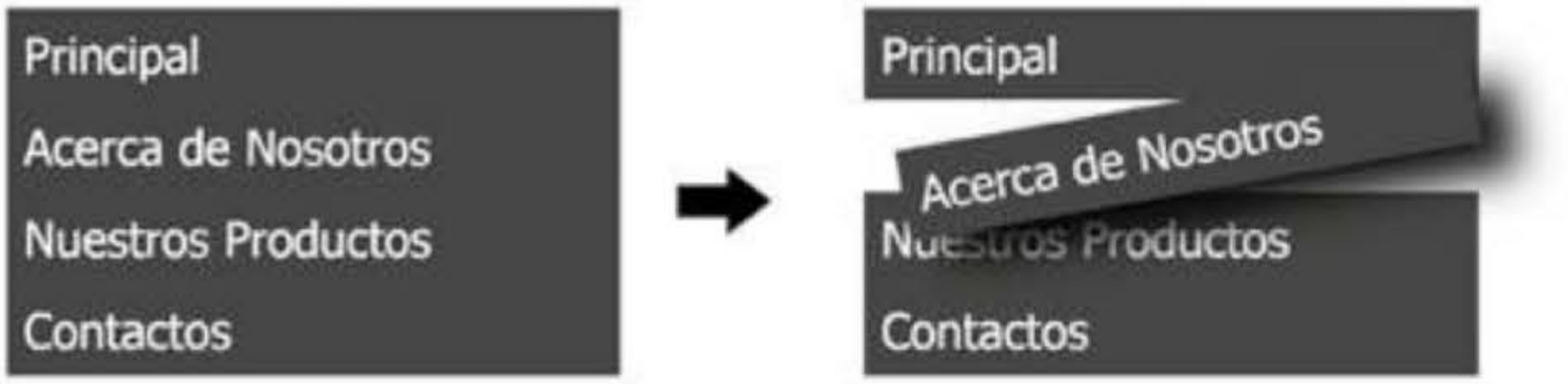
  En el siguiente ejemplo se asume que la URL de origen es la carpeta páginas, que se encuentra dentro de la carpeta principal, y se desea hacer referencia a la imagen llamada **banner.jpg**, que se encuentra dentro de la carpeta **imágenes**.

  Origen: <http://www.agencia.com/páginas/nosotros.html>
  URL absoluta: <http://www.agencia.com/imágenes/banner.jpg>
  URL relativa: </imágenes/banner.jpg>

  Con respecto a la URL relativa, hay que considerar que los dos puntos seguidos hacen subir un nivel y de allí se puede acceder a otro directorio como en el caso anterior.

Ejemplo 4

Véase la implementación de un menú de opciones, haciendo uso de los tipos de URL.



Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
/*Definición del marco*/
section{
    width:180px;
    font-family:Tahoma;
    font-size:14px;
}
```

```
/*Quintando las viñetas y el espacio de la viñeta*/
#lista ul {
    list-style:none;
    padding: 0px;
}

/*Definición del enlace*/
#lista ul li a {
    display:block;
    background-color:#333;
    padding: 5px;
    text-decoration:none;
    color: white;
}

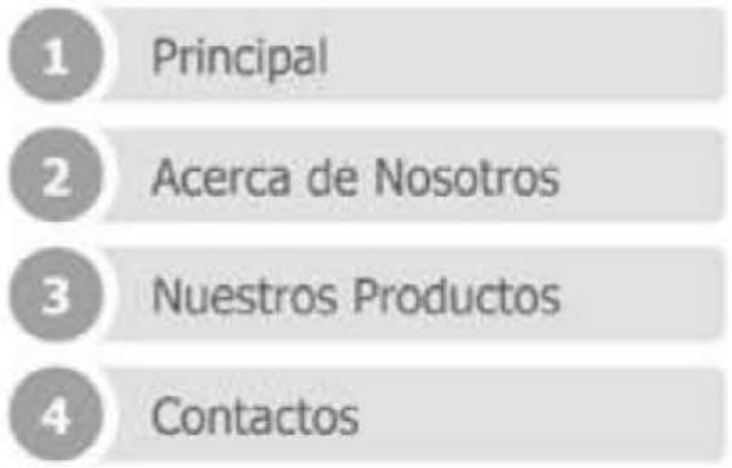
/*Definición del enlace al posicionar el puntero del mouse*/
#lista ul li a:hover {
    -moz-transform:rotate(-5deg);
    -moz-box-shadow:10px 10px 20px #000000;
    -webkit-transform:rotate(-5deg);
    -webkit-box-shadow:10px 10px 20px #000000;
    transform:rotate(-10deg);
    box-shadow:10px 10px 20px #000000;
    margin-left: 10px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section id='lista'>
            <ul>
                <li><a href='#'>Principal</a></li>
                <li><a href='#'>Acerca de Nosotros</a></li>
                <li><a href='#'>Nuestros Productos</a></li>
                <li><a href='#'>Contactos</a></li>
            </ul>
        </section>
    </body>
</html>
```

Ejemplo 5

Véase la implementación de un menú de opciones, haciendo uso de los tipos de URL.



Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
section{
    width: 250px;
}

#lista {
    counter-reset: li;
    list-style: decimal;
    font: 15px 'Tahoma';
    padding: 0;
}

#lista li{
    position: relative;
    display: block;
    padding: .4em .4em .4em 2em;
    *padding: .4em;
    margin: .5em 0;
    background: #ddd;
    color: white;
    text-decoration: none;
    border-radius: .3em;
    transition: all .3s ease-out;
}

#lista li:hover{
    background: #eee;
}

#lista li:hover:before{
    transform: rotate(180deg);
}
```

```
#lista li:before{
    content: counter(li);
    counter-increment: li;
    position: absolute;
    left: -1.3em;
    top: 50%;
    margin-top: -1.3em;
    background: red;
    height: 2em;
    width: 2em;
    line-height: 2em;
    border: .3em solid #fff;
    text-align: center;
    font-weight: bold;
    border-radius: 2em;
    transition: all .3s ease-out;
}

#lista ul li a {
    text-decoration:none;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section id='lista'>
            <ul>
                <li><a href='#'>Principal</a></li>
                <li><a href='#'>Acerca de Nosotros</a></li>
                <li><a href='#'>Nuestros Productos</a></li>
                <li><a href='#'>Contactos</a></li>
            </ul>
        </section>
    </body>
</html>
```

Ejemplo 6

Véase la implementación de un menú de opciones, haciendo uso de los tipos de URL.

1. Principal
2. Acerca de Nosotros
3. Nuestros Productos
4. Contactos

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
section{
   width: 250px;
}

#lista {
   counter-reset: li;
   list-style: none;
   font: 15px 'Tahoma';
   padding: 0;
   margin-bottom: 4em;
   text-shadow: 0 1px 0 rgba(255,255,255,.5);
}

#lista ol {
   margin: 0 0 0 2em;
}

#lista li{
   position: relative;
   display: block;
   padding: 8px;
   margin: 2px;;
   background: #ddd;
   color: white;
}

#lista li:hover{
   background: #eee;
}
```

```
#lista li:before{
    content: counter(li);
    counter-increment: li;
    position: absolute;
    left: -2.5em;
    top: 50%;
    margin-top: -1em;
    background: #fa8072;
    height: 2em;
    width: 2em;
    line-height: 2em;
    text-align: center;
}

#lista li:after{
    position: absolute;
    content: '';
    border: .5em solid transparent;
    left: -1em;
    top: 50%;
    margin-top: -.5em;
    transition: all .3s ease-out;
}

#lista li:hover:after{
    left: -.5em;
    border-left-color: #fa8072;
}

#lista ul li a {
    text-decoration:none;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section id='lista'>
            <ul>
                <li><a href='#'>Principal</a></li>
                <li><a href='#'>Acerca de Nosotros</a></li>
                <li><a href='#'>Nuestros Productos</a></li>
                <li><a href='#'>Contactos</a></li>
            </ul>
```

```
        </section>
    </body>
</html>
```

6.5 La etiqueta <base>

Permite definir una URL absoluta en todo el documento web, es decir, se podrá hacer referencia a todos los elementos que sean relativos a la página web. Esto significa que define un comportamiento genérico para todos los enlaces web.

A continuación, sus características:

- La etiqueta **<base>** permite especificar una URL y un destino por defecto para todos los enlaces relativos del documento web.
- Se puede decir que es una etiqueta que declara una URL de forma global.
- Solo puede aparecer una sola vez dentro del documento web.
- Su especificación se realiza dentro de la etiqueta **<head>**.

Su formato es el siguiente:

```
<head>
    <base href='valor' target='valor' / >
</head>
```

Donde:

- **href**: Aquí se especifica la URL base para todas las URL relativas que contengan el documento web.
- **target**: Aquí se especifica cómo se presentará el documento al momento de realizar el enlace. Se tienen las siguientes opciones:
  - **_blank**: Especifica que el destino sea una nueva ventana del navegador o una nueva pestaña según la naturaleza del navegador web.
  - **_self**: Es el valor por defecto, este abre el destino en el mismo lugar donde se haya llamado al enlace.

6.6 Atributos de la etiqueta <a>

Hasta ahora solo se ha definido la propiedad **href** de la etiqueta **<a>**. Ahora se tratarán algunas propiedades más que resultan interesantes al momento de definir los enlaces.

6.6.1 *Atributo accesskey*

Es la definición de atajos mediante el teclado. Los atajos se definen y accionan por la combinación de una letra con la tecla **<Alt>**. El estándar HTML5 incorpora los **accesskeys** como atributo global que se pueden usar en cualquier etiqueta HTML.

Ejemplo 7

```
<a href='http://www.editorialmacro.com' accesskey='e'>Editorial Macro</a>
```

En el script anterior se está definiendo la combinación de teclas <Alt>+<E> para poder acceder a la URL <www.editorialmacro.com >.

Ahora bien, véase lo siguiente:

```
<form>
   name: <input type='text' accesskey='n'/>
   password: <input type='password' accesskey='p'/>
   <input type='submit' accesskey='s'/>
</form>
```

Aquí se está definiendo las combinaciones de teclas para el acceso a las cajas de textos de un formulario.

6.6.2 Atributo hreflang

Cuando se visita un sitio web, debe tenerse en cuenta que todo el mundo puede visualizar dicha información. Existe una gran cantidad de resultados que se pueden obtener en Google sobre un determinado tema. Basado en lo anterior, hay que imaginar las distintas versiones que puede tener un sitio web.

Un ejemplo muy claro se puede observar con la URL <http://www.mercadolibre.com>, pues al no especificar qué versión de país se quiere mostrar, Google lo organizará según la versión del buscador que se use y el número de IP. Esto quiere decir que si se busca desde <http://www.google.com.pe>, el resultado de la búsqueda sería:



Como se observa Google muestra la lista habitual de búsqueda mostrando como prioridad la URL <http://www.mercadolibre.com.pe>. Esto es debido a que la empresa MercadoLibre tiene una organización indexada de sus sitios web según la región disponible. Entonces, si se busca la misma página desde Google Colombia (<http://www.google.com.co>) el resultado sería el siguiente:



La etiqueta **hreflang** define el lenguaje de destino según la región donde se encuentre. Esto ayudará a optimizar las búsquedas que se realiza sobre un sitio web de la manera más correcta posible. Google usa esta etiqueta para identificar el idioma y el país al que va dirigido un enlace web. Su formato es el siguiente:

```
<a href='url' hreflang='idioma'>Texto de enlace</a>
```

Con respecto al código que se debe asignar al **hreflang**, debe usarse el estándar basado en el ISO-639-1 para los formatos de idiomas, véanse algunos códigos:

AF	afrikaans
BG	búlgaro
DA	danés
DE	alemán
EN	inglés
ES	español (o castellano)
FR	francés
GA	irlandés (o gaélico)
IE	occidental
IT	italiano

JA	japonés
KO	coreano
LA	latín
PT	portugués
QU	quechua
RO	rumano
RU	ruso
TH	tailandés
ZH	chino

Opcionalmente, también se puede hacer referencia a los países, basándose en el ISO-3166-1 Alfa 2. Véase una lista de los países más conocidos:

ARGENTINA	ar	**FRANCIA**	fr
CANADÁ	ca	**HONDURAS**	hn
CHILE	cl	**MÉXICO**	mx
CHINA	cn	**NICARAGUA**	ni
COLOMBIA	co	**PARAGUAY**	py
COSTA RICA	cr	**PERÚ**	pe
CUBA	cu	**PUERTO RICO**	pr
ECUADOR	ec	**URUGUAY**	uy
ESPAÑA	es	**VENEZUELA**	ve
EE.UU.	us		

Ejemplo 8

```
<a href='index.html' hreflang='pe'>Perú</a>
<a href='index.html' hreflang='es-ve'>Venezuela</a>
<a href='index.html' hreflang='en-us'>EE.UU.</a>
```

En los scripts pueden verse las referencias a la página **index.html** que corresponde a los países de Perú, Venezuela y Estados Unidos, respectivamente.

6.6.3 *Atributo rel*

Permite establecer una relación entre la página actual y la página web de destino, por la que se accederá mediante un enlace. Presenta los siguientes atributos:

- **alternate**: Asigna una versión alternativa del documento web. Esto es muy usado al hacer referencia al documento web en diferentes idiomas.

```
<a rel='alternate' hreflang='en' href='index.html'>Principal</a>
```

- **author**: Especifica el nombre del autor al realizar un enlace hacia un artículo con derechos de autor.

```
<a rel='author' href='index.html'> Principal </a>
```

- **bookmark:** Establece un enlace constante dentro de un documento web. Podría emplearse para definir un título, un artículo externo, algún comentario, etc.

```
<a href='http://www.google.com.pe' rel='bookmark'>Ir a Google</a>
```

- **help:** Hacer referencia a enlaces que proveen de ayuda en el documento web.

```
<a rel='help' href='mapa.html'> Mapa del sitio </a>
```

- **license:** Permite referenciar a un documento que proporciona información de derechos de autor sobre el contenido de la página.

```
<p>Todos los derechos reservados <a rel='license'>@2015</a></p>
```

- **nofollow:** Se utiliza para que los motores de búsqueda no tengan en cuenta los enlaces externos y, así, pueda reducirse el *spam*.

```
<a href='http://www.google.com.pe' rel='nofollow'>Ir a Google </a>
```

- **canonical**: Cuando Google detecta una página con la especificación canónica, sus algoritmos mostrarán el título y un fragmento de la versión canónica en los resultados de búsqueda.

```
<link rel='canonical' hreflang='es' href='http://www.google.com' />
```

6.6.4 *Atributo target*

Permite establecer el modo de abrir un enlace web. Recurrentemente un usuario se enlaza a los documentos haciendo clic sobre un link y, seguidamente, la página de destino sustituye a la página actual en la misma ventana del navegador. Esto puede ser o no lo que el usuario quería; eso se sabrá solo si vuelve a la página anterior. En este caso **target** presenta cuatro valores que son:

- **_self:** Es el valor por defecto. La carga de una página web se realiza en la misma ventana.

```
<a href='index.html' target="_self">Principal</a>
```

- **_blank:** La carga de la página web se realiza en una nueva ventana y envía el contenido del enlace allí. En algunos navegadores se apertura una pestaña y es allí donde se muestra el contenido, como es el caso de Google Chrome.

```
<a href='contacto.html' target="_blank"> Contacto </a>
```

- **_top:** Permite identificar a la ventana de nivel superior según la navegabilidad realizada en la web. Se mostrará el contenido web en la ventana inicial de navegación.

```
<a href='contacto.html' target='_top'> Contacto </a>
```

- **_parent:** Permite identificar a la ventana de nivel inmediatamente superior. Se muestra el contenido web en la ventana inmediata superior.

```
<a href='contacto.html' target='_parent'> Contacto </a>
```

6.6.5 *Atributo title*

Permite asignar un texto alternativo sobre un enlace al posicionar el puntero del mouse por encima del mismo. Si se quiere mostrar un mensaje como **Visita nuestra web** en el enlace **Editorial Macro**, véase el código siguiente:




```
<a href='http://www.editorialmacro.com'
   title='Visita nuestra web'>Editorial Macro
</a>
```

6.6.6 *Atributo type*

Permite especificar los tipos MIME (*Internet Media Types*) para definir los formatos de archivos en los enlaces web y, así, el sistema sepa cómo manejarlos.

```
<a href='http://www.editorialmacro.com'
   type='text/html'>Editorial Macro
</a>
```

Véase una lista de los MIME más usados en la web:

application/acad	.dwg	text/html	.html
application/excel	.xls	text/pascal	.pas
application/hlp	.hlp	text/plain	.c
application/java	.clas	text/plain	.c++
application/java-byte-code	.clas	text/plain	.jav
application/mspowerpoint	.pps	text/plain	.java
application/mspowerpoint	.ppt	text/plain	.log
application/msword	.doc	text/plain	.text
application/x-compressed	.zip	text/plain	.txt
audio/mpeg3	.mp3	text/xml	.xml
image/bmp	.bmp	video/avi	.avi
image/jpeg	.jpeg	video/mpeg	.mpeg
image/jpeg	.jpg	video/mpeg	.mpg
text/asp	.asp	video/msvideo	.avi
text/css	.css	video/quicktime	.mov
text/html	.htm		

6.7 Implementación de enlaces por anclas

En el caso anterior se vio cómo usar los enlaces entre las páginas haciendo uso de los estilos CSS, pero en ocasiones se tiene que enlazar directamente al contenido de la misma página web, ya que el usuario

puede encontrarse en un artículo demasiado extenso. Esto puede ser controlado por medio de un tipo de enlace a los que se le denomina ancla.

Un ancla puede definirse como el hito o llave de un determinado lugar dentro del mismo documento web. Véase el formato para la creación de anclas:

```
<a name='nombre del ancla'>punto de retorno</a>
```



Figura 6.5 Modelo de enlaces por anclaje.
Fuente: Elaboración propia

Se debe separar la definición de ancla y la del enlace que se hace a esta. La definición del ancla permite asignar un hito o punto de retorno. Este será determinado, por ahora, en el mismo documento. El código HTML5 para la imagen anterior sería el siguiente:

```
<a name='arriba'>Listado de enlaces</a>
<a name='ancla1'>ancla1</a>
<a name='ancla2'>ancla2</a>
<a name='ancla3'>ancla3</a>
```

La propiedad **name** de la etiqueta **<a>** permite asignar un nombre a una sección del documento web. Ahora bien, la referencia hacia el ancla debe seguir el siguiente formato:

```
<a href='#nombre del ancla'>Enlace al ancla</a>
```

Siguiendo con el ejemplo basado de la imagen anterior, se tendría el siguiente código para hacer las referencias correspondientes a las anclas:

```
<a href='#arriba'>arriba</a>
<a href='#ancla1'> ancla1 </a>
<a href='#ancla2'> ancla2 </a>
<a href='#ancla3'> ancla3 </a>
```

Caso desarrollado 1
Uso de la etiqueta <a> y <nav>

Haciendo uso de las etiquetas de enlace y navegación, implemente el siguiente diseño de menú de opciones:



Consideraciones:

- Cree un proyecto web llamado **Caso_Menu_Basico**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para el fondo del documento*/
body{
    font-family:Tahoma, Geneva, sans-serif;
    font-size:12px;
}

nav{
    /*Estilos para Bordes redondeados*/
    border-radius:10px;
    /*Estilo para Degradados*/
    background-image: -webkit-gradient(linear, left top,
    left bottom, from(#FFF), to(#CCC));
    overflow:hidden;
    padding:5px;
    width:600px;
}

/*Estilo para la lista desordenada*/
nav ul{
    list-style:none;
    margin:0 10px 0 10px;
}

/*Bordes redondeados para los elementos de la lista*/
nav ul li{
    border-radius:5px;/*Estandar por defecto*/
    float:left;
    font-family:Tahoma, Geneva, sans-serif;
```

```
    font-size:16px;
    font-weight:bold;
    margin-right:10px;
    text-align:center;
}

/*Sombras al posicionar el mouse por encima de los elementos*/
nav ul li:hover{
    box-shadow: 1px -1px 0px #999;
    border:1px solid #E3E3E3;
}

/*Estilo para los enlaces*/
nav ul li a{
    color:#000;
    display:block;
    padding:10px;
    text-decoration:none;
    /*Transiciones*/
    -webkit-transition: 0.4s linear all;
    -moz-transition: 0.4s linear all;
    -o-transition: 0.4s linear all;
    transition: 0.4s linear all;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset='UTF-8'>
    <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <h2>EDITORIAL MACRO S.A.C</h2>
        </header>
        <nav>
        <ul>
            <li><a href='inicio.html'>Principal</a></li>
            <li><a href='nosotros.html'>Nosotros</a></li>
            <li><a href='servicios.html'>Servicios</a></li>
            <li><a href='recursos.html'>Recursos</a></li>
            <li><a href='contactenos.html'>Contáctenos</a></li>
        </ul>
        </nav>
    </body>
</html>
```

Caso desarrollado 2
Uso de la etiqueta <base>

Haciendo uso de las etiquetas de enlace y navegación, implemente el siguiente diseño de menú de opciones:



Consideraciones:

- Cree un proyecto web llamado **Caso_Base**.
- Cree una carpeta con el nombre **css** dentro del proyecto para almacenar el archivo **estilo.css**.
- Cree una carpeta con el nombre **img** dentro del proyecto para guardar las cuatro imágenes mostradas en el diseño.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
body{
    font-family:Tahoma;
    font-size:12px;
}

section{
    width:750px;
    background: beige;
    padding: 10px;
}

p{
    font-size:16px;
}

img{
    width: 180px;
    height: 145px;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <base href='img/'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <h2>Agencia de Viajes</h2>
        </header>
        <section>
            <p>Vista de las principales ciudades</p>
            <img src='ciudad1.jpg'/>
            <img src='ciudad2.jpg'/>
            <img src='ciudad3.jpg'/>
            <img src='ciudad4.jpg'/>
        </section>
    </body>
</html>
```

- Si la página que se visualiza es **index.html** y a su vez se encuentra dentro de la carpeta principal del proyecto, entonces esta página será el origen desde donde se debe tener en cuenta las demás URL.

```
<base href='img/'>
```

El código anterior hace referencia a la carpeta **img** creada dentro del proyecto web. Por lo tanto, se puede acceder a las imágenes de la siguiente manera: ****.

Así se evita escribir esta forma de enlace: ****.

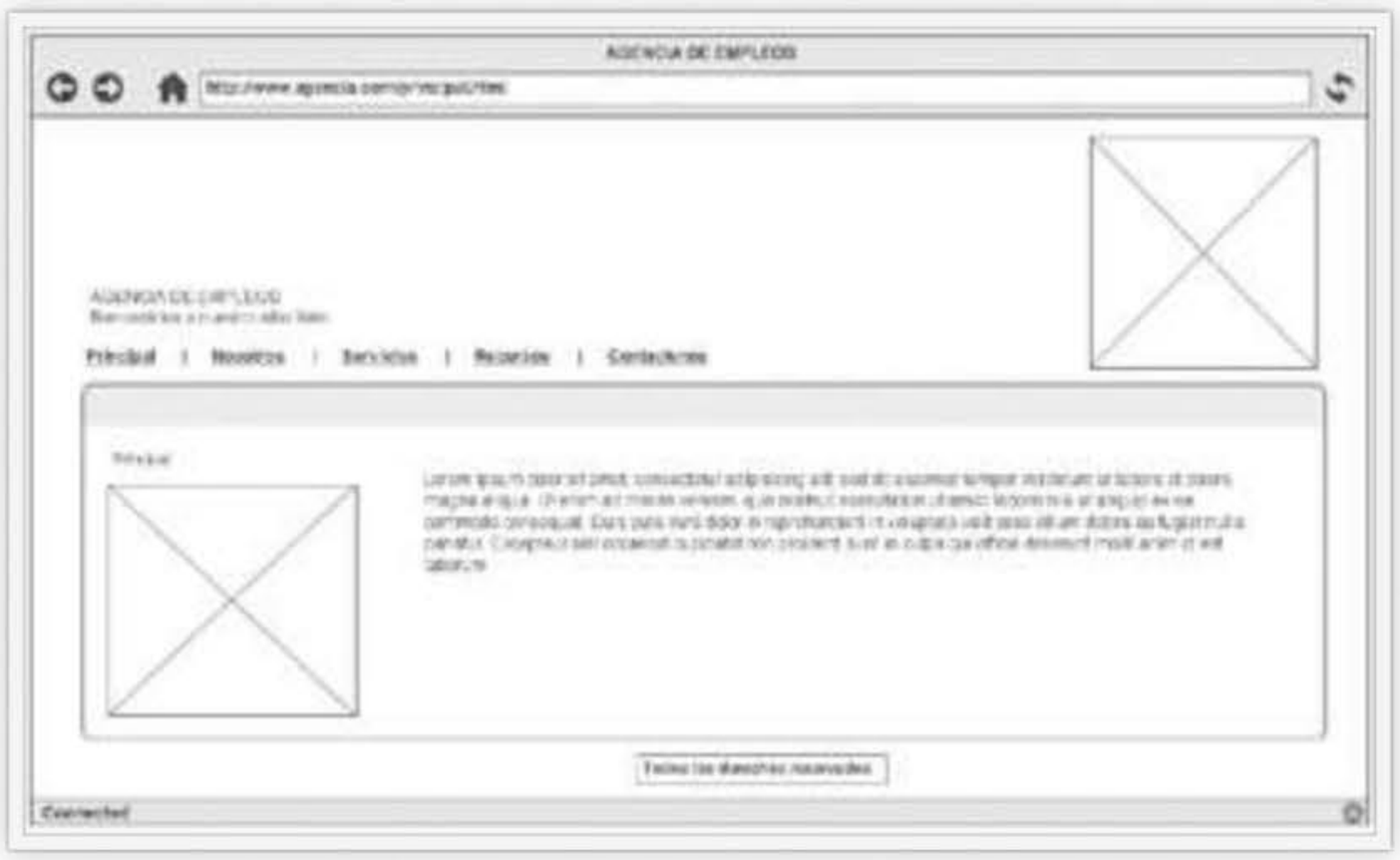
Caso desarrollado 3
Proyecto web - Agencia de empleos

Una agencia de empleos en Lima ha pedido la implementación de un sitio web basado en HTML5 y CSS3. El diseñador web a cargo le propone el siguiente esquema:



Donde:

- **img**: Servirá para alojar a todas las imágenes usadas en la implementación del sitio web.
- **css**: Alojará los archivos de tipo CSS, es decir, los estilos que darán formato a los elementos contenidos en los documentos web.
- **Principal**: Es el archivo principal del proyecto, el cual albergará a la página de presentación del sitio web. Este archivo deberá tener el siguiente esquema:



› **Nosotros**: Es el archivo que permitirá describir quiénes son en la empresa y qué los diferencia de las demás empresas del mismo rubro. Este archivo deberá contar con el siguiente esquema:



› **Servicios**: Es el archivo que permitirá describir los servicios que brinda la agencia de empleos y deberá contar con el siguiente esquema:



› **Recursos**: Es el archivo que permitirá mostrar los recursos de del proyecto web. Este archivo deberá contar con el siguiente esquema:



› **Contáctenos:** Usará el enlace hacia el correo electrónico informa@agencia.com

Según todo lo anterior, elabore el sitio web de la agencia de empleos basado en la propuesta del diseñador.

Consideraciones:

› Cree un proyecto web llamado **Caso_ProyectoWeb_Agencia**.
› Cree una carpeta llamada **css** dentro del proyecto para almacenar el archivo **estilo.css**.
› Cree una carpeta con el nombre **img** dentro del proyecto para almacenar las cuatro imágenes mostradas en el diseño.
› La siguiente imagen muestra el factor común que deben tener todas las páginas web del sitio Agencia de Empleos:



› Los enlaces deberán estar asignados a sus respectivas páginas web, por lo que se debe respetar el siguiente patrón:

ENLACE	ARCHIVO WEB	TECLA ABREVIADA
Principal	index.html	I
Nosotros	nosotros.html	N
Servicios	servicios.html	S
Recursos	recursos.html	R
Contáctenos	mailto:informa@agencia.com	C

› Asigne a cada elemento del menú un título, en el que se muestre un mensaje referente al elemento de enlace.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```
/*Estilo para todo el documento*/
body {
    background: #1EB0D7;
    font-family: Tahoma;
    font-size: 11px;
    width: 1200px;
    margin: auto;
}

/*Estilo para la imagen arriba*/
#logo {
    background: url(../img/logo.png) no-repeat right;
    margin: 0 0px;
    background-size: 60%;
    padding: 25em 0em 5em 0em;
    text-transform: uppercase;
}

/*Estilo para la etiqueta a incluida en H1*/
#logo h1 a {
    margin: 0em;
    padding: 0;
    letter-spacing: -.1em;
    text-decoration: none;
    font-size: 2em;
    font-weight: bold;
    color: #FFFFFF;
}
```

```
/*Estilo para el subtitulo 'BIENVENIDOS A NUESTRO SITIO WEB'*/
#subtitulo {
    padding: 0em;
    margin-top: -1em;
    font-size: 1.5em;
    color: #BDF2FF;
}

/*Estilo para los elementos de la lista del menú*/
#menu li {
    display: inline-block;
    margin-top: -2em;
}

/*Estilo para la etiqueta del Menú de opciones*/
#menu a {
    display: block;
    float: left;
    margin-right: 0.12em;
    padding: 1em 1.5em;
    background: #178EAE;
    box-shadow: inset 0px -2px 0px 0px rgba(0, 0, 0, 0.25);
    text-decoration: none;
    text-shadow: 1px 1px 0px #0C749C;
    text-transform: uppercase;
    font-size: .90em;
    color: #FFFFFF;
}

/*Estilo para la sección que muestra el contenido*/
#seccion {
    width: 1150px;
    padding: 0px 10px;
    background: #FFF;
    height: 270px;
}

/*Estilo para la sección los servicios*/
#seccionServicios {
    width: 1150px;
    padding: 20px 40px;
    background: #FFF;
    height: 280px;
}
```

```
/*Estilo para la imagen que se encuentra dentro de la sección*/
#imagen {
    margin: 0px;
    width: 380px;
    height: 200px;
    text-align: center;
}

/*Estilo para el titulo dentro de la sección*/
#subtitulo2 {
    font-family: Tahoma, Geneva, sans-serif;
    font-size: 25px;
    color: #039;
}

/*Estilo para el texto del lado derecho*/
#ladoderecho {
    margin: -5px 20px 0px 0px;
    float: right;
    width: 65%;
    text-align: justify;
    color: #666;
}

/*Estilo para el pie del documento */
#pie {
    text-align: center;
}

/*Estilo para todo los párrafos del documento*/
p {
    font-family: Tahoma;
    font-size: 11pt;
}

/*Estilo para el texto mostrado en el lado izquierdo NOSOTROS*/
#ladoizquierdo {
    margin: -5px 20px 0px 0px;
    float: left;
    width: 65%;
    font-size: 11px;
    text-align: justify;
    color: #666;
}
```

```
/* */

#columna {
    margin: -5px 20px 0px 0px;
    float: left;
    width: 30%;
    color: #666;
}

#seccionextensa {
    width: 1050px;
    padding: 20px 40px;
    background: #FFF;
    height: auto;
}

#centrado {
    text-align: center;
}

#lista li {
    font-size: 11px;
}
```

2. Cree el archivo **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='../css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header id='logo'>
            <h1><a href='principal.html'>Agencia de Empleos</a></h1>
            <p id='subtitulo'>Bienvenidos a nuestro sitio web</p>
        </header>
        <section>
            <nav id='menu'>
                <ul>
                    <li><a href='index.html' title='Pagina Principal'
                        HREFLANG='es-pe' ACCESSKEY='I'>Inicio</a></li>
                    <li><a href='nosotros.html' title='Conocenos'
                        HREFLANG='es-pe' ACCESSKEY='N'>Nosotros</a></li>
```

```
            <li><a href='servicios.html' title='Nuestros Servicios'
                HREFLANG='es-pe' ACCESSKEY='S'>Servicios</a></li>
            <li><a href='recursos.html' title='Recursos'
                HREFLANG='es-pe' ACCESSKEY='R'>Recursos</a></li>
            <li><a href='mailto:informa@agencia.com' title='Escribanos'
                HREFLANG='es-pe' ACCESSKEY='C'>Contáctenos</a></li>
        </ul>
    </nav>
</section>
<section id='seccion'>
    <header id='subtitulo2'>Nuestra Agencia de Empleo</header>
    <img src='../img/empleados.png' id='imagen'/>
    <aside id='ladoderecho'>
        <p>
        Acá encontrarás la forma más fácil de encontrar a la trabajadora del
        hogar que necesitas: para todo servicio, para el cuidado de niño y
        para la atención de adultos mayores. Te asesoramos en la búsqueda,
        selección y contrato de trabajadoras del hogar de acuerdo a tus
        requerimientos.
        Nuestras trabajadoras cuentan con sus documentos en regla, voluntad
        de hacer bien su trabajo y han asistido a un taller de capacitación
        laboral por la Asociación Grupo de Trabajo Redes, organización con veinte
        años de experiencia en servicio doméstico.
        </br></br>
        Nuestro servicio incluye el seguimiento del desempeño de la
        trabajadora, garantía de reemplazo durante dos meses y orientación
        legal permanente.
        En la Agencia de Empleos trabajamos de acuerdo a la Ley 27986, Ley de
        los Trabajadores del Hogar. Está prohibido que niños y niñas menores de
        14 años trabajen en servicio doméstico.
        </p>
        <p>La Empresa.</p>
    </aside>
</section>
<footer id='pie'><p>Todos los derechos reservados-@2019</p></footer>
</body>
</html>
```

El resultado será:

Index.html



3. Cree el archivo **nosotros.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset='UTF-8'>
    <link href='../css/estilo.css' rel='stylesheet' type='text/css'/>
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>
</head>
<body>
    <header id='logo'>
        <h1><a href='principal.html'>Agencia de Empleos</a></h1>
        <p id='subtitulo'>Bienvenidos a nuestro sitio web</p>
    </header>
    <section>
        <nav id='menu'>
            <ul>
                <li><a href='index.html' title='Pagina Principal'
                        HREFLANG='es-pe' ACCESSKEY='I'>Inicio</a></li>
                <li><a href='nosotros.html' title='Conocenos'
                        HREFLANG='es-pe' ACCESSKEY='N'>Nosotros</a></li>
                <li><a href='servicios.html' title='Nuestros Servicios'
                        HREFLANG='es-pe' ACCESSKEY='S'>Servicios</a></li>
                <li><a href='recursos.html' title='Recursos'
                        HREFLANG='es-pe' ACCESSKEY='R'>Recursos</a></li>
                <li><a href='mailto:informa@agencia.com' title='Escribanos'
                        HREFLANG='es-pe' ACCESSKEY='C'>Contáctenos</a></li>
            </ul>
        </nav>
    </section>
```

```
<section id='seccion'>
    <header id='subtitulo2'>Nosotros</header>
    <aside id='ladoizquierdo'>
        <p>
        Somos una empresa social creada por la Asociación Grupo de Trabajo
        Redes. Bajo el lema compromiso y responsabilidad, te ofrecemos la
        mejor alternativa en la búsqueda, selección y contratación de una
        trabajadora del hogar responsable y eficiente a través de un servicio
        innovador por su calidez, garantía y procedimientos enfocados en
        satisfacer tus necesidades y crear relaciones laborales largas y exitosas.
        </br></br>
        La Agencia de Empleos está inscrita en el Registro Nacional de
        Agencias Privadas de Empleo del Ministerio de Trabajo y Promoción
        del Empleo. Asimismo, contamos con Licencia de Funcionamiento de la
        Municipalidad y de Defensa Civil. Esto nos permite ofrecerte nuestros
        servicios con total seguridad y legalidad.
        </p>
      </aside>
      <img src='../img/nosotros.png' id='imagen'/>
    </section>
    <footer id='pie'><p>Todos los derechos reservados-@2019</p></footer>
  </body>
</html>
```

El resultado será:

nosotros.html



4. Cree el archivo **servicios.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='../css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
<body>
    <header id='logo'>
        <h1><a href='principal.html'>Agencia de Empleos</a></h1>
        <p id='subtitulo'>Bienvenidos a nuestro sitio web</p>
    </header>
    <section>
        <nav id='menu'>
            <ul>
                <li><a href='index.html' title='Pagina Principal'
                        HREFLANG='es-pe' ACCESSKEY='I'>Inicio</a></li>
                <li><a href='nosotros.html' title='Conocenos'
                        HREFLANG='es-pe' ACCESSKEY='N'>Nosotros</a></li>
                <li><a href='servicios.html' title='Nuestros Servicios'
                        HREFLANG='es-pe' ACCESSKEY='S'>Servicios</a></li>
                <li><a href='recursos.html' title='Recursos'
                        HREFLANG='es-pe' ACCESSKEY='R'>Recursos</a></li>
                <li><a href='mailto:informa@agencia.com' title='Escribanos'
                        HREFLANG='es-pe' ACCESSKEY='C'>Contáctenos</a></li>
            </ul>
        </nav>
    </section>
    <section id='seccionServicios'>
        <header id='subtitulo2'>Nuestros Servicios</header>
        <aside id='columna'>
            <ul>
                <li id='li'>
                    <p><strong>Niñeras</strong></p>
                    <p>
                    Nuestro servicio de Niñeras ayuda a los padres de familia que
                    necesitan dejar en manos responsables a sus hijos, ya sea
                    porque se van a trabajar antes de que sus niños se vayan al
                    colegio o porque son niños muy pequeños y necesitan de alguien
                    para atenderlos durante todo el día, brindándoles no solo
                    seguridad o compañía, sino aprovechar el tiempo juntos para
                    reforzar sus habilidades con juegos educativos.
                    </p>
    </li>
```

```
            <li id='li'>
                <strong>Pagos mensuales: </strong>
                de 22 años  S/. 1200
                de 52 años  S/. 1300
                de 26 años  S/. 1200
                de 36 años  S/. 950
                de 18 años  S/. 900
            </li>
        </ul>
    </aside>
    <aside id='columna'>
        <ul>
            <li id='li'>
                <p><strong>Cocineras</strong></p>
                <p>
                Múltiples recetas hechas de la forma más tradicional, ¡como la
                abuela…! Guisos, Menestras, Comida Criolla, Postres, Pastas,
                etc. Instrucciones sobre: Conservación, manipulación y
                descongelado de alimentos. Le presentaremos una lista con más
                de 50 recetas para pueda elejir 5 por cada día del curso.
                </p>
            </li>
            <li id='li'>
                <strong>Curso:</strong>
                de 8 horas (12 recetas) | Nuestra recomendación sería impartirlas
                durante: 2 semanas / 1 día a la semana / 4 horas semanales. De
                10 a 14 h., cuando los niños estén en el colegio.
            </li>
        </ul>
    </aside>
    <aside id='columna'>
        <ul>
            <li id='li'>
                <p><strong>Empleadas del Hogar</strong></p>
                <p>
                Organizamos el tiempo diario de trabajo de la empleada del
                hogar, dándole las pautas adecuadas para conseguir que adopte
                una misma rutina de trabajo para el día a día. De esta manera,
                conseguimos que optimice al máximo el tiempo del que dispone
                para realizar un trabajo, profesional y correcto.
                </p>
    </li>
    <li id='li'>
        <strong>Servicios:</strong>
        Camas perfectas. Cobertura de cama. Cuartos de baño. Limpieza
        de polvo. Aspirado y fregado de suelos. Cambio de sábanas y
```

```
                    toallas. Limpieza de cocina. Aspirador: funciones Utilización
                    de utensilios y productos de limpieza.
                </li>
            </ul>
        </aside>
    </section>
    <footer id='pie'><p>Todos los derechos reservados-@2019</p></footer>
</body>
</html>
```

El resultado será:

servicios.html



5. Cree el archivo **recursos.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='../css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header id='logo'>
            <h1><a href='principal.html' name='arriba'>
                Agencia de Empleos
                </a>
            </h1>
            <p id='subtitulo'>Bienvenidos a nuestro sitio web</p>
        </header>
```

```
<section>
    <nav id='menu'>
        <ul>
            <li><a href='index.html' title='Pagina Principal'
                HREFLANG='es-pe' ACCESSKEY='I'>Inicio</a></li>
            <li><a href='nosotros.html' title='Conocenos'
                HREFLANG='es-pe' ACCESSKEY='N'>Nosotros</a></li>
            <li><a href='servicios.html' title='Nuestros Servicios'
                HREFLANG='es-pe' ACCESSKEY='S'>Servicios</a></li>
            <li><a href='recursos.html' title='Recursos'
                HREFLANG='es-pe' ACCESSKEY='R'>Recursos</a></li>
            <li><a href='mailto:informa@agencia.com' title='Escribanos'
                HREFLANG='es-pe' ACCESSKEY='C'>Contáctenos</a></li>
        </ul>
    </nav>
</section>
<section id='seccion'>
    <header id='subtitulo2'>Recursos</header>
    <aside id='columna'>
        <img src='../img/recurso1.jpg' id='imagen'/>
        <p><a href='#biblicos'>Mensajes Biblicos</a></p>
    </aside>
    <aside id='columna'>
        <img src='../img/recurso2.jpg' id='imagen'/>
        <p><a href='#evaluacion'>Evaluación de Personal</a></p>
    </aside>
    <aside id='columna'>
        <img src='../img/recurso3.jpg' id='imagen'/>
        <p><a href='#consejos'>Consejos Útiles</a></p>
    </aside>
</section>
<br/><br/>
<section id='seccionextensa'>
    <article>
        <p><strong><a name='biblicos'>Mensajes Biblicos</a></strong></p>
        <p>
        Es un mandato de dios 20:9 Seis días trabajarás, y harás toda tu
        obra; es un mandato de dios, 2 Timoteo 2:6: El labrador, para
        participar de los frutos, debe TRABAJAR primero Colosenses 3:22:
        SIERVOS, obedeced en todo a vuestros amos terrenales, no sirviendo
        al ojo, como los que quieren agradar a los hombres, sino con corazón
        sincero, temiendo a Dios. Colosenses 4:1: Amos, haced lo que es justo
        y recto con vuestros SIERVOS, sabiendo que también vosotros tenéis un
        Amo en los cielos. El diccionario de la real academia española dice:
```

```
</br>
TRABAJO.- Acción y efecto de trabajar. 2. m. Ocupación retribuida.
Trabajar 1. intr. Ocuparse en cualquier actividad física o
intelectual.
2. intr. Tener una ocupación remunerada en una empresa, una institución,
etc. 3. intr. Ejercer determinada profesión u oficio.
EMPLEADOR,- empleador, ra. 1. adj. Que emplea. Empleada.- Persona que
por un pago o sueldo desempeña los trabajos domésticos o ayuda en
ellos. pago.- pago.(Del lat. salarĭum, de sal, sal).
        </p>
        <a href='#arriba'>Subir</a>
    </article>
</section>
<section id='seccionextensa'>
    <article>
        <p><strong><a name='evaluacion'>Evaluación Personal
            </a>
            </strong>
        </p>
        <p>
        Para garantizar un alto nivel de satisfacción del trabajo de
        nuestra empleada de hogar, intervienen múltiples factores, pero
        por nuestra experiencia las principales recomendaciones serían
        las siguientes:
<ul>
    <li>
        <p>
        Ofrecerle las instrucciones lo más detalladas posible durante
        los primeros días sobre las costumbres personales en su
        domicilio y con su familia, horarios de las distintas comidas,
        etc ( TEMA 2) y corregirle lo antes posible aquellos puntos que
        no esté realizando a su gusto.
        </p>
    </li>
    <li>
        <p>
        Recompensarle con el pago de un buen pago, respetar tanto sus
        días libres ó festivos, como sus horas de descanso y cumplir
        con los derechos recogidos en el actual convenio de servicio
        doméstico (que les detallamos en este capítulo).
        </p>
</li>
<li>
        <p>
        Tenemos que hacer también una reflexión sobre el nivel de
        responsabilidad que realmente les vamos a exigir, son
        trabajadoras de apoyo en la casa y obviamente esperaremos de
        ellas responsabilidad, iniciativa y colaboración, pero en
```

```
                algunos casos creo que las expectativas son demasiado altas
                desde el primer día de trabajo.
                </p>
            </li>
        </ul>
        </p>
        <a href='#arriba'>Subir</a>
    </article>
</section>
<section id='seccionextensa'>
    <article>
        <p><strong><a name='consejos'>Consejos</a></strong></p>
        <p>
        Antes de entrevistar a las posibles candidatas, debemos asegurarnos
        a través de una primera conversación telefónica que realmente ellas
        estén interesadas en trabajar en la modalidad que usted lo requiera;
        externa o interna.
        </br>
        Las trabajadoras de este sector se dividen en estas dos modalidades.
        Como trabajadoras externas podemos encontrar personal más formado
        (la gran mayoría han sido internas durante años), conocen mejor sus
        obligaciones y realizarán un trabajo más mecanizado. </br>
        La mayoría de las trabajadoras interesadas en trabajar como internas,
        son habitualmente las que llevan poco tiempo en Lima y no suelen
        disponer de demasiada experiencia y referencias, prefieren trabajar
        como internas por seguridad y conveniencia para ahorrar el máximo
        y no incrementar sus gastos con alquileres de habitaciones, manutención
        y transporte y durarán el tiempo que consideren necesario para
        estabilizar el pago de las deudas adquiridas para su desplazamiento a
        Lima y ahorrar lo suficiente para, en la mayoría de los casos, traerse
        a su familia a Lima o volverse a su país. Por ello, debemos de aceptar
        que la duración media de una trabajadora interna suele estar en 2
        años, y siempre que se encuentren valoradas y cuidadas por sus empleadores.
        </p>
        <a href='#arriba'>Subir</a>
    </article>
</section>
<footer id='pie'><p>Todos los derechos reservados-@2019</p></footer>
</body>
</html>
```

El resultado será:

recursos.html



Mensajes Bíblicos

Es un mandato de dios 20:9 Seis días trabajarás, y harás toda tu obra; es un mandato de dios, 2 Timoteo 2:6: El labrador, para participar de los frutos, debe TRABAJAR primero Colosenses 3:22: SIERVOS, obedeced en todo a vuestros amos terrenales; no sirviendo al ojo, como los que quieren agradar a los hombres, sino con corazón sincero, temiendo a Dios. Colosenses 4:1: Amos, haced lo que es justo y recto con vuestros SIERVOS, sabiendo que también vosotros tenéis un Amo en los cielos. El diccionario de la real academia española dice:
TRABAJO.- Acción y efecto de trabajar. 2. m. Ocupación remunerada. Trabajar 1. intr. Ocuparse en cualquier actividad física o intelectual. 2. intr. Tener una ocupación remunerada en una empresa, una institución, etc. 3. intr. Ejercer determinada profesión u oficio. EMPLEADOR,- empleador, ra. 1. adj. Que emplea. Empleada.- Persona que por un pago o sueldo desempeña los trabajos domésticos o ayuda en ellos. pago.- pago (Del lat. salarium; de sal, sal).

Subir

Evaluación Personal

Para garantizar un alto nivel de satisfacción del trabajo de nuestra empleada de hogar, intervienen múltiples factores, pero por nuestra experiencia las principales recomendaciones serían las siguientes:

- Ofrecerle las instrucciones lo más detalladas posible durante los primeros días sobre las costumbres personales en su domicilio y con su familia, horarios de las distintas comidas, etc (TIPS 2) y corregirle lo antes posible aquellos puntos que no esté realizando a su gusto.
- Recompensarle con el pago de un buen pago, respetar tanto sus días libres o festivos, como sus horas de descanso y cumplir con los derechos recogidos en el actual convenio de servicio doméstico (que les detallamos en este capítulo).
- Tenemos que hacer también una reflexión sobre el nivel de responsabilidad que realmente les vamos a exigir, son trabajadoras de apoyo en la casa y dependiendo exigencias de ellas responsabilidad, iniciativa y colaboración, pero en algunos casos creo que las expectativas son demasiado altas desde el primer día de trabajo.

Subir

Consejos

Antes de entrevistar a las posibles candidatas, debemos asegurarnos a través de una primera conversación telefónica que realmente ellas están interesadas en trabajar en la modalidad que usted lo requiera: externa o interna.
Las trabajadoras de este sector se dividen en estas dos modalidades. Como trabajadoras externas podemos encontrar personal más formado (la gran mayoría han sido internas durante años), conocen mejor sus obligaciones y realizarán un trabajo más meticuloso.
La mayoría de las trabajadoras interesadas en trabajar como internas, son habitualmente las que llevan poco tiempo en Lima y no suelen disponer de demasiada experiencia y referencias, prefieren trabajar como internas por seguridad y conveniencia para ahorrar el alquiler y no incrementar sus gastos con alquileres de habitaciones, manutención y transporte y durarán el tiempo que consideren necesario para estabilizar el pago de las deudas adquiridas para su desplazamiento a Lima y ahorrar lo suficiente para, en la mayoría de los casos, traerse a su familia a Lima o volverse a su país. Por ello, debemos de aceptar que la duración media de una trabajadora interna suele estar en 2 años, y siempre que se encuentren valoradas y cuidadas por sus empleadores.

Subir

Todos los derechos reservados - @2019

CAPÍTULO 7

Imágenes, audio y video



Competencias

- Selecciona imágenes y las agrega a un documento web.
- Implementa imágenes como enlaces y añade archivos de audio y video a un documento web.

Contenido

7.1 Seleccionando de manera correcta las imágenes
7.2 Agregando imágenes con la etiqueta <img>
7.3 Usando imágenes como enlaces
7.4 Agregando videos de YouTube
7.5 Agregando audio y video con HTML5

7.1 Seleccionando de manera correcta las imágenes



Cuando se desarrolla una página web, uno de los primeros problemas que se presenta es elegir las imágenes que contendrá y el tipo y la calidad de las mismas. Por ello, se debe conocer todos estos aspectos antes de usar cualquier tipo de imagen, porque no sería recomendable ir simplemente a la sección de imágenes de Google, descargar las imágenes necesarias y colocarlas en el documento web. Hay muchas razones para no realizar esta actividad una de ellas es el problema del propietario o licencia de la imagen; claro está que en algunos países esto aún no ha tomado mucha fuerza, pero lo llegará a tener, es un hecho; por ahora, se observan páginas web que usan imágenes copiadas de otras; ahora el lector decidirá si las emplea o no.

Si no se cuenta con imágenes adecuadas y solo se inicia la búsqueda en la internet se debe tener en cuenta los siguientes puntos:

- **Revisar la licencia o derechos de autor:** Primero se debe revisar si la imagen cuenta con la firma del autor o la licencia de la imagen de parte de una empresa u organización. Si es así, solicitar el permiso necesario para su uso, también se debe considerar que hay páginas que ofrecen imágenes gratuitas (previa suscripción) como, por ejemplo, <http://www.sxc.hu/>.

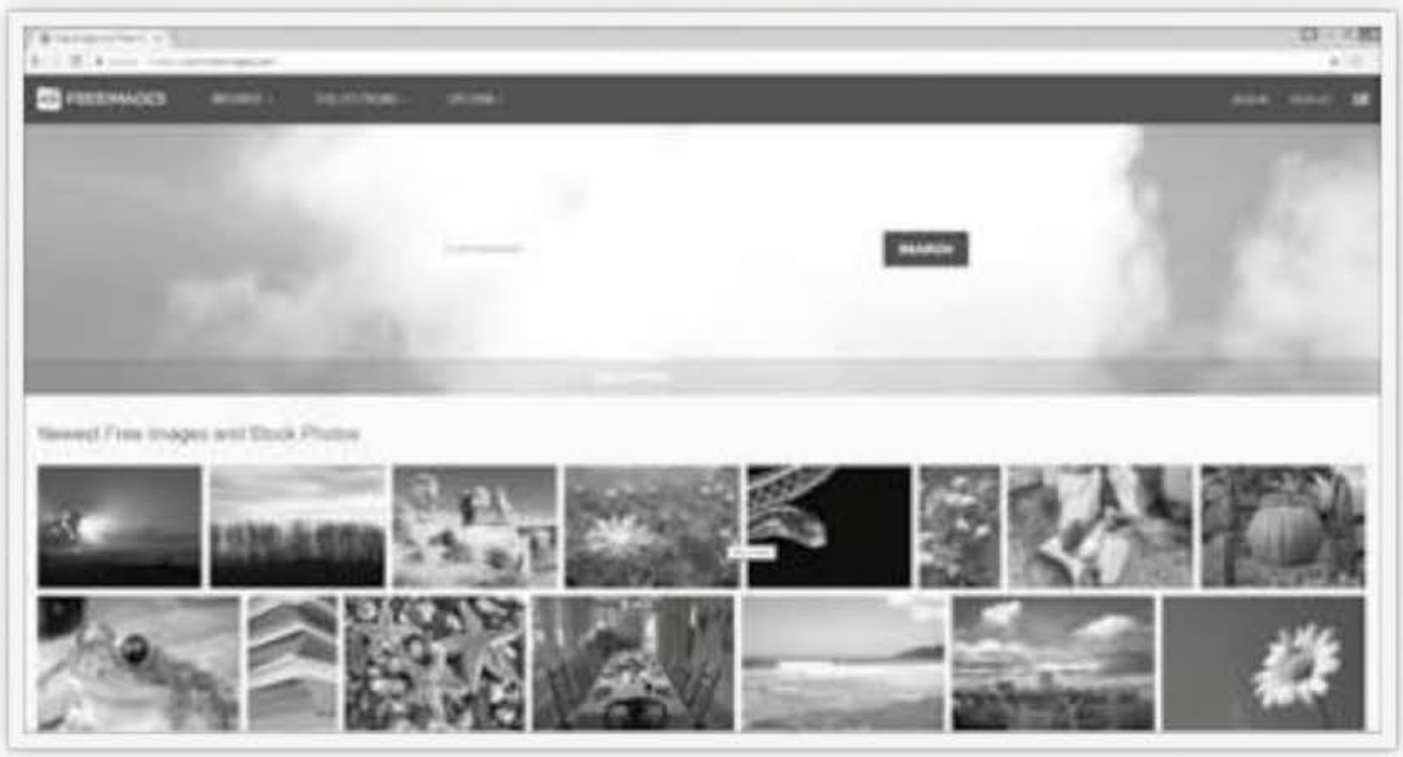


Figura 7.1 Repositorio de imágenes.
Fuente: <http://www.sxc.hu/>.

- **Compartir imágenes en las redes sociales:** Facebook, Twitter y muchas redes sociales ponen a nuestra libre disposición la opción **Comparte** o **Compartir**. Si las imágenes se encuentran bajo un usuario, implicaría que se conoce al autor y, por lo tanto, que el trámite de propiedad es mucho más rápido.



- **Respetar la autoría mencionando al autor:** Si se va a usar imágenes que no son propias, se debe mencionar el nombre del fotógrafo o la empresa dueña de la imagen, previa autorización de la misma. Se debe tener en cuenta que no solo se habla de imágenes estáticas, sino también de videos, GIF, dibujos, etc.

› **Referenciar la imagen a su lugar de origen:** Se recomienda siempre mencionar de dónde se obtuvo dicha imagen y así evitar problemas legales. Véase una URL que muestra una noticia relacionada a Bill Gates: <http://diarioti.com/descontento-entre-accionistas-de-microsoft-por-la-reorganizacion-gerencial/74151>.



Figura 7.2 Modelo web con imagen referenciada.
Fuente: <http://diarioti.com/descontento-entre-accionistas-de-microsoft-por-la-reorganizacion-gerencial/74151> .

El artículo usa la imagen de Bill Gates (multimillonario estadounidense). Ahora bien, hay que imaginar cuántos fotógrafos habrán tomado una foto a este personaje, por lo que, para cumplir con las formas correctas, los redactores anotaron lo siguiente al final de este artículo:

> Microsoft se negó a comentar los comentarios. La cotización de las acciones de Microsoft mantuvo inalterable después del nombramiento de Satya Nadella como nuevo CEO.
>
> El propio Bill Gates es el cuarto mayor inversionista de la empresa, con una participación del 4% de sus acciones.
>
> ***Fotografía de Bill Gates © Russell Watkins, Department for International Development.***

La especificación que se realiza sobre la imagen de Bill Gates dice: "Russell Watkins, Departamento for International Development", donde se menciona los datos del fotógrafo y en qué empresa labora.

7.1.1 *Imágenes GIF*

Es un tipo de codificación para gráficos muy usado sobre todo en Internet para la implementación de páginas web. Su gran uso se debe a que es un formato comprimido y, por lo tanto, su tamaño también es pequeño, por lo que es ideal para ganar



tiempo cuando carga un documento web. Hay que tener en cuenta que las imágenes GIF tienen buena resolución cuando no se modifica el tamaño original de la imagen.

› **Características más importantes**

» GIF utiliza un protocolo de compresión llamado LZW para agrupar datos idénticos.
» Ha sido diseñado específicamente para comprimir imágenes digitales y ser portable frente a cualquier dispositivo que permita visualizar un documento web.
» Reduce la paleta de colores a 256 colores como máximo, lo que hace que no tenga una muy buena resolución, pero esta desventaja es compensada con el poco tamaño que ocupa.
» El formato GIF no tiene licencia de uso, gracias a la expiración de la patente LZW.
» Admite gamas de menor número de colores y esto permite optimizar el tamaño del archivo que contiene la imagen.
» Un archivo GIF puede contener varias imágenes dentro del mismo archivo creando así animaciones de poca resolución, pero que son suficientes para un documento web.
» No es recomendable usar GIF para fotografías de cierta calidad ni muestras originales, ya que el color real o verdadero utiliza una paleta de más de 256 colores.

7.1.2 *Imágenes JPG*



JPG es uno de los formatos más conocidos para la compresión de fotografías digitales. Es ahora uno de los principales tipos de imágenes que se usan para la web. Tiene un soporte de color verdadero, es decir, de 24 bits. JPG ofrece imágenes con una gama de colores de hasta 16 millones de colores.

Hoy en día las cámaras digitales suelen almacenar las imágenes en formato JPG de forma predeterminada. Debido al nivel de compresión que una imagen JPG puede tener es que se puede alterar la calidad de la resolución, escogiendo entre diferentes niveles de compresión.

› **Características más importantes**

» JPG permite ocupar poco espacio de uso para la web o para envío de *e-mail*.
» JPEG puede comprimir una imagen hasta solo un 10% de su tamaño original, sin que el ojo humano pueda percibir grandes diferencias.
» Cada vez que una imagen JPG es modificada o editada sufre una pérdida de calidad en su resolución.
» No permite almacenar varias imágenes que pueden generar una animación.

7.1.3 *Imágenes de alta calidad en el buscador Google*

Si se necesitan imágenes de alta calidad, como recursos de documentos web, se puede optar por la búsqueda que se realiza en Google. Para esto, a continuación, se muestran los pasos.

Ejemplo 1

Pasos:

1. Ingrese la siguiente URL en su navegador <http://www.google.com>.
2. Seleccione **Imágenes** desde el menú lateral.



3. Ahora ingrese el nombre de la imagen a buscar.



Se sugieren los siguientes textos:

» Wallpaper blank
» Wallpaper orange
» Wallpaper blue
» Wallpaper sky blue
» Wallpaper brown

4. Para buscar imágenes de alta calidad, seleccione **Herramientas** y elija en **Tamaño** o **Tipo**, las opciones que crea conveniente.



5. Las imágenes se mostrarán de la siguiente manera:



6. Para guardar una imagen, primero, seleccione la imagen haciendo clic sobre ella.



7. Luego, haga clic en el botón **Ver imagen** para visualizar solo la imagen a guardar.

8. Haga clic derecho sobre la imagen y luego seleccione **Guardar imagen como...**

7.1.4 *Imágenes PNG*



Son también conocidas como gráficos de red portátiles. Es un formato de archivos de gráficos de mapa de bits. Fue desarrollado en el año 1995 como una alternativa gratuita al formato GIF cuando este era patentada; pero hoy ambos son imágenes libres utilizables para la web.

- Características del formato PNG
  - Permite almacenar imágenes en blanco y negro con una profundidad de color de 16 bits por píxel.
  - En color real tiene una profundidad de color de 48 bits por píxel.
  - También trabaja con imágenes indexadas utilizando una paleta de 256 colores.
  - Soporta la transparencia del canal alfa, es decir, la posibilidad de definir 256 niveles de transparencia, mientras que el formato GIF permite que se defina como transparente solo un color de la paleta.
  - PNG posee una función de entrelazado que permite mostrar la imagen de forma gradual.
  - La compresión que ofrece este formato es compresión sin pérdida de 5 a 25% mejor que la compresión GIF.
  - Véase una comparación entre una imagen JPG y una PNG con transparencia de fondo.

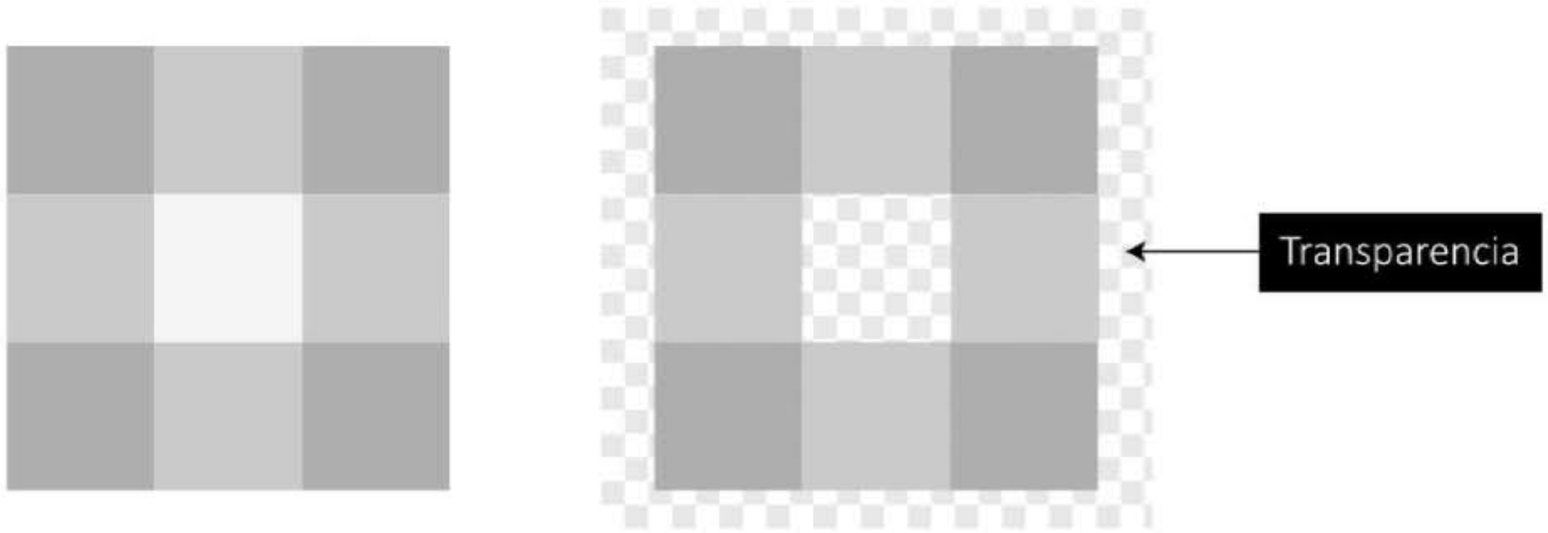


Figura 7.3 Imagen con transparencia PNG.
Fuente: Elaboración propia.

7.2 Agregando imágenes con la etiqueta <img>

Para insertar una imagen dentro de un documento web, hay que realizar la referencia de igual manera que con la etiqueta de enlaces, es decir, todas las imágenes deben estar registradas en un lugar específico y conocido por el diseñador. Entonces, se consideran los siguientes aspectos:

- Se debe tener una carpeta de imágenes en el proyecto web; si no es así, se debe conocer obligatoriamente dónde está guardada.
- Conocer las características que tienen los diferentes tipos de imágenes como JPG, GIF y PNG.
- Y lo más importante que hay que considerar: las imágenes no deben tener un peso mayor a 2MB, pues harían que la página web sea muy lenta y, además, llenaría prontamente nuestro servidor.

Su formato es:

```
<img src='ruta de la imagen' alt='título de la imagen'
height='valor de altura' width='valor de anchura'
ismap usemap='#nombre del mapa'/>
```

7.2.1 *Atributo src*

Permite definir la ubicación de la imagen dentro del proyecto web. Tiene las siguientes variaciones:

- Si la imagen se encuentra dentro de la carpeta **Imagenes** del proyecto web:

```
<img src='Imagenes/banner.jpg'>
```

- Si la imagen se encuentra en la carpeta raíz del proyecto web:

```
<img src='banner.jpg'>
```

- Si la imagen se encuentra en una página web externa también llamada *hotlinking*:

```
<img src='http://www.servidor.com/imagen/banner.jpg'>
```

En esta última variación hay que tener en cuenta que no es muy segura dicha actividad, pues se podría modificar la imagen en el servidor de origen o simplemente podría ser borrada y se corre el riesgo de que nuestra página muestre el siguiente aspecto:



Figura 7.4 Imagen sin contenido.
Fuente: Elaboración propia.

7.2.2 *Atributo alt*

Permite especificar un nombre alternativo a la imagen. Se usa de la siguiente forma:

```
<img src='imagenes/banner.jpg' alt='banner de empleados'>
```

7.2.3 *Atributo height y width*

Permite definir el alto y ancho de la imagen, aquí hay que tener en cuenta que la imagen tiene un tamaño original, y si no se especifica el alto y el ancho, la imagen se insertará en el documento web con su tamaño original. Esto genera dos posibilidades: primero, que la imagen no se presente distorsionada, es decir, ni muy alta ni muy ancha; la segunda, que la imagen puede ser muy grande y podría ocupar mucho espacio en el documento web.

Antes de especificar el alto y ancho de la imagen se debe considerar que mediante las hojas de estilo también pueden configurarse esas dimensiones. La ventaja de esta última forma radica en que no se tendría que realizar los cambios a todas las imágenes, pues solo bastaría con modificar los estilos. Se usa de la siguiente forma:

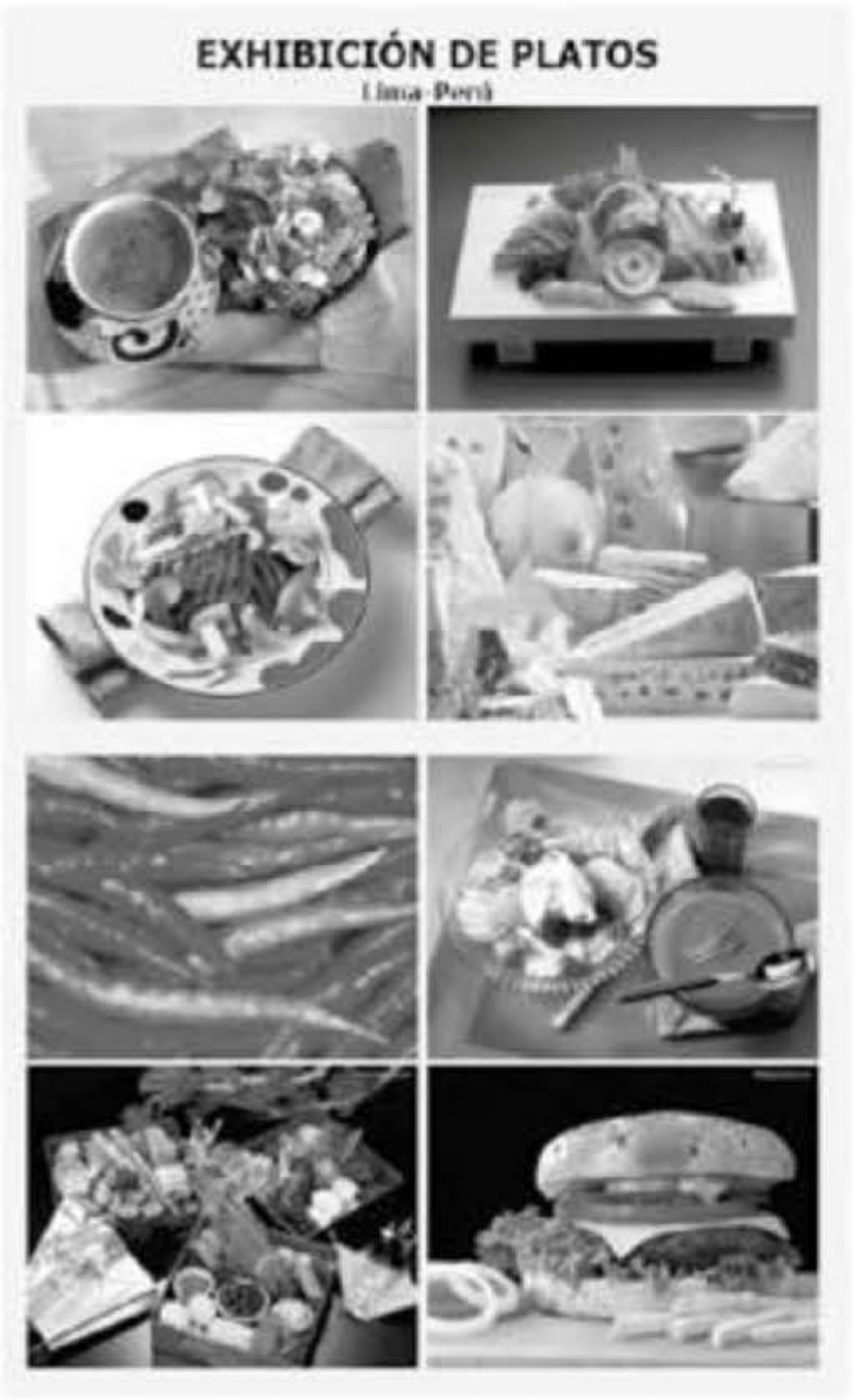
```
<img src='imagenes/banner.jpg' height='150px' width='250px'>
```

Su análogo en script CSS sería:

```
img {
    width:250px;
    height:150px;
}
```

Ejemplo 2

Implemente el siguiente diseño usando la etiqueta **<img>**. El control del tamaño de la imagen se debe realizar en el archivo **estilo.css**.



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
section{
    font-family: 'Tahoma ';
    width: 510px;
    background-color: beige;
    padding: 15px;
}

h1{
    position: static;
    text-align: center;
    border: #663366 10px;
    margin: 0px;
    color:#000;
}
```

```
h4 {
    position:static;
    text-align:center;
    border: #663366 5px;
    color:#000;
    margin: 0px;
}

img {
    position:static;
    width: 250px;
    height:194px;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='estilo.css' rel='stylesheet' type='text/css'/>
        <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section>
            <h1>EXHIBICIÓN DE PLATOS</h1>
            <h4>Lima-Perú</h4>
            <img src='img/imagen1.jpg'/>
            <img src='img/imagen2.jpg'/>
            <img src='img/imagen3.jpg'/>
            <img src='img/imagen4.jpg'/>
            <br><br>
            <img src='img/imagen5.jpg'/>
            <img src='img/imagen6.jpg'/>
            <img src='img/imagen7.jpg'/>
            <img src='img/imagen8.jpg'/>
        </section>
    </body>
</html>
```

7.3 Usando imágenes como enlaces

Muchas veces se recurre al empleo de imágenes como enlaces, es decir, a la forma de poder acceder a cualquier enlace mediante una imagen. Por ejemplo, un menú de opciones con una botonera de imágenes.

Hasta ahora se ha visto que los enlaces se realizan mediante un texto descriptivo. Esta vez se colocará el script para la inserción de imágenes dentro de la etiqueta `<a>`. Para realizar esto se empleará el siguiente formato:

```
<a href='url'>
   <img src='dirección de la imagen'/>
</a>
```

Ejemplo 3

Implemente el siguiente diseño usando la etiqueta `<img>`, donde las imágenes hagan referencia a una página web externa.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body{
    padding:5px;
    margin:10px;
    font-family: Tahoma;
    font-size: 13px;
}

#enlace1 {
    position: absolute;
    padding: 15px;
    color: #000;
    top: 10px;
    left: 80px;
}

#enlace2 {
    position: absolute;
    padding: 15px;
    color: #000;
    top: 10px;
    left: 320px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv='content-type' content='text/html; charset=UTF-8'>
      <link href='estilo.css' rel='stylesheet' type='text/css'>
   </head>
   <body>
      <section>
         <a href='http://www.google.com.pe'>
            <p id='enlace1'> Ir a Google</p>
            <img src='images/b2.jpg' >
         </a>
         <a href='http://www.facebook.com'>
            <p id='enlace2'> Ir a Facebook</p>
            <img src='images/b3.jpg'>
         </a><br>
      </section>
   </body>
</html>
```

7.3.1 *Definición de mapas*

Hasta ahora se ha visto dos formas de enlace en la web: la primera por medio de textos, es decir, al dar clic sobre el texto esta acción enlaza a una referencia específica, y la segunda por medio de imágenes. Ahora véase la tercera forma llamada enlaces por mapas.

Los mapas de imágenes presentan una serie de enlaces o links en una misma imagen, haciendo que desde una sola imagen pueda direccionarse a varios enlaces web. Parece una idea ilógica, pero más adelante se verá qué cosas sustentan la idea de su uso.

Un mapa es una representación de áreas mediante figuras geométricas en una misma imagen. La idea es que, por cada mapa, se tenga un enlace definido. Sus principales características son:

- La implementación de mapas sobre una imagen tiene el mismo manejo que los enlaces por medio de textos e imágenes.
- Se puede usar los mapas para reconocer secciones de un sitio web desde una sola imagen, por ejemplo, una foto familiar en una página web puede llevarnos a cada uno de los blogs o Facebook de los integrantes.
- Siempre que se implementa un mapa se recomienda acompañarlo de un texto descriptivo para indicarle al usuario a donde le dirigirá el enlace.
- Para reconocer un mapa, solo basta posicionarse dentro del mismo y aparecerá la manito del enlace web.

Ejemplo 4

Si se tiene como objetivo enlazar una URL externa a partir de la selección de un rostro, podría tenerse el siguiente mapeo:



Hay que tener en cuenta que un mapeo de imagen es como una plantilla transparente de forma rectangular, circular o poligonal, que no será visible por el usuario, pero que tendrá un enlace URL sobre una determinada imagen. Para esto, hay que considerar los tamaños de una imagen. Véase la siguiente imagen:



Asúmase que la imagen mostrada tiene un alto de 120px y un ancho de 147px, entonces se tendrían los siguientes valores:

- **Posición inicial de la imagen**: 0,0 ubicada en la parte de la esquina superior izquierda.
- **Posición final de la imagen**: 147,120, donde 147px será el punto para los valores del eje X, y 120px, para los valores del eje Y.

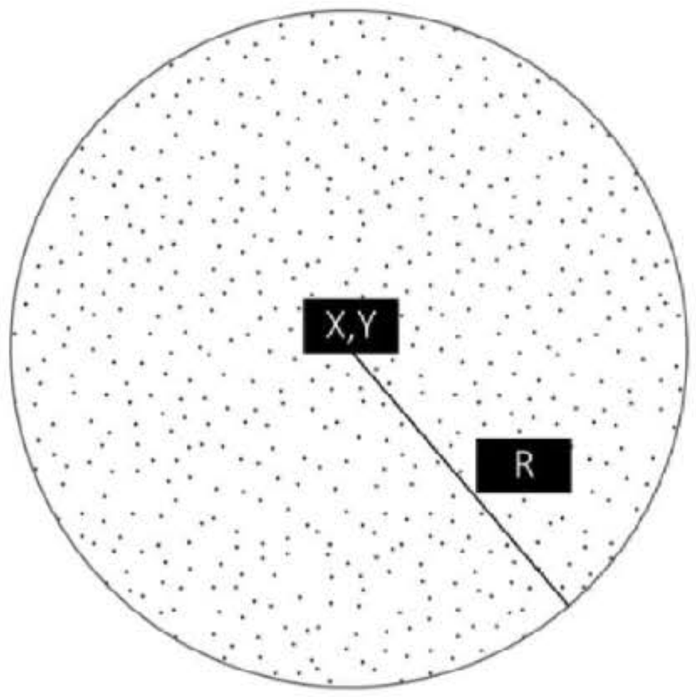
7.3.2 La etiqueta <map>

Se utiliza para definir un mapa que toma forma de polígonos en una imagen. La etiqueta `<map>` contiene un número de elementos `<area>` que define todas las zonas posibles en una imagen, cada zona tiene un enlace que será administrada por el diseñador web.

Para relacionar la imagen con los mapas se usa la propiedad `<name>` de la etiqueta anp; de esta manera la imagen lo asociara mediante su propiedad USEMAP. Veamos los tipos de mapas que podemos seleccionar:

A. Mapa circular

Crea un área circular, que se indica con la coordenada del centro del círculo y el radio.



Véase el formato para un mapa de tipo circular:

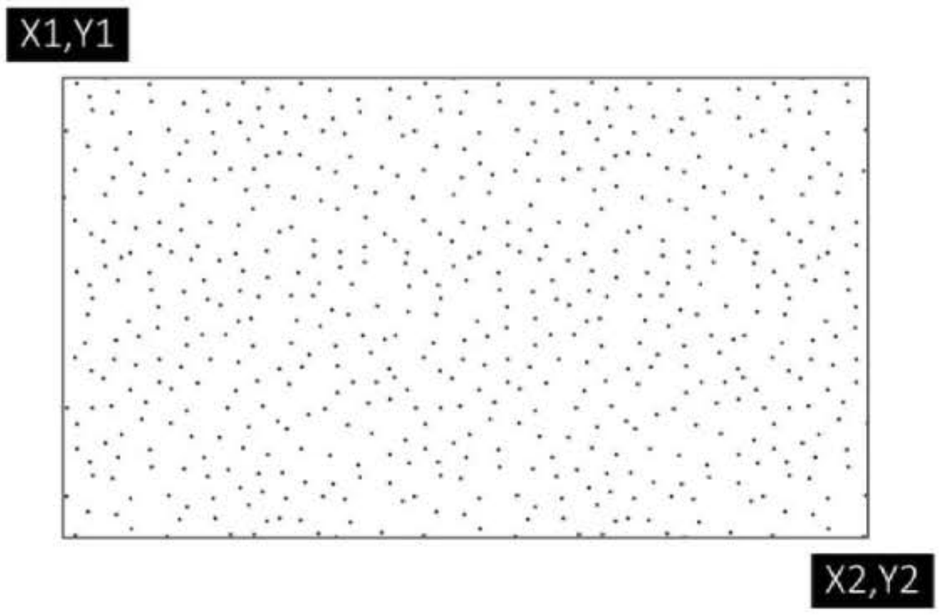
```
<map name='nombre del mapeo'>
   <area shape='circle' coords='x,y,r' href='url'>
</map>
```

Donde:

- **name**: Es el nombre asignado al mapeo. En el futuro definirá la asociación que tiene con la imagen, por tal motivo, siempre se tendrá que asignar un nombre y este no deberá ser repetido para otras imágenes.
- **area shape**: Define el tipo de mapeo. En este caso se usó `circle` para definir un círculo como mapa de imagen.
- **coords**: Esta opción define las coordenadas para el diseño de un mapa de tipo circular, donde X representa la coordenada X de la imagen, e igualmente Y representa la coordenada Y de la imagen, y R es la definición de la radio del óvalo.
- **href**: Es la referencia a un URL de destino. Esto quiere decir que, al seleccionar el mapa, este atributo llevará al usuario al enlace definido en el `href`.

B. Mapa rectangular

Crea un área rectangular basándose en dos pares de coordenadas.



Véase el formato para un mapa de tipo rectangular:

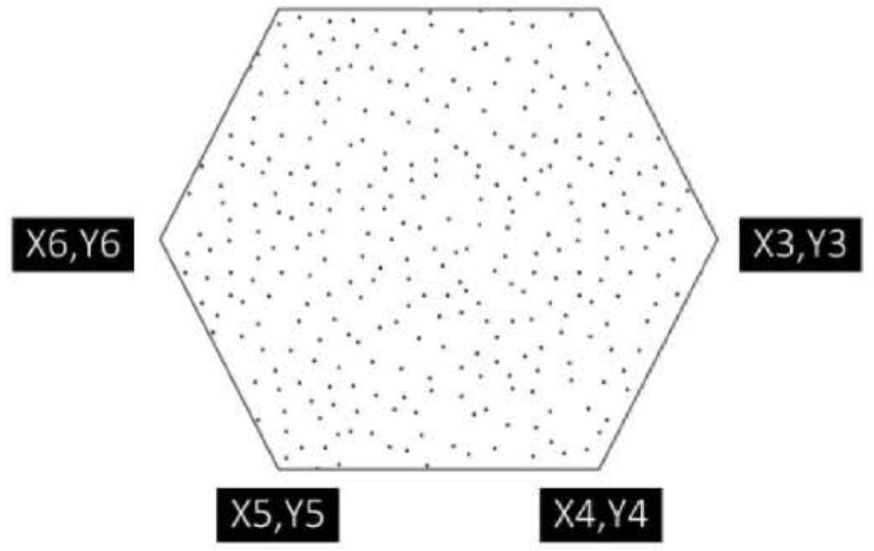
```
<map name='nombre del mapeo'>
   <area shape='rect' coords='x1,y1,x2,y2' href='url'>
</map>
```

Donde:

- **name**: Es el nombre asignado al mapeo. Pero si se está haciendo referencia a una imagen que ya tenía un mapeo, entonces, tendrá el mismo nombre, ya que por imagen solo habrá un nombre de mapa.
- **area shape**: Define el tipo de mapeo. En este caso se usó `rect` para definir un rectángulo como mapa de imagen.
- **coords**: Esta opción define las coordenadas para el diseño de un mapa de tipo rectangular, donde X1, Y1 representa las coordenadas x,y de la esquina superior izquierda de la imagen, mientras que X2, Y2 representa las coordenadas x,y de la esquina inferior derecha de la imagen.
- **href**: Es la referencia a un URL de destino. Esto quiere decir que, al seleccionar el mapa, este atributo llevará al usuario al enlace definido en el `href`.

c. Mapa poligonal

Crea un área poligonal de N lados.



Véase el formato para un mapa de tipo poligonal:

```
<map name='nombre del mapeo'>
   <area shape='poly' coords='x1,y1,x2,y2,x3,y3,x4,y4,x5,y5,x6,y6' href='url'>
</map>
```

Donde:

- **name**: Es el nombre asignado al mapeo. Pero si se hace referencia a una imagen que ya tenía un mapeo, entonces, tendrá el mismo nombre, puesto que por imagen solo habrá un nombre de mapa.
- **area shape**: Define el tipo de mapeo. En este caso se usó `poly` para definir un polígono de N lados como mapa de imagen.

- **coords**: Esta opción define las coordenadas para el diseño de un mapa de tipo polígono, donde X1,Y1 representa las coordenadas del primer punto que forma el polígono; lo mismo se aplica para X2,Y2 y las demás coordenadas.
- **href**: Es la referencia a un URL de destino. Esto quiere decir que, al seleccionar el mapa, este atributo llevará al usuario al enlace definido en el **href**.

7.3.3 *Asociación entre la imagen y los mapas*

Para poder relacionar la imagen con sus respectivos mapas se debe seguir el siguiente formato:

```
<img src='url de la imagen' usemap='#nombre del mapa'/>
```

Donde:

usemap: Permite especificar el nombre del mapa previamente creado dentro de la misma página web. Observe que se debe empezar con el símbolo # obligatoriamente.

Ejemplo 5

Implemente el siguiente diseño web haciendo uso de mapas. Se pide que al seleccionar el rostro de cada uno de los integrantes, esta acción conduzca a su descripción personal.



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
#marco{
    position: absolute;
    color:#000;
    top: 15px;
    left: 448px;
```

```
    width: 1023px;
    height: 120px;
    background-color: beige;
    opacity: 0.8;
}

figure{
    text-align:center;
}

img{
    width:1024px;
    height:728px;
}

#titulo{
    position: absolute;
    font-family: Tahoma;
    font-size:36px;
    font-weight:bold;
    color: #000;
    position:relative;
    top: 15px;
    left: 10px;
}

pre{
    position: absolute;
    font-family:tahoma;
    color: #000;
    position: relative;
    top: 15px;
    left: 610px;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header id='marco'>
            <pre>principal | discografia | fotos | conciertos</pre>
            <span id='titulo'>The Beatles</span>
        </header>
```

```
<figure>
    <map name='beatles'>
        <area shape='circle' coords='769,216,117'
              href='http://es.wikipedia.org/wiki/george_harrison'
              title='george harrison'>
        <area shape='circle' coords='770,417,96'
              href='http://es.wikipedia.org/wiki/john_lennon'
              title='jhon lenon'>
        <area shape='circle' coords='470,334,110'
              href='http://es.wikipedia.org/wiki/ringo_starr'
              title='ringo starkey'>
        <area shape='circle' coords='181,251,99'
              href='http://es.wikipedia.org/wiki/paul_mccartney'
              title='paul mccartney'>
    </map>
    <img src='img/fondo.jpg' usemap='#beatles'/>
</figure>
</body>
</html>
```

7.4 Agregando videos de YouTube

YouTube hoy en día es el sitio más popular de hospedaje de videos del planeta; por lo tanto, se debe conocer cómo insertar un video dentro de un documento web por medio de un enlace desde la página web cliente hacia YouTube. Este último ofrece un código que puede ser pegado en el documento web.

HTML5 y YouTube están asociados perfectamente para esta actividad que también es llamada código embebido.

YouTube ha llegado hasta donde está, gracias a la portabilidad que ofrece en sus videos. Estos podrán ser insertados en blogs, foros y lógicamente en páginas web.



Figura 7.5 Listado de videos desde Youtube.
Fuente: <www.youtube.com>.

› **Cómo se insertan los parámetros de Youtube**

Antes de enlazar un video se debe obtener el código embebido desde el portal de YouTube. Para ello, hay que seguir los siguientes pasos:

Ejemplo 6

Se mencionan los pasos para obtener el código embebido.

Pasos:

1. Ingrese al portal YouTube desde la URL **<http://www.youtube.com>.**
2. Busque el video de su preferencia. Para este caso, coloque **Jelous Guy** y ejecute el video.



3. Seleccione la opción **Compartir** (**Share**) y luego **Incorporar** (**Embed**).



4. Se muestra el código embebido que permitirá mostrar el video directamente en nuestro documento web, ahora solo haga clic en la opción **Copiar** (**Copy**).

5. Analice el código resultante:

```
<iframe width='560'
    height='315'
    src='https://www.youtube.com/embed/wADRRYNHhOA'
    frameborder='0'
    allow='autoplay;
    encrypted-media'
    allowfullscreen>
</iframe>
```

Donde:

- **width**: Define el ancho del video dentro del documento web.
- **height**: Define la altura del video dentro del documento web.
- **src**: Define la URL del video proveniente de YouTube.
- **frameborder**: Define un borde al marco generado por el código embebido.
- **allowfullscreen**: Habilita la opción para maximizar el video desde el documento web.

Ejemplo 7

Implemente el siguiente diseño web haciendo uso del código embebido que permite visualizar videos desde YouTube.



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
#columna {
    position:relative;
    float:left;
    left:0px;
    width: 350px;
    border-style:solid;
    border-color:#666;
    top:0px;
    text-align:center;
    font-family:Tahoma, Geneva, sans-serif;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <article id='columna'>
            <p>video 01-jealous guy</p>
            <iframe width='300' height='270' src="https://www.youtube.com/
                    embed/wADRRYNHhOA' frameborder='0' allowfullscreen>
            </iframe>
        </article>
        <aside id='columna'>
            <p>video 02-my sweet lord</p>
            <iframe width='300' height='270' src="https://www.youtube.com/
                    embed/8qJTJNfzvr8' frameborder='0' allowfullscreen>
            </iframe>
        </aside>
        <aside id='columna'>
            <p>video 03-say say say</p>
            <iframe width='300' height='270' src="https://www.youtube.com/
                    embed/aLEhh_XpJ-0' frameborder='0' allowfullscreen>
            </iframe>
        </aside>
    </body>
</html>
```

7.5 Agregando audio y video con HTML5

Una de las características más importantes sobre el nuevo HTML es su soporte multimedia, que permite la reproducción de archivos de audio y video con sus propios controles. Antes esto se realizaba con la ayuda de archivos SWF de Flash, pero con la llegada de HTML5 se hace más fácil mediante el uso de las etiquetas **<audio>** y **<video>** que se explicará a continuación.



Figura 7.6 Modelos de reproductor según el sistema operativo.
Fuente: <http://blog.michelledinan.com/12/2011/html5-audio/>.

7.5.1 *La etiqueta <video>*

Permite incorporar un archivo de video y reproducirlo en un documento web. Para esto último se muestra una barra de reproducción. Su formato es:

```
<video
   poster='url de la imagen'
   width='ancho del video'
   height='alto del video'
   controls>
   <source src='url del video' type='especificación del tipo' />
</video>
```

Donde:

- **poster**: Determina la imagen que puede ser visible antes de ejecutar el video.
- **width**: Especifica el ancho del video dentro del documento web.
- **height**: Establece el alto del video dentro del documento web.
- **controls**: Presenta los controles del video para poder ejecutarlo o pararlo.



Para especificar propiamente al archivo de video, se debe usar el siguiente código:

```
<source src='url del video' type='especificación del tipo' />
```

Donde:

- **src:** Especifica el nombre y extensión del video.
- **type:** Detalla el tipo de video que se desea visualizar.

A. Tipos de videos compatibles con HTML5

Existen tres tipos de formatos compatibles con HTML y la mayoría de los navegadores:

- **WebM**
  Este formato se basa en una versión restringida del formato contenedor Matroska. Siempre utiliza el códec de video VP8 y el códec de audio Vorbis. Lo más importante de este tipo es que tienen licencia para todo el mundo de forma gratuita.

- **MP4**
  El formato contenedor MP4 con el códec de video H.264, o bien, el códec de audio AAC o el códec de audio MP3 son nativamente compatibles con Internet Explorer, Safari y Chrome.

- **Ogg**
  El formato contenedor Ogg con el códec de video Theora y el códec de audio Vorbis son compatibles con Gecko (Firefox), Chrome y Opera, a este grupo también se puede agregar a Safari mediante la instalación de un add-on. El formato no es compatible con Internet Explorer.

B. ¿Cómo obtengo videos MP4 desde YouTube?

Existe muchas aplicaciones que permiten descargar videos desde YouTube. En este caso se usará la dirección URL del video mostrado en YouTube.

Ejemplo 8

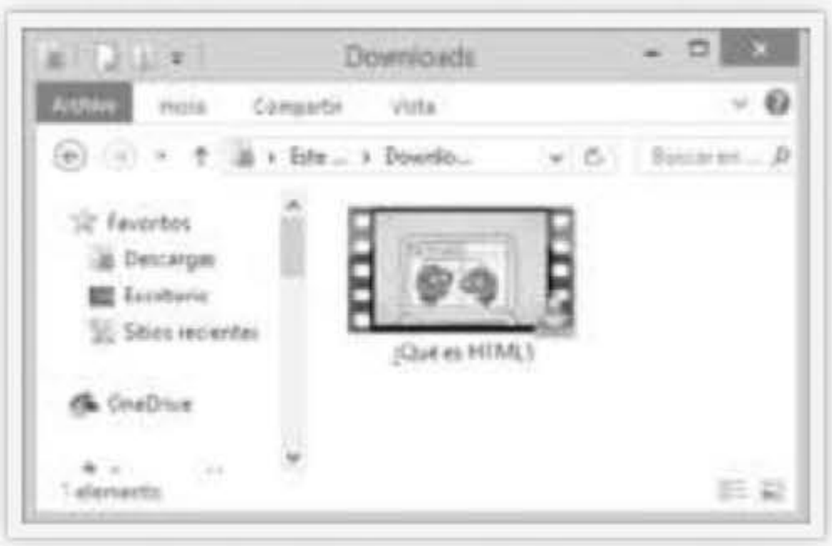
A continuación, se mencionan los pasos para descargar videos de YouTube.

Pasos:

1. Ingrese a YouTube y visualice el video que desea descargar.
2. Copie la dirección URL que se presenta <https://www.youtube.com/watch?v=rI1cxxIcX5E> y péguelo en una nueva ventana web modificando la dirección URL por <https://www.yout.com/watch?v=rI1cxxIcX5E>. Luego se debe mostrar la siguiente web:



3. Seleccione **MP4 (video)** y presione el botón **Record MP4** para iniciar la descarga del video en el formato seleccionado. Una vez terminada la descarga, el video se debe encontrar en la carpeta de descargas.



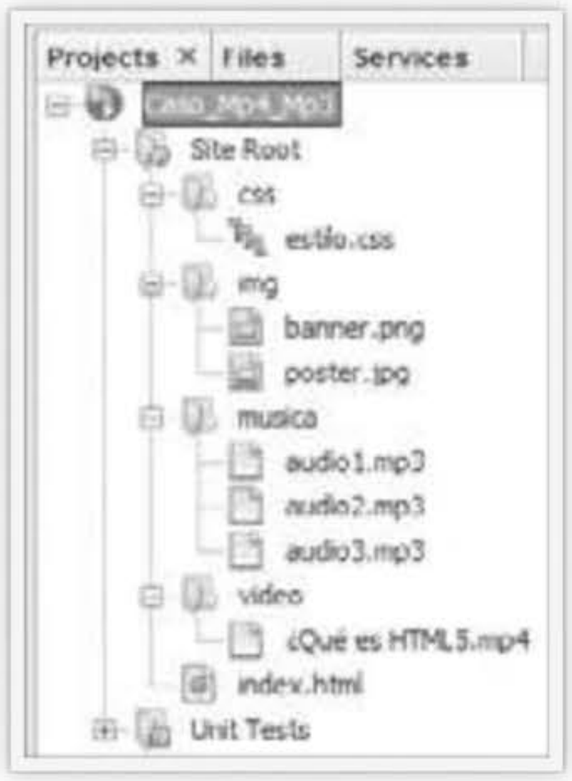
4. Copie el video a la carpeta de su proyecto web.

Ejemplo 9

Implemente el siguiente modelo web haciendo uso de videos y audios de tipo MP4 y MP3, respectivamente.



Antes de iniciar con el desarrollo de la aplicación hay que tener en cuenta que el proyecto debería terminar de la siguiente manera:



Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body {
    color:#FFF;
}

figure {
    text-align:center;
}

img {
    width:1074px;
    height:158px;
}

#franja {
    font-family: Tahoma;
    font-size:14px;
    text-align:center;
    position: relative;
    left:200px;
    top:-50px;
    width:2100px;
    height:30px;
}

pre {
    font-family:tahoma;
    font-size:medium;
    color: #000;
}

#columna {
    position:relative;
    float:left;
    left:417px;
    width: 17%;
    height: 350px;
    top:-50px;
    text-align:justify;
    color:#039;
    padding: 0px 15px 0px 10px;
    font-family:Tahoma, Geneva, sans-serif;
    border: 1px solid;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='UTF-8'>
      <meta name='viewport' content='width=device-width, initial-scale=1.0'>
   </head>
<body>
   <figure><img src='img/banner.png'></figure>
   <section id='franja'>
      <pre>Principal | Misión | Visión | Contactos </pre>
   </section>
   <section >
      <article id='columna'>
         <video poster='img/poster.jpg' width='320' height='240' controls>
            <source src='video/¿Qué es HTML5.mp4' type='video/mp4'>
            Your browser does not support the <code>video</code> tag.
         </video>
      </article>
      <article id='columna'>
         <p>
         <strong>HTML 5</strong><br/><br/>
         HTML5 es la última versión de HTML. El término representa dos
         conceptos diferentes: Se trata de una nueva versión de HTML,
         con nuevos elementos, atributos y comportamientos.
         Contiene un conjunto más amplio de tecnologías que permite a
         los sitios Web y a las aplicaciones ser más diversas y de gran
         alcance. A este conjunto se le llama HTML5 y amigos, a menudo
         reducido a HTML5.
         Diseñado para ser utilizable por todos los desarrolladores de
         Open Web, esta página referencia numerosos recursos sobre las
         tecnologías de HTML5, clasificados en varios grupos según su
         función.
         </p>
</article>
<article id='columna'>
   <p>Audios de Muestra:</p>
<p>audio1.mp3</p>
<audio controls>
   <source src='musica/audio1.mp3' />
</audio>
<p>audio2.mp3</p>
<audio controls>
   <source src='musica/audio2.mp3' />
</audio>
```

```
            <p>audio3.mp3</p>
            <audio controls>
               <source src='musica/audio3.mp3' />
            </audio>
         </article>
      </section>
   </body>
</html>
```

7.5.2 *La etiqueta <audio>*

Desde aquí se puede agregar rápidamente y de una forma muy sencilla pistas de audio con formato MP3 a nuestra página web. En su forma más simple, funciona más o menos de igual manera que el elemento **<img>**, por lo que solo se tiene que utilizar el atributo **src** para vincular el archivo. Su formato es:

```
<audio controls
   autoplay='true'
   loop
   preload >
   <source src='url de la musica' />
</audio>
```

Sus principales propiedades son:

- **Autoplay o reproducción automática**
  Se le puede dar la opción de reproducción automática a los elementos de audio en nuestra página web. Los valores aceptados son **true** o **false**.

- **Loop**
  Es un atributo directo. Al especificarlo, el audio se repetirá indefinidamente.

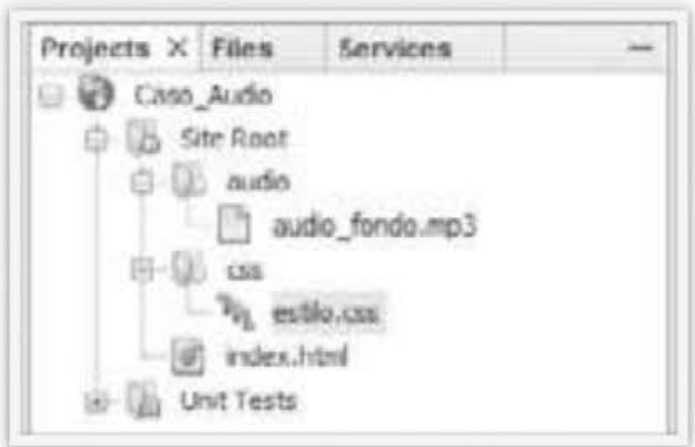
- **Controls**
  El uso de esta propiedad permite al visitante utilizar los controles nativos que cada navegador proporciona para la reproducción del archivo de audio.

Ejemplo 10

Implemente el siguiente modelo web donde se incluya el uso de un audio de tipo MP3 para usarlo como música de fondo. Este audio debe repetirse indefinidamente y de forma automática.



Antes de iniciar con el desarrollo de la aplicación hay que tener en cuenta que nuestro proyecto debería terminar de la siguiente manera:



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    font-family: Tahoma;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <header>
            <p>Música de fondo usando la etiqueta audio</p>
            <audio controls hidden autoplay loop>
                <source src='audio/audio_fondo.mp3' type='audio/mpeg'>
                Your browser does not support the <code>audio</code> tag.
            </audio>
        </header>
    </body>
</html>
```

7.5.3 *La etiqueta <object>*

Se sabe que un documento web no solo contiene textos, imágenes y enlaces. En algún momento del diseño se le pedirá al webmaster implementar al documento web archivos multimedia como los audios MP3, los videos MPG, los videos AVI, entre otros. En el caso de que no se conozca el manejo de alguna etiqueta que controle de manera correcta los archivos multimedia, se puede optar por un código parecido a:

```
<a href='sonido.mp3'>canción</a>
<a href='video.avi'>video</a>
```

Con esta implementación se debe tener cuidado sobre cómo reaccionará el navegador web, puesto que podría mostrar el resultado en una ventana independiente o también darse el caso de que un reproductor podría cargarse inadecuadamente al intentar ejecutar de la mejor manera los archivos multimedia. Esta tecnología se apoya en un tipo de configuración llamado MIME (*Multipurpose Internet Mail Extensions*) que permite intercambiar todo tipo de archivos en Internet como, por ejemplo, audios, videos, imágenes, etc. Las configuraciones que puede realizar MIME son:

CONFIGURACIÓN	DESCRIPCIÓN
image/JPEG	Configuración para imágenes en formato JPG.
audio/MPEG	Configuración para audio en formato MP3.
video/AVI	Configuración para videos en formato AVI.
image/GIF	Configuración para imágenes en formato GIF.
texto/HTML	Configuración para páginas web normales.
texto/PLANO	Configuración para texto común con extensión TXT.
aplicación/x-Java-Applet	Configuración para uso de Applets de Java.
aplicación/PDF	Configuración para documentos PDF.

La implementación en los códigos anteriores sería de la siguiente manera:

```
<a href='sonido.mp3' type='audio/mpeg'>canción</a>
```

El formato de la etiqueta **<object>** sería el siguiente:

```
<object data='url del archivo'
   width='ancho del elemento'
   height='alto del elemento'
   type='mime type'>
   <param name='nombre' value='valor' />
</object>
```

Donde:

- **data:** Hace referencia a la localización del archivo en cuestión, es decir, un URL del tipo: **<http://la-direccion/del/archivo/animacion.swf>** (**swf** por tratarse en el ejemplo de una animación flash). La dirección puede ser relativa del tipo **/micarpetaflash/animación.swf**.
- **type:** Indica el tipo de MIME al que pertenece el objeto especificado en **data**. Con este dato se le dice al navegador de qué archivo se trata: si es, por ejemplo, un documento HTML, un sonido

en MP3 o una animación flash (SWF). Hay que tener cuidado, ya que el tipo que se coloque aquí ha de ser, lógicamente, el mismo que el correspondiente a la extensión del archivo al que se hace referencia en la data.

7.5.4 *La etiqueta <param>*

Se utiliza para especificar parámetros que sean necesarios en una especificación de objeto. Estos parámetros se ejecutan en tiempo de ejecución. Su formato es:

```
<param name='nombre' value='valor' />
```

Donde:

- **name:** Es el nombre asignado al parámetro según el tipo de objeto.
- **value:** Es el valor que se le da según el nombre del parámetro.

Véase la descripción de algunos parámetros relacionados con la muestra de videos:

- **type="tipo_fichero":** Permite declarar el tipo de video que se empleará.
- **width="n":** Determina el ancho del objeto expresado en píxeles.
- **height="n":** Establece el alto del objeto expresado en píxeles.
- **autostart="true/false":** Determina si el video debe empezar a reproducirse de forma automática o esperar que el usuario lo active para la visualización del mismo.
- **param name="filename" value="ruta_fichero":** Establece la ruta del fichero del video, ya sea en forma relativa a la estructura de carpetas del sitio web o en forma de URL absoluta.
- **param name="showcontrols" value="1/0":** Determina si se visualizará o no la barra de control de video.
- **param name="autostart" value="true/false":** Establece si el navegador debe empezar a reproducir el video de forma automática.

Ejemplo 11

Implemente el siguiente diseño web. Para ello, haga uso del código embebido que permite visualizar videos desde YouTube usando la etiqueta **<object>**.



Para obtener la dirección que será asignada a la etiqueta **<object>**, se debe seleccionar un video desde YouTube, luego, seleccione **Compartir** y copie el código seleccionado en la imagen. Este código será el valor del parámetro solicitado para mostrar el video.



Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
#columna {
    position:relative;
    float:left;
    left:0px;
    width: 350px;
    border-style:solid;
    border-color:#666;
    top:0px;
    text-align:center;
    font-family:Tahoma, Geneva, sans-serif;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <article id='columna'>
            <p>video 01-jealous guy</p>
            <object type='application/x-shockwave-flash'
                    data='http://www.youtube.com/v/wADRRYNHhOA'
                    width='300' height='270'>
                <param name='movie'
                    value='http://www.youtube.com/v/wADRRYNHhOA' />
            </object>
        </article>
```

```
<aside id='columna'>
   <p>video 02-my sweet lord</p>
   <object type='application/x-shockwave-flash'
         data='http://www.youtube.com/v/8qJTJNfzvr8'
         width='300' height='270'>
      <param name='movie'
         value='http://www.youtube.com/v/8qJTJNfzvr8' />
   </object>
   </aside>
   <aside id='columna'>
      <p>video 03-say say say</p>
      <object type='application/x-shockwave-flash'
         data='http://www.youtube.com/v/aLEhh_XpJ-0'
         width='300' height='270'>
      <param name='movie'
         value='http://www.youtube.com/v/aLEhh_XpJ-0' />
      </object>
   </aside>
</body>
</html>
```

CAPÍTULO 8

Manejo de tablas para la web



Competencias

- Crea tablas en un documento web a través de etiquetas básicas.
- Divide y agrupa las tablas en un documento web.

Contenido

8.1 Introducción a las tablas web
8.2 Elementos básicos y atributos de una tabla
8.3 La etiqueta <table>
8.4 La etiqueta <tr>
8.5 Las etiquetas <td> y <th>
8.6 Atributo rowspan
8.7 Atributo colspan
8.8 Atributo headers
8.9 Atributo scope
8.10 Dividir la tabla en <thead>, <tbody> y <tfoot>
8.11 Agrupando columnas - la etiqueta <colgroup>
8.12 La etiqueta <caption>

8.1 Introducción a las tablas web

Hasta antes del HTML5 las tablas eran usadas para maquetar una página web, es decir, se utilizaban para distribuir todos los elementos de la página. Hoy en día ese concepto es anticuado por la incorporación del código CSS en nuestros documentos web.

En HTML5, las tablas se comportan según su objetivo inicial: el de mostrar información textual, numérica o de imágenes en forma tabulada o mejor dicho maquetada. Véase la imagen de una maquetación realizada con HTML4 desde Dreamweaver:



Figura 8.1 Maquetado en Dreamweaver.
Fuente: Elaboración propia.

Las zonas llamadas «tabla de disposición» eran simplemente una tabla incluida dentro de otra tabla. Cada tabla de disposición contenía su propia estructura. Véanse algunos ejemplos de tablas que pueden desarrollarse con HTML5:



Pos.	Selección	Pts.	PJ	G	E	P	GF	GC	Dif.
1	Brasil	37	16	11	4	1	36	11	27
2	Uruguay	27	16	8	3	5	26	16	10
3	Colombia	26	16	7	5	4	19	16	3
4	Perú	24	16	7	3	6	26	25	1
5	Argentina	24	16	6	6	4	19	15	3
6	Chile	23	16	7	2	7	24	23	1
7	Paraguay	21	16	6	3	7	17	23	-6
8	Ecuador	20	16	6	2	8	24	24	0
9	Bolivia	13	16	4	1	11	14	34	-20
10	Venezuela	8	16	1	5	10	16	33	-17

Figura 8.2 Ejemplo de tabla web.
Fuente: <https://www.eluniverso.com/deportes/2017/09/04/nota/6365635/horarios-canales-tv-ver-fecha-16-eliminatorias-rusia-2018>.

Como se observa en la tabla anterior se combinan los textos, números, imágenes y, especialmente, los colores que se pueden aplicar en la misma. En la siguiente imagen se observa una tabla implementada para mostrar artículos y sus detalles:



Figura 8.3 Manejo de tablas web para venta de artículos.
Fuente: <https://www.amazon.com>.

El objetivo de las tablas en este libro es llegar a implementar un formulario, esto último se desarrollará más adelante.



Figura 8.4 Integración de tablas y elementos de formulario.
Fuente: <https://www.amazon.com>.



Figura 8.5 Empleo de tablas y formulario.
Fuente: <http://manantialdevida.com.pe/contactanos.php>.

8.2 Elementos básicos y atributos de una tabla

Para listar los elementos básicos de una tabla, hay que tomar en cuenta la siguiente implementación básica:

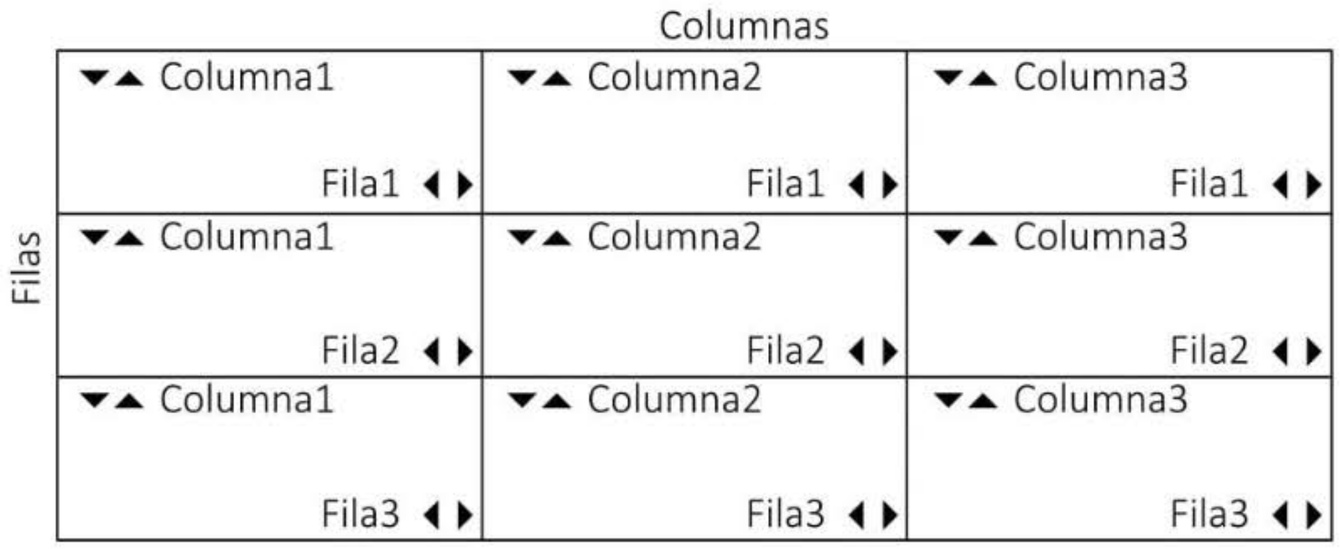


Figura 8.6 Empleo de tablas y formulario.
Fuente: Elaboración propia.

Ahora bien, véanse algunos conceptos:

- **Tabla**: Se refiere a todo el conjunto de filas y columnas y se implementa mediante la etiqueta `<table>`.
- **Fila**: Hace referencia a las filas que componen la tabla y se implementa con la etiqueta `<tr>`.
- **Celda**: Se refiere a las celdas que contendrán cada fila que componen la tabla y se implementa con la etiqueta `<td>`.

El código HTML5 para la implementación de la tabla anterior es:

```
<table>
   <tr>
      <td> </td>
      <td> </td>
      <td> </td>
   </tr>
   <tr>
      <td> </td>
      <td> </td>
      <td> </td>
   </tr>
   <tr>
      <td> </td>
      <td> </td>
      <td> </td>
   </tr>
</table>
```

8.3 La etiqueta <table>

Es un contenedor de elementos propios de la tabla como una fila o celda, también de textos, imágenes, etc. Una tabla puede implementarse dentro de las etiquetas de grupos de HTML5 como, por ejemplo, **<section>**, **<article>**, etc. Su formato es:

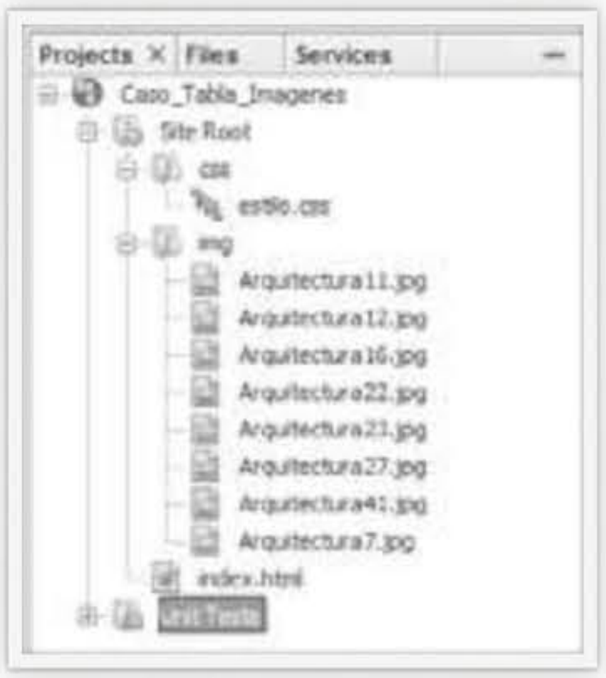
```
<table>
    <!– contenido de la tabla –>
</table>
```

Ejemplo 1

Implemente el siguiente documento web haciendo uso de las tablas.



Hay que tomar en cuenta que el proyecto debe mostrarse de la siguiente manera:



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
table {
    margin-right: auto;
    margin-left: auto;
    border-spacing: 0px;
}

td {
    padding: 0px;
    margin: 0px;
}

img {
    width: 350px;
    height: 200px;
}

/*Efecto de Texto*/
@import url(https://fonts.googleapis.com/css?family=Erica+One);

/* BODY */
body {
    background-color: pink;
    background-attachment: fixed;
    background-size: 100% 100%;
    overflow: hidden;
}

::selection {
    background: transparent;
}

h1 {
    cursor: default;
    position: absolute;
    top: 10px;
    left: 0;
    right: 0;
    bottom: 0;
    height: 100px;
    margin: auto;
    display: block;
    -webkit-animation: bounce .3s ease infinite alternate;
    font-family: 'Erica One', cursive;
    font-size: 80px;
    color: #FFF;
    text-align: center;
    line-height: 100px;
    text-shadow: 0 1px 0 #CCC,
```

```
            <tr>
                <td><img alt='' src='img/Arquitectura27.jpg"/></td>
                <td><img alt='' src='img/Arquitectura7.jpg"/></td>
            </tr>
            <tr>
                <td><img alt='' src='img/Arquitectura22.jpg"/></td>
                <td><img alt='' src='img/Arquitectura23.jpg"/></td>
            </tr>
        </table>
    </section>
</body>
</html>
```

8.3.1 *Atributo border*

Establece el ancho del borde externo que compone la tabla. El valor es especificado en píxeles. El siguiente ejemplo asigna un grosor de borde de 1px:

```
<table border=1>
    ...
</table>
```

Ejemplo 2

Implemente el siguiente modelo de tabla especificando el grosor de borde de 1px.

TIPO DE AUTO / DÍAS	LUNES-JUEVES	VIERNES	SÁBADO	DOMINGO
 Automóvil	3.50	4.00	4.50	3.50
 Camioneta	5.50	6.00	6.50	5.50
 Ómnibus	10.50	11.00	11.50	10.50
 Motos	2.50	3.00	3.50	2.50

Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
table {
    font-family: Tahoma;
    text-align: right;
}

table, tr, th, td {
    border: 1px solid #333;
    line-height: 130px;
}

th {
    background-color: #F5F5F5;
    padding: 0 .2em;
}

td {
    padding: 0 .3em;
}

th.automovil {
    background: #E6F3FF url(../img/automovil.png) no-repeat left center;
    padding: 0 .3sem 0 10.2em;
}

th.camioneta {
    background: #E6F3FF url(../img/camioneta.png) no-repeat left center;
    padding: 0 .3em 0 10.2em;
}

th.omnibus {
    background: #E6F3FF url(../img/omnibus.png) no-repeat left center;
    padding: 0 .3em 0 10.2em;
}

th.moto {
    background: #E6F3FF url(../img/moto.png) no-repeat left center;
    padding: 0 .3em 0 10.2em;
}

tr:nth-child(n+1) td {
    text-align: right;
}

tr:nth-child(2n) {
    background-color: #FFFFCC;
}

tr:hover {
    background: #FFFF66;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table>
            <tr>
                <th>TIPO DE AUTO / DÍAS</th>
                <th>LUNES-JUEVES</th>
                <th>VIERNES</th>
                <th>SÁBADO</th>
                <th>DOMINGO</th>
            </tr>
            <tr>
                <th class='automovil'>Automovil</th>
                <td>3.50</td>
                <td>4.00</td>
                <td>4.50</td>
                <td>3.50</td>
            </tr>
            <tr>
                <th class='camioneta'>Camioneta</th>
                <td>5.50</td>
                <td>6.00</td>
                <td>6.50</td>
                <td>5.50</td>
            </tr>
            <tr>
                <th class='omnibus'>Omnibus</th>
                <td>10.50</td>
                <td>11.00</td>
                <td>11.50</td>
                <td>10.50</td>
            </tr>
            <tr>
                <th class='moto'>Motos</th>
                <td>2.50</td>
                <td>3.00</td>
                <td>3.50</td>
                <td>2.50</td>
            </tr>
        </table>
    </body>
</html>
```

Ejemplo 3

Ahora bien, tomando como referencia la tabla anterior, configure la tabla de tal modo que los bordes no muestren doble línea.

TIPO DE AUTO / DÍAS	LUNES-JUEVES	VIERNES	SÁBADO	DOMINGO
 Automóvil	3.50	4.00	4.50	3.50
 Camioneta	5.50	6.00	6.50	5.50
 Ómnibus	10.50	11.00	11.50	10.50
 Motos	2.50	3.00	3.50	2.50

Código:

Solo será necesario modificar el archivo **estilo.css** donde se coloca el siguiente código:

```
table {
    font-family: Tahoma;
    text-align: right;
}

table, tr, th, td {
    border: 1px solid #333;
    line-height: 130px;
    border-spacing: 0;
}

th {
    background-color: #F5F5F5;
    padding: 0 .2em;
}
```

```
td {
   padding: 0 .3em;
}

th.automovil {
   background: #E6F3FF url(../img/automovil.png) no-repeat left center;
   padding: 0 .3sem 0 10.2em;
}

th.camioneta {
   background: #E6F3FF url(../img/camioneta.png) no-repeat left center;
   padding: 0 .3em 0 10.2em;
}

th.omnibus {
   background: #E6F3FF url(../img/omnibus.png) no-repeat left center;
   padding: 0 .3em 0 10.2em;
}

th.moto {
   background: #E6F3FF url(../img/moto.png) no-repeat left center;
   padding: 0 .3em 0 10.2em;
}

tr:nth-child(n+1) td {
   text-align: right;
}

tr:nth-child(2n) {
   background-color: #FFFFCC;
}

tr:hover {
   background: #FFFF66;
}
```

8.3.2 *Atributo bordercolor*

Establece todos los bordes que componen el color de la tabla. El siguiente ejemplo asigna un color de borde rojo:

```
<table border=1 bordercolor='#FF0000'>
   …
</table>
```

Ejemplo 4

Implemente el siguiente diseño de tabla. Para ello, use el color de borde con una asignación de color rojo y las filas con el estilo de cebra.

Libro	País	Nº Páginas	Costo
Programación Visual C#	Perú	480	65.00
HTML5 y CSS3	Perú	300	$46.00
VBA con Excel	Perú	350	$40.00
Programación PHP	Perú	450	$50.00

Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body {
    font-family: Tahoma;
}

table {
    border: 1px solid;
    border-color: red;
    padding: 5px;
}

/*Definir el color cada dos filas*/
table tr:nth-child(even) {
    background-color: #eee;
}

table tr:nth-child(odd) {
    background-color: #fff;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table>
            <tr>
                <th>Libro</th>
                <th>País</th>
                <th>Nº Páginas</th>
                <th>Costo</th>
            </tr>
```

```
            <tr>
                <td>Programación Visual C#</td>
                <td>Perú</td>
                <td>480</td>
                <td>65.00</td>
            </tr>
            <tr>
                <td>HTML5 y CSS3</td>
                <td>Perú</td>
                <td>300</td>
                <td>$46.00</td>
            </tr>
            <tr>
                <td>VBA con Excel</td>
                <td>Perú</td>
                <td>350</td>
                <td>$40.00</td>
            </tr>
            <tr>
                <td>Programación PHP</td>
                <td>Perú</td>
                <td>450</td>
                <td>$50.00</td>
            </tr>
        </table>
    </body>
</html>
```

8.3.3 *Atributo width*

Establece el ancho de una tabla con respecto al documento web. Aquí se tiene que considerar el tamaño proporcionado por el navegador.

Los valores pueden ser dados en píxeles (también llamados valores absolutos) o en porcentajes (denominados también valores relativos, ya que su tamaño es de acuerdo al tamaño proporcionado por el navegador). El siguiente ejemplo asigna un ancho de 650px:

```
<table border=1 width=650px>
    …
</table>
```

Ejemplo 5

Implemente el siguiente modelo de tabla. Use el atributo **width** con una asignación de 650px de ancho de tabla.

CATEGORÍA\INVERSIÓN	MINERÍA	CONSTRUCCIÓN	COMERCIO
MICRO-EMPRESA	250000.00	238500.00	544521.00
PEQUEÑA EMPRESA	120512.00	126458.00	823412.00
MEDIANA EMPRESA	575441.00	458572.00	234424.00
EMPRESA	425624.00	426852.00	456221.00

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body {
    font-family: Tahoma;
}

table {
    width: 650px;
    border: 2px solid;
    text-align: center;
    padding: 10px;
    background-color: beige;
}

td {
    border: 1px solid;
    background-color: bisque;
}

th {
    background-color: beige;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
```

```
<table>
    <tr>
        <th>CATEGORÍA\INVERSIÓN</th>
        <th><img src='img/mineria.jpg' alt=""/><br>
        MINERÍA</th>
        <th><img src='img/construccion.jpg' alt=""/><br>
        CONSTRUCCIÓN</th>
        <th><img src='img/comercio.jpg' alt=""/><br>
        COMERCIO</th>
    </tr>
    <tr>
        <td>MICRO-EMPRESA</td>
        <td>250000.00</td>
        <td>238500.00</td>
        <td>544521.00</td>
    </tr>
    <tr>
        <td>PEQUEÑA EMPRESA</td>
        <td>120512.00</td>
        <td>126458.00</td>
        <td>823412.00</td>
    </tr>
    <tr>
        <td>MEDIANA EMPRESA</td>
        <td>575441.00</td>
        <td>458572.00</td>
        <td>234424.00</td>
    </tr>
    <tr>
        <td>EMPRESA</td>
        <td>425624.00</td>
        <td>426852.00</td>
        <td>456221.00</td>
    </tr>
</table>
</body>
</html>
```

En caso, se necesite tener el ancho de la tabla con el mismo tamaño del navegador se puede hacer la especificación en porcentaje. Para este caso se le asignó 100% en el ancho:

CATEGORÍA\INVERSIÓN	MINERÍA	CONSTRUCCIÓN	COMERCIO
MICRO-EMPRESA	250000.00	238500.00	544521.00
PEQUEÑA EMPRESA	120512.00	126458.00	823412.00
MEDIANA EMPRESA	575441.00	458572.00	234424.00
EMPRESA	425624.00	426852.00	456221.00

Entonces, el archivo **estilo.css** sería de la siguiente forma:

```css
body {
    font-family: Tahoma;
}

table {
    width: 100%;
    border: 2px solid;
    text-align: center;
    padding: 10px;
    background-color: beige;
}

td {
    border: 1px solid;
    background-color: bisque;
}

th {
    background-color: beige;
}
```

8.3.4 *Atributo summary*

Permite agregar información acerca del contenido y la naturaleza de la tabla. El valor de este atributo no es visible por el usuario. Solo está pensado para dar referencias a la tabla implementada. Su formato es:

```html
<table summary='Descripción de la tabla'>
    <!- contenido de la tabla ->
</table>
```

El siguiente ejemplo asigna una descripción a la tabla:

```
<table width='600' border='1'
   cellspacing='0' cellpadding='0'
   summary='Comparación entre navegadores web por empresa, precio, y
            sistema operativo'>
   <tr>
      <td> </td>
      <td>Chrome</td>
      <td>Firefox</td>
      <td>Internet Explorer</td>
</tr>
<tr>
      <td>Empresa</td>
      <td>Google</td>
      <td>Mozilla fundation</td>
      <td>Microsoft</td>
</tr>
<tr>
      <td>Precio</td>
      <td>Gratuito</td>
      <td>Gratuito</td>
      <td>Gratuito</td>
</tr>
   <tr>
      <td>Sistema operativo</td>
      <td>Windows</td>
      <td>Windows-Mac </td>
      <td>Windows</td>
   </tr>
   <tr>
      <td>Uso de memoria</td>
      <td>26.50mb</td>
      <td>26.20mb</td>
      <td>32.40mb</td>
   </tr>
</table>
```

8.3.5 *Atributo cellspacing*

Establece el espaciado que puede tener la celda. Siempre debe ir especificado en píxeles, el valor por defecto es dos. El siguiente ejemplo asigna 20px de espacio en la celda:

```
<table border=1 cellspacing=20px>
   …
</table>
```

Ejemplo 6

Implemente el siguiente modelo de tabla en la que se especifique 20px de espacio entre las celdas. Además, la tabla debe mostrarse centrada en la página web.



Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
img {
    width: 148px;
    height: 189px;
}

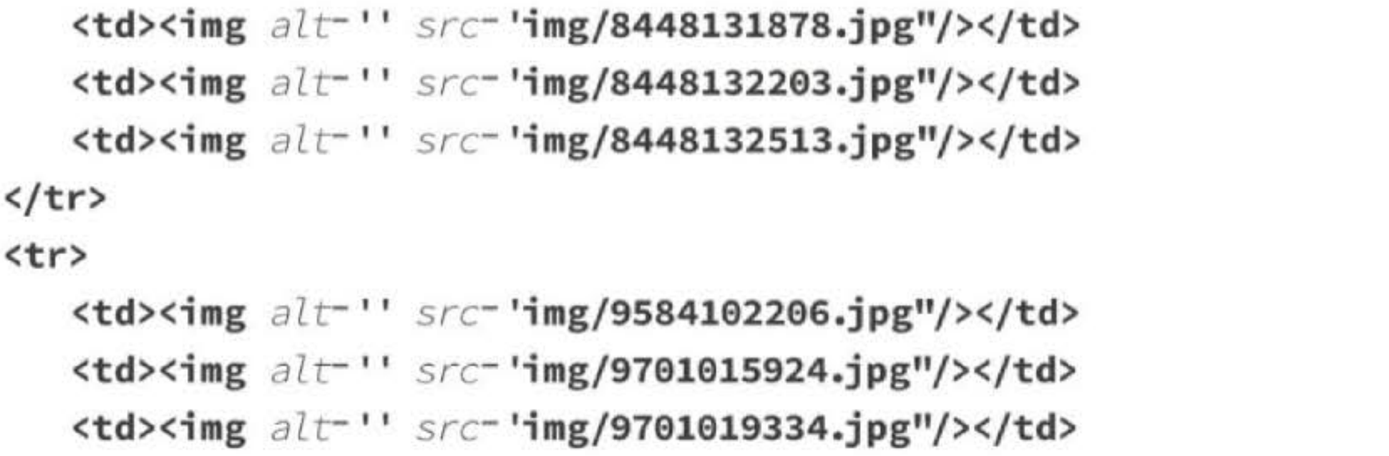
table {
    margin-left: auto;
    margin-right: auto;
    border: 1px solid;
    border-spacing: 20px;
}

td {
    border: 1px dashed;
    border-color: red;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset='utf-8'>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta content='width=device-width, initial-scale=1.0' name='viewport'>
   </head>
   <body>
      <table>
         <tr>
            <td><img alt='' src='img/8448128168.jpg"/></td>
            <td><img alt='' src='img/8448128761.jpg"/></td>
            <td><img alt='' src='img/8448129059.jpg"/></td>
            <td><img alt='' src='img/8448130332.jpg"/></td>
         </tr>
         <tr>
            <td><img alt='' src='img/8448130723.jpg"/></td>
            <td><img alt='' src='img/8448131126.jpg"/></td>
            <td><img alt='' src='img/8448131355.jpg"/></td>
            <td><img alt='' src='img/8448131541.jpg"/></td>
         </tr>
         <tr>
            <td><img alt='' src='img/8448131835.jpg"/></td>
            <td><img alt='' src='img/8448131878.jpg"/></td>
            <td><img alt='' src='img/8448132203.jpg"/></td>
            <td><img alt='' src='img/8448132513.jpg"/></td>
         </tr>
         <tr>
            <td><img alt='' src='img/9584102206.jpg"/></td>
            <td><img alt='' src='img/9701015924.jpg"/></td>
            <td><img alt='' src='img/9701019334.jpg"/></td>
            <td><img alt='' src='img/9701019335.jpg"/></td>
         </tr>
      </table>
   </body>
</html>
```

Las imágenes mostradas tienen una distribución de 20px por cada lado con respecto a la otra imagen, como se visualiza en el siguiente cuadro:



Si se anula el espacio entre imágenes, la tabla tendría el siguiente aspecto:



Para cumplir con lo anterior, se debe modificar el archivo **estilo.css** de la siguiente manera:

```
img {
    width: 148px;
    height: 189px;
}

table {
    margin-left: auto;
    margin-right: auto;
    border: 1px solid;
    border-spacing: 0px;
}

td {
    border: 1px dashed;
    border-color: red;
}
```

8.3.6 *Atributo cellpadding*

Establece el espaciado que puede tener el borde de la celda respecto a su contenido, dicho valor es especificado en píxeles. El siguiente ejemplo asigna un espacio interno de 20px:

```
<table border=1 cellpadding=20px>
   …
</table>
```

Ejemplo 7

Implemente el siguiente modelo de tabla. Use el atributo **cellpadding** con un valor de 20px.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

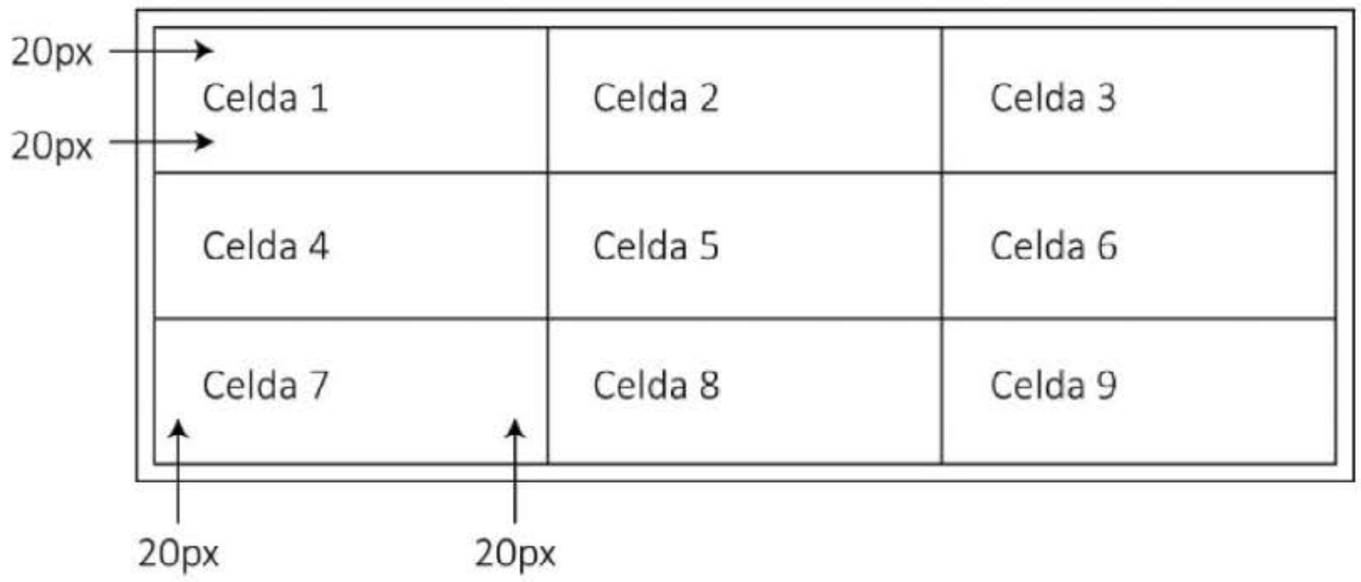
```
img {
   width: 148px;
   height: 189px;
}
table {
   margin-left: auto;
   margin-right: auto;
```

```
    border: 1px solid;
    border-spacing: 0px;
}
td {
    border: 1px dashed;
    border-color: red;
    padding: 20px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='utf-8'>
        <meta content='width=device-width, initial-scale=1.0' name='viewport'>
    </head>
    <body>
        <table>
            <tr>
                <td><img alt='' src='img/8448128168.jpg"/></td>
                <td><img alt='' src='img/8448128761.jpg"/></td>
                <td><img alt='' src='img/8448129059.jpg"/></td>
                <td><img alt='' src='img/8448130332.jpg"/></td>
            </tr>
            <tr>
                <td><img alt='' src='img/8448130723.jpg"/></td>
                <td><img alt='' src='img/8448131126.jpg"/></td>
                <td><img alt='' src='img/8448131355.jpg"/></td>
                <td><img alt='' src='img/8448131541.jpg"/></td>
            </tr>
            <tr>
                <td><img alt='' src='img/8448131835.jpg"/></td>
                <td><img alt='' src='img/8448131878.jpg"/></td>
                <td><img alt='' src='img/8448132203.jpg"/></td>
                <td><img alt='' src='img/8448132513.jpg"/></td>
            </tr>
            <tr>
                <td><img alt='' src='img/9584102206.jpg"/></td>
                <td><img alt='' src='img/9701015924.jpg"/></td>
                <td><img alt='' src='img/9701019334.jpg"/></td>
                <td><img alt='' src='img/9701019335.jpg"/></td>
            </tr>
        </table>
    </body>
</html>
```

Las imágenes mostradas tienen una especificación de espacio interno de 20px por cada lado con respecto a la otra imagen, como se visualiza en el siguiente cuadro:



8.3.7 *Atributo align*

Establece la alineación izquierda, derecha y centro de la tabla. El siguiente ejemplo centra la tabla en el documento web:

```
<table border=1 align='center'>
    ...
</table>
```

Ejemplo 8

Implemente el siguiente modelo de tabla. Emplee el atributo **align** para centrar la tabla en el documento web.

CATEGORÍA\INVERSIÓN	MINERÍA	CONSTRUCCIÓN	COMERCIO
MICRO-EMPRESA	250000.00	238500.00	544521.00
PEQUEÑA EMPRESA	120512.00	126458.00	823412.00
MEDIANA EMPRESA	575441.00	458572.00	234424.00
EMPRESA	425624.00	426852.00	456221.00

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body {
    font-family: Tahoma;
}

table {
    width: 650px;
    border: 2px solid;
    text-align: center;
```

```
    padding: 10px;
    background-color: beige;
}

td {
    border: 1px solid;
    background-color: bisque;
}

th {
    background-color: beige;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table align='center'>
            <tr>
                <th>CATEGORÍA\INVERSIÓN</th>
                <th><img src='img/mineria.jpg' alt=""/><br>
                MINERÍA</th>
                <th><img src='img/construccion.jpg' alt=""/><br>
                CONSTRUCCIÓN</th>
                <th><img src='img/comercio.jpg' alt=""/><br>
                COMERCIO</th>
            </tr>
            <tr>
                <td>MICRO-EMPRESA</td>
                <td>250000.00</td>
                <td>238500.00</td>
                <td>544521.00</td>
            </tr>
            <tr>
                <td>PEQUEÑA EMPRESA</td>
                <td>120512.00</td>
                <td>126458.00</td>
                <td>823412.00</td>
            </tr>
            <tr>
                <td>MEDIANA EMPRESA</td>
                <td>575441.00</td>
```

```
            <td>458572.00</td>
            <td>234424.00</td>
        </tr>
        <tr>
            <td>EMPRESA</td>
            <td>425624.00</td>
            <td>426852.00</td>
            <td>456221.00</td>
        </tr>
    </table>
</body>
</html>
```

En HTML5 no se recomienda centrar por medio de este atributo, ya que es mejor implementarlo mediante código CSS. Su código podría ser de la siguiente manera:

```
table{
    margin-left: auto;
    margin-right: auto;
}
```

8.3.8 *Atributo background*

Establece una imagen de fondo en la tabla. Dicha imagen se debe encontrar registrada dentro de la misma carpeta del sitio web. Su formato es:

```
<table border=1 background='../img/fondo.jpg'>
    ...
</table>
```

Ejemplo 9

El siguiente ejemplo asigna una imagen de fondo a la tabla.



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
table {
    margin-left: auto;
    margin-right: auto;
    width: 850px;
    height: 550px;
    border: 2px solid;
    text-align: center;
    padding: 10px;
    background-image: url(../img/fondo.jpg);
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table background='img/fondo.jpg'></table>
    </body>
</html>
```

8.3.9 *Atributo bgcolor*

Establece un color de fondo a la tabla. No es recomendable su uso para HTML5, ya que se puede implementar el mismo atributo por medio de código CSS. El siguiente ejemplo asigna el color rojo al fondo de la tabla.

```
<table width='350' border='1' cellspacing='20' bgcolor='#ff0000'>
    <tr>
        <td> </td> <td> </td> <td> </td>
    </tr>
    <tr>
        <td> </td> <td> </td> <td> </td>
    </tr>
    <tr>
        <td> </td> <td> </td> <td> </td>
    </tr>
</table>
```

Ejemplo 10

Implemente el siguiente modelo de tabla. Agregue un color de fondo a la tabla:

	Oro	Plata	Bronce	Cobre
Número de computadoras	Ilimitado	100	20	4
HelpDesk	Ilimitado	10 por día	1 por día	No
Renovación	$ 0.00	$ 50.00	$ 150.00	$ 450.00
Responsive móvil	Sí	Sí	No	No
CORREO CORPORATIVO	✓	✓	✓	✓
ANTIVIRUS	✓	✓	✓	✓
Precio anual por licencia	$ 1500.00	$ 1000.00	$ 850.00	$ 250.00

Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
table.table1 {
    font-family: 'Tahoma';
    font-size: 16px;
}

.table1 thead th {
    padding: 15px;
    color: #fff;
    text-shadow: 1px 1px 1px #568F23;
    border: 1px solid #93CE37;
    background-color: #9DD929;
    background: -webkit-gradient( linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(123, 192, 67)),
        color-stop(0.51, rgb(139, 198, 66)),
        color-stop(0.87, rgb(158, 217, 41))
    );
    -webkit-border-top-left-radius: 5px;
    -webkit-border-top-right-radius: 5px;
    -moz-border-radius: 5px 5px 0px 0px;
}

.table1 thead th:empty {
    background: transparent;
    border: none;
}
```

```
.table1 tfoot td {
    color: #9CD009;
    font-size: 32px;
    text-align: center;
    padding: 10px 0px;
    text-shadow: 1px 1px 1px #444;
}

.table1 tbody td {
    padding: 10px;
    text-align: center;
    background-color: #DEF3CA;
    -moz-border-radius: 2px;
    -webkit-border-radius: 2px;
    border-radius: 5px;
}

.table1 tbody span.check::before {
    content: url(../img/check.png);
}

#derecha {
    text-align: right;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table class='table1'>
            <thead>
                <tr>
                    <th></th>
                    <th>Oro</th>
                    <th>Plata</th>
                    <th>Bronce</th>
                    <th>Cobre</th>
                </tr>
            </thead>
```

```
<tfoot>
    <tr>
        <th>Precio anual por licencia</th>
        <td>$ 1500.00</td>
        <td>$ 1000.00</td>
        <td>$ 850.00</td>
        <td>$ 250.00</td>
    </tr>
</tfoot>
<tbody>
    <tr>
        <th id='derecha'>Número de computadoras</th>
        <td>Ilimitado</td>
        <td>100</td>
        <td>20</td>
        <td>4</td>
    </tr>
    <tr>
        <th id='derecha'>HelpDesk</th>
        <td>Ilimitado</td>
        <td>10 por día</td>
        <td>1 por día</td>
        <td>No</td>
    </tr>
    <tr>
        <th id='derecha'>Renovación</th>
        <td>$ 0.00</td>
        <td>$ 50.00</td>
        <td>$ 150.00</td>
        <td>$ 450.00</td>
    </tr>
    <tr>
        <th id='derecha'>Responsive móvil</th>
        <td>Sí</td>
        <td>Sí</td>
        <td>No</td>
        <td>No</td>
    </tr>
    <tr>
        <th id='derecha'>CORREO CORPORATIVO</th>
        <td><span class='check'></span></td>
        <td><span class='check'></span></td>
        <td><span class='check'></span></td>
        <td><span class='check'></span></td>
    </tr>
```

```
            <tr>
                <th id='derecha'>ANTIVIRUS</th>
                <td><span class='check'></span></td>
                <td><span class='check'></span></td>
                <td><span class='check'></span></td>
                <td><span class='check'></span></td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```

8.4 La etiqueta <tr>

La etiqueta `<tr>` permite implementar filas que componen a una tabla. Esta etiqueta puede ser hija de los siguientes elementos: `<table>`, `<thead>`, `<tfoot>` y `<tbody>`. Por regla general, la etiqueta `<tr>` debe contener uno o más elementos `<td>` que son los que forman la columna dentro de la tabla. Su formato es:

```
<table>
    <tr>
        <!– contenido de la celda –>
    </tr>
</table>
```

Véanse los principales parámetros de la etiqueta `<tr>`:

- **Atributo bgColor**
  Establece el color de fondo de la fila completa. Hay que tener en cuenta que dicho atributo queda obsoleto para HTML5, ya que dicha implementación podría manejarse mediante código CSS. El siguiente ejemplo asigna el color rojo a las filas alternadas de la tabla:

Ejemplo 11

Implemente el siguiente modelo de tabla usando el color de fondo mediante la etiqueta `<tr>`.

	ORO	PLATA	BRONCE	COBRE
Número de computadoras	Ilimitado	100	20	4
HelpDesk	Ilimitado	10 por día	1 por día	No
Renovación	$ 0.00	$ 50.00	$ 150.00	$ 450.00
Responsive móvil	Sí	Sí	No	No
Precio anual por licencia	$ 1500.00	$ 1000.00	$ 850.00	$ 250.00

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body {
    margin: 0;
    font-family: 'Tahoma';
    font-size: 18px;
    padding: 10px;
    background: #ccc;
    text-align: center;
    font-family: arial;
}

table {
    margin: auto;
    width: 890px;
    text-align: left;
    background: white;
    border: 1px solid silver;
}

table td {
    padding: 8px;
}

tr:nth-child(2n+1) {
    background-color: beige;
    color: red;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table class='table1'>
            <thead>
                <tr>
                    <th></th>
                    <th>ORO</th>
                    <th>PLATA</th>
                    <th>BRONCE</th>
                    <th>COBRE</th>
                </tr>
            </thead>
```

```
        <tfoot>
            <tr>
                <th>Precio anual por licencia</th>
                <td>$ 1500.00</td>
                <td>$ 1000.00</td>
                <td>$ 850.00</td>
                <td>$ 250.00</td>
            </tr>
        </tfoot>
        <tbody>
            <tr>
                <th id='derecha'>Número de computadoras</th>
                <td>Ilimitado</td>
                <td>100</td>
                <td>20</td>
                <td>4</td>
            </tr>
            <tr>
                <th id='derecha'>HelpDesk</th>
                <td>Ilimitado</td>
                <td>10 por día</td>
                <td>1 por día</td>
                <td>No</td>
            </tr>
            <tr>
                <th id='derecha'>Renovación</th>
                <td>$ 0.00</td>
                <td>$ 50.00</td>
                <td>$ 150.00</td>
                <td>$ 450.00</td>
            </tr>
            <tr>
                <th id='derecha'>Responsive móvil</th>
                <td>Sí</td>
                <td>Sí</td>
                <td>No</td>
                <td>No</td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```

- **Atributo align**

  Establece la alineación de los valores contenidos en toda la fila de la tabla. Hay que tener en cuenta que dicho atributo queda obsoleto para HTML5, puesto que dicha implementación podría manejarse mediante código CSS.

Ejemplo 12

Implemente el siguiente modelo de tabla haciendo uso de la alineación derecha en la etiqueta `<tr>`.

	CHROME	FIREFOX	INTERNET
CORPORATIVO	$ 500.00	$ 400.00	$ 600.00
PERSONAL	$ 50.00	$ 25.00	$ 15.00
EDUCATIVO	$ 5.00	$ 2.00	$ 1.00
COMUNIDAD	$ 0.00	$ 0.00	$ 0.00

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
    margin: 0;
    font-size: 18px;
    padding: 10px;
    background: #ccc;
    font-family: 'Tahoma';
    font-size: 16px;
}

.table1 {
    width: 750px;
}

.table1 tr td {
    padding: 15px;
    color: #fff;
    text-shadow: 1px 1px 1px #568F23;
    border: 1px solid #93CE37;
    background: -webkit-gradient( linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(0, 92, 255)),
        color-stop(0.51, rgb(0, 98, 255)),
        color-stop(0.87, rgb(0, 17, 255))
    );
    -webkit-border-top-left-radius: 5px;
    -webkit-border-top-right-radius: 5px;
    -moz-border-radius: 5px 5px 0px 0px;
}
```

```
#centro {
    text-align: center;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table class='table1'>
            <tr id='centro'>
                <td> </td>
                <td>CHROME</td>
                <td>FIREFOX</td>
                <td>INTERNET</td>
            </tr>
            <tr align='right'>
                <td>CORPORATIVO</td>
                <td>$ 500.00</td>
                <td>$ 400.00</td>
                <td>$ 600.00</td>
            </tr>
            <tr align='right'>
                <td>PERSONAL</td>
                <td>$ 50.00</td>
                <td>$ 25.00</td>
                <td>$ 15.00</td>
            </tr>
            <tr align='right'>
                <td>EDUCATIVO</td>
                <td>$ 5.00</td>
                <td>$ 2.00</td>
                <td>$ 1.00</td>
            </tr>
            <tr align='right'>
                <td>COMUNIDAD</td>
                <td>$ 0.00</td>
                <td>$ 0.00</td>
                <td>$ 0.00</td>
            </tr>
        </table>
    </body>
</html>
```

8.5 Las etiquetas <td> y <th>

La etiqueta **<td>** permite definir las columnas dentro de las filas implementadas de la tabla. Entre los atributos más importantes se encuentran **bgcolor** y **align**.

Por regla general, la etiqueta **<td>** debe estar contenida dentro de las etiquetas **<tr>** para formar una tabla, ya que cada **<td>** forma la columna de la tabla. Su formato es:

```
<table>
   <tr>
      <td>
         <!– contenido de la celda –>
      </td>
   </tr>
</table>
```

Ejemplo 13

Implemente el siguiente modelo de tabla usando las etiquetas **<td>** y **<th>**.

Apellidos	Nombres	Correo Electrónico	Cuota Inicial	Teléfono
Rojas	Juan	jrojas@gmail.com	$150.00	051-985632541
Barraza	Franco	fbarraza@hotmail.com	$500.00	051-999858582
García	Hurtado	hgarcia@hotmail.com	$1000.00	051-5589685
Lázaro	Ángela	alazaro@nk.net	$250.00	051-5256526

Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body {
   font: normal medium/1.4 sans-serif;
}

table {
   border-collapse: collapse;
   width: 40%;
}

th, td {
   padding: 0.25rem;
   text-align: left;
   border: 1px solid #ccc;
}

tbody tr:nth-child(odd) {
   background: #eee;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
<body>
    <table class='zebra'>
        <thead>
            <tr>
                <th>Apellidos</th>
                <th>Nombres</th>
                <th>Correo Electrónico</th>
                <th>Cuota Inicial</th>
                <th>Teléfono</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>Rojas</td>
                <td>Juan</td>
                <td>jrojas@gmail.com</td>
                <td>$150.00</td>
                <td>051-985632541</td>
            </tr>
            <tr>
                <td>Barraza</td>
                <td>Franco</td>
                <td>fbarraza@hotmail.com</td>
                <td>$500.00</td>
                <td>051-999858582</td>
            </tr>
            <tr>
                <td>García</td>
                <td>Hurtado</td>
                <td>hgarcia@hotmail.com</td>
                <td>$1000.00</td>
                <td>051-5589685</td>
            </tr>
            <tr>
                <td>Lázaro</td>
                <td>Ángela</td>
                <td>alazaro@nk.net</td>
```

```
                <td>$250.00</td>
                <td>051-5258526</td>
            </tr>
        </tbody>
    </table>
</body>
</html>
```

8.6 Atributo rowspan

Permite combinar dos o más celdas en una misma columna dentro de la tabla. Hay que tener en cuenta que **rowspan** debe implementarse dentro de la etiqueta **<td>**. Su formato es:

```
<td rowspan='numero de filas a combinar'>
    <!- contenido de la fila combinada ->
</td>
```

Ejemplo 14

Implemente el siguiente modelo de tabla usando la propiedad **rowspan**.

ENERO	Ingreso	$ 6520.00
	Egreso	$ 2500.00
	TOTAL	$ 4020.00
FEBRERO	Ingreso	$ 15000.00
	Egreso	$ 1500.00
	TOTAL	$ 13500.00
MARZO	Ingreso	$ 12200.00
	Egreso	$ 12000.00
	TOTAL	$ 200.00

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body{
    margin: 0;
    font-size: 18px;
    padding: 10px;
    background: #ccc;
    font-family: 'Tahoma';
    font-size: 16px;
}

.table1 {
    width: 750px;
}

.table1 tr td{
    padding:15px;
    color:#fff;
    text-shadow:1px 1px 1px #568F23;
    border:1px solid #93CE37;
    background:-webkit-gradient(
        linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(0,92,255)),
        color-stop(0.51, rgb(0,98,255)),
        color-stop(0.87, rgb(0,17,255))
    );
    -webkit-border-top-left-radius:5px;
    -webkit-border-top-right-radius:5px;
    -moz-border-radius:5px 5px 0px 0px;
}

.table1 tr td#filaTotal{
    color:#fff;
    background:-webkit-gradient(
        linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(250,0,100)),
        color-stop(0.51, rgb(250,0,100)),
        color-stop(0.87, rgb(250,0,100))
        );
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table class='table1'>
            <tr>
                <td rowspan='3'>ENERO</td>
                <td>Ingreso</td>
                <td>$ 6520.00</td>
            </tr>
            <tr>
                <td>Egreso</td>
                <td>$ 2500.00</td>
            </tr>
            <tr>
                <td id='filaTotal'>TOTAL</td>
                <td>$ 4020.00</td>
            </tr>
            <tr>
                <td rowspan='3'>FEBRERO</td>
                <td>Ingreso</td>
                <td>$ 15000.00</td>
            </tr>
            <tr>
                <td>Egreso</td>
                <td>$ 1500.00</td>
            </tr>
            <tr>
                <td id='filaTotal'>TOTAL</td>
                <td>$ 13500.00</td>
            </tr>
            <tr>
                <td rowspan='3'>MARZO</td>
                <td>Ingreso</td>
                <td>$ 12200.00</td>
            </tr>
            <tr>
                <td>Egreso</td>
                <td>$ 12000.00</td>
            </tr>
```

```
            <tr>
                <td id='filaTotal'>TOTAL</td>
                <td>$ 200.00</td>
            </tr>
        </table>
    </body>
</html>
```

8.7 Atributo colspan

Permite combinar dos o más celdas en una misma fila dentro de la tabla. Hay que tener en cuenta que **colspan** debe implementarse dentro de la etiqueta **<td>**. Su formato es:

```
<td colspan='numero de columnas a combinar'>
    <!- contenido de la columna combinada ->
</td>
```

Ejemplo 15

Implemente el siguiente modelo de tabla usando la propiedad **colspan**.

ENERO			FEBRERO			MARZO		
Ingreso	Egreso	TOTAL	Ingreso	Egreso	TOTAL	Ingreso	Egreso	TOTAL
$ 10000.00	$ 5000.00	$ 5000.00	$ 15500.00	$ 1500.00	$ 14000.00	$ 20000.00	$ 2500.00	$ 17500.00

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body{
    margin: 0;
    font-size: 18px;
    padding: 10px;
    background: #ccc;
    font-family: 'Tahoma';
    font-size: 16px;
}

.table1 {
    width: 1050px;
}

.table1 tr td{
    padding:15px;
    color:#fff;
    text-shadow:1px 1px 1px #568F23;
```

```
    border:1px solid #93CE37;
    background:-webkit-gradient(
        linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(0,92,255)),
        color-stop(0.51, rgb(0,98,255)),
        color-stop(0.87, rgb(0,17,255))
    );
    -webkit-border-top-left-radius:5px;
    -webkit-border-top-right-radius:5px;
    -moz-border-radius:5px 5px 0px 0px;
}

.table1 tr td#filaTotal{
    color:#fff;
    background:-webkit-gradient(
        linear,
        left bottom,
        left top,
        color-stop(0.02, rgb(250,0,100)),
        color-stop(0.51, rgb(250,0,100)),
        color-stop(0.87, rgb(250,0,100))
    );
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table class='table1'>
            <tr>
                <td colspan='3'>ENERO</td>
                <td colspan='3'>FEBRERO</td>
                <td colspan='3'>MARZO</td>
            </tr>
            <tr>
                <td>Ingreso</td>
                <td>Egreso</td>
                <td id='filaTotal'>TOTAL</td>
                <td>Ingreso</td>
                <td>Egreso</td>
                <td id='filaTotal'>TOTAL</td>
                <td>Ingreso</td>
```

```
            <td>Egreso</td>
            <td id='filaTotal'>TOTAL</td>
        </tr>
        <tr>
            <td>$ 10000.00</td>
            <td>$ 5000.00</td>
            <td>$ 5000.00</td>
            <td>$ 15500.00</td>
            <td>$ 1500.00</td>
            <td>$ 14000.00</td>
            <td>$ 20000.00</td>
            <td>$ 2500.00</td>
            <td>$ 17500.00</td>
        </tr>
    </table>
</body>
</html>
```

8.8 Atributo headers

A través del valor dado a los atributos **id**, se utiliza para referenciar las celdas superiores a las que se está aplicando dicho atributo.

```
<td headers="">
    <!- contenido de la columna combinada ->
</td>
```

Ejemplo 16

Implemente el siguiente modelo de tabla usando el atributo **headers**.

TRABAJOS DOMICILIARIOS	EVALUACIONES			PROYECTOS		
	EV1	EV1	FINAL	PP1	PP2	INFORME
15%	15%	15%	20%	10%	10%	15%

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
    font: normal medium/1.4 sans-serif;
}

table {
    border-collapse: collapse;
    width: 40%;
}
```

```
th, td {
    padding: 0.25rem;
    text-align: center;
    border: 1px solid #ccc;
}

tbody tr:nth-child(odd) {
    background: #eee;
}
```

En el archivo **index.html** se agrega el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <table>
            <tr>
                <th rowspan='2' id='td'>TRABAJOS DOMICILIARIOS</th>
                <th colspan='3' id='ev'>EVALUACIONES</th>
                <th colspan='3' id='pp'>PROYECTOS</th>
            </tr>
            <tr>
                <th id='lc1' headers='ev'>EV1</th>
                <th id='lc2' headers='ev'>EV1</th>
                <th id='lf1' headers='ev'>FINAL</th>
                <th id='p1' headers='pp'>PP1</th>
                <th id='p2' headers='pp'>PP2</th>
                <th id='lf2' headers='pp'>INFORME</th>
            </tr>
            <tr>
                <td headers='td'>15%</td>
                <td headers='ev lc1'>15%</td>
                <td headers='ev lc2'>15%</td>
                <td headers='ev lf1'>20%</td>
                <td headers='pp p1'>10%</td>
                <td headers='pp p2'>10%</td>
                <td headers='pp lf2'>15%</td>
            </tr>
        </table>
    </body>
</html>
```

8.9 Atributo scope

El atributo **scope** se puede usar en tablas sencillas y en los encabezamientos de la etiqueta **<th>**. Tiene por misión indicar un conjunto de celdas según la información proporcionada. Se dice que es una alternativa al atributo **<headers>**.

Su principal característica es vincular la información contenida en las diferentes celdas de una tabla asignándole un valor determinado. Estos valores pueden ser:

- **row:** Se refiere al resto de celdas que componen su fila.
- **col:** Indica al resto de celdas que componen su columna.
- **rowgroup:** Se refiere a las celdas que componen su grupo de filas.
- **colgroup:** Indica a las celdas que componen su grupo de columnas.

Ejemplo 17

Implemente el siguiente modelo de tabla haciendo uso de los encabezados

COMPARACIÓN ENTRE LOS NAVEGADORES WEB				
NAVEGADOR	**ÚLTIMA VERSIÓN**	**AUTOR**	**LICENCIA**	**FECHA DE PUBLICACIÓN**
Google Chrome	32.0.1312.52	Google	Libre BSD	Enero 2018
Mozilla Firefox	46.0	Mozilla Foundation	Libre: mpl, gnu gpl, gnu lgnu	Setiembre 2017
Internet Explorer	9.0.8112.16421	Microsoft	Microsoft	Agosto 1995

Códigos:

En el archivo **estilo.css** se inserta el siguiente código:

```
body {
    font: normal medium/1.4 sans-serif;
}

table {
    border-collapse: collapse;
    width: 40%;
}

th, td {
    padding: 0.25rem;
    text-align: center;
    border: 1px solid #ccc;
}

tbody tr:nth-child(odd) {
    background: #eee;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='utf-8'>
      <meta content='width=device-width, initial-scale=1.0' name='viewport'>
   </head>
   <body>
      <table border='1' cellpadding='0' cellspacing='0'>
         <tr>
            <th colspan='5' scope='colgroup'>
               COMPARACIÓN ENTRE LOS NAVEGADORES WEB
            </th>
         </tr>
         <tr>
            <th abbr='nombre' scope='col'>NAVEGADOR</th>
            <th abbr='tutor' scope='col'>ÚLTIMA VERSIÓN</th>
            <th scope='col'>AUTOR</th>
            <th scope='col'>LICENCIA</th>
            <th scope='col'>FECHA DE PUBLICACIÓN</th>
         </tr>
         <tr>
            <td scope='row'>Google Chrome</td>
            <td>32.0.1312.52</td>
            <td>Google</td>
            <td>Libre BSD</td>
            <td>Enero 2018</td>
         </tr>
         <tr>
            <td scope='row'>Mozilla Firefox</td>
            <td>46.0</td>
            <td>Mozilla Foundation</td>
            <td>Libre: mpl, gnu gpl, gnu lgnu</td>
            <td>Setiembre 2017</td>
         </tr>
         <tr>
            <td scope='row'>Internet Explorer</td>
            <td>9.0.8112.16421</td>
            <td>Microsoft</td>
            <td>Microsoft</td>
            <td>Agosto 1995</td>
         </tr>
      </table>
   </body>
</html>
```

8.10 Dividir la tabla en <thead>, <tbody> y <tfoot>

8.10.1 *La etiqueta <thead>*

Permite definir una sección del encabezado en una tabla. Su trabajo es similar a la etiqueta **<header>** de un documento web. Un **<thead>** debe contener una o más filas que permitan mostrar información de encabezado.

La implementación de un **<thead>** en una tabla ayudará a replicar el mismo encabezado en todas las hojas que ocupe la tabla cuando esta sea extensa. Esto mejora la visibilidad de la información contenida en la tabla. Su formato es:

```
<thead>
    <tr>
        <th>…</th>
        <th>…</th>
        <th>…</th>
    </tr>
</thead>
```

8.10.2 *La etiqueta <tbody>*

Permite definir una sección del cuerpo en una tabla. Su trabajo es similar a la etiqueta **<body>** de un documento web. Al igual que la etiqueta **<thead>** siempre debe contener una o más filas que permitan mostrar información. Esta debe dividirse de la mejor manera posible.

La etiqueta **<tbody>** no es visible para el usuario, pero a nivel de código es recomendable por la semántica que le proporciona a la implementación de una tabla. Su formato es:

```
<tbody>
    <tr>
        <th>…</th>
        <th>…</th>
    </tr>
    <tr>
        <th>…</th>
        <th>…</th>
    </tr>
</tbody>
```

8.10.3 *La etiqueta <tfoot>*

Permite definir una sección de pie en una tabla. Su trabajo es similar a la etiqueta **<footer>** de un documento web. Al igual que sus predecesores siempre deberá contener una o más filas que permitan mostrar información referente al pie de la tabla. Cuando la tabla es demasiada extensa en contenido del **<tfoot>**, tiende a repetirse en las páginas que ocupe.

La etiqueta **<tfoot>** no es visible para el usuario, pero a nivel de código es recomendable por la semántica que le proporciona a la implementación de una tabla. Su formato es:

```
<tfoot>
   <tr>
      <th>…</th>
      <th>…</th>
   </tr>
</tfoot>
```

Ejemplo 18

Implemente el siguiente modelo de tabla haciendo uso de la división de una tabla.

NAVEGADOR	VERSIÓN	LICENCIA
Google Chrome	24.0	Gratuita
Internet Explorer	9.0	
Mozilla Firefox	26.0	
Fuente: http://es.wikipedia.org/wiki/navegador_web		

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
body {
   font: normal medium/1.4 sans-serif;
}

table {
   border-collapse: collapse;
   width: 40%;
}

th, td {
   padding: 0.25rem;
   text-align: center;
   border: 1px solid #ccc;
}

tbody tr:nth-child(odd) {
   background: #eee;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='utf-8'>
         <meta content='width=device-width, initial-scale=1.0' name='viewport'>
         </meta>
      </meta>
   </head>
   <body>
      <table border='1' cellpadding='0' cellspacing='0'>
         <thead>
            <tr>
               <th>NAVEGADOR</th>
               <th>VERSIÓN</th>
               <th>LICENCIA</th>
            </tr>
         </thead>
         <tbody>
            <tr>
               <td>Google Chrome</td>
               <td>24.0</td>
               <td rowspan='3'>Gratuita</td>
            </tr>
            <tr>
               <td>Internet Explorer</td>
               <td>9.0</td>
            </tr>
            <tr>
               <td>Mozilla Firefox</td>
               <td>26.0</td>
            </tr>
         </tbody>
         <tfoot>
            <tr>
               <td colspan='3'>Fuente: http://es.wikipedia.org/wiki/navegador_web
               </td>
            </tr>
         </tfoot>
      </table>
   </body>
</html>
```

8.11 Agrupando columnas–la etiqueta <colgroup>

La etiqueta **<colgroup>** tiene por finalidad agrupar a las columnas de una misma tabla para una especificación de estilos específicos, es decir, se podrá asignar, por ejemplo, diferentes colores a las columnas sin necesidad de especificar celda por celda. La característica principal es que **<colgroup>** siempre se define al inicio de la implementación de la tabla.

Ejemplo 19

Implemente el siguiente modelo de tabla haciendo uso de **<colgroup>**.

NAVEGADOR	VERSIÓN	LICENCIA	SISTEMA OPERATIVO	PRECIO	MEMORIA
Chrome	24	Gratuita	Windows	-	26.50
Firefox	26	Gratuita	Windows–Mac	-	26.20
Internet Explorer	9	Gratuita	Windows	-	32.40

Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body{
    font-family: Tahoma;
    font-size: 16px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='utf-8'>
            <meta content='width=device-width, initial-scale=1.0' name='viewport'>
            </meta>
        </meta>
    </head>
    <body>
        <table border='1' cellpadding='5' cellspacing='0'>
            <colgroup style='background: #bbbbff;'></colgroup>
            <colgroup>
                <col style='background: #aaffaa;'></col>
                <col style='background: #aaffaa;'></col>
            </colgroup>
```

```
<colgroup>
    <col style='background: #ffbbbb;'></col>
    <col style='background: #ffbbbb;'></col>
    <col style='background: #ffbbbb;'></col>
</colgroup>
<tr>
    <th>NAVEGADOR</th>
    <th>VERSIÓN</th>
    <th>LICENCIA</th>
    <th>SISTEMA OPERATIVO</th>
    <th>PRECIO</th>
    <th>MEMORIA</th>
</tr>
<tr>
    <td>Chrome</td>
    <td>24</td>
    <td>Gratuita</td>
    <td>Windows</td>
    <td>-</td>
    <td>26.50</td>
</tr>
<tr>
    <td>Firefox</td>
    <td>26</td>
    <td>Gratuita</td>
    <td>Windows-Mac</td>
    <td>-</td>
    <td>26.20</td>
</tr>
<tr>
    <td>Internet Explorer</td>
    <td>9</td>
    <td>Gratuita</td>
    <td>Windows</td>
    <td>-</td>
    <td>32.40</td>
</tr>
</table>
</body>
</html>
```

8.12 La etiqueta <caption>

Permite insertar un título a la tabla. No cuenta con especificación de celdas, por ende, siempre aparecerá en la parte superior de la tabla. Siempre debe implementarse en la línea siguiente a la etiqueta **<table>**. Su formato es:

```
<table border='1' cellspacing='0' cellpadding='5'>
   <caption>
      <!- contenido del titulo de la tabla ->
   </caption>
</table>
```

Ejemplo 20

Implemente el siguiente modelo de la tabla haciendo uso de la etiqueta **<caption>**.

COMPARACIÓN ENTRE NAVEGADORES WEB

NAVEGADORES WEB	VERSIÓN	LICENCIA	SISTEMA OPERATIVO	PRECIO	MEMORIA
Chrome	24	Gratuita	Windows	-	26.50
Firefox	26	Gratuita	Windows-Mac	-	26.20
Internet Explorer	9	Gratuita	Windows	-	32.40

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

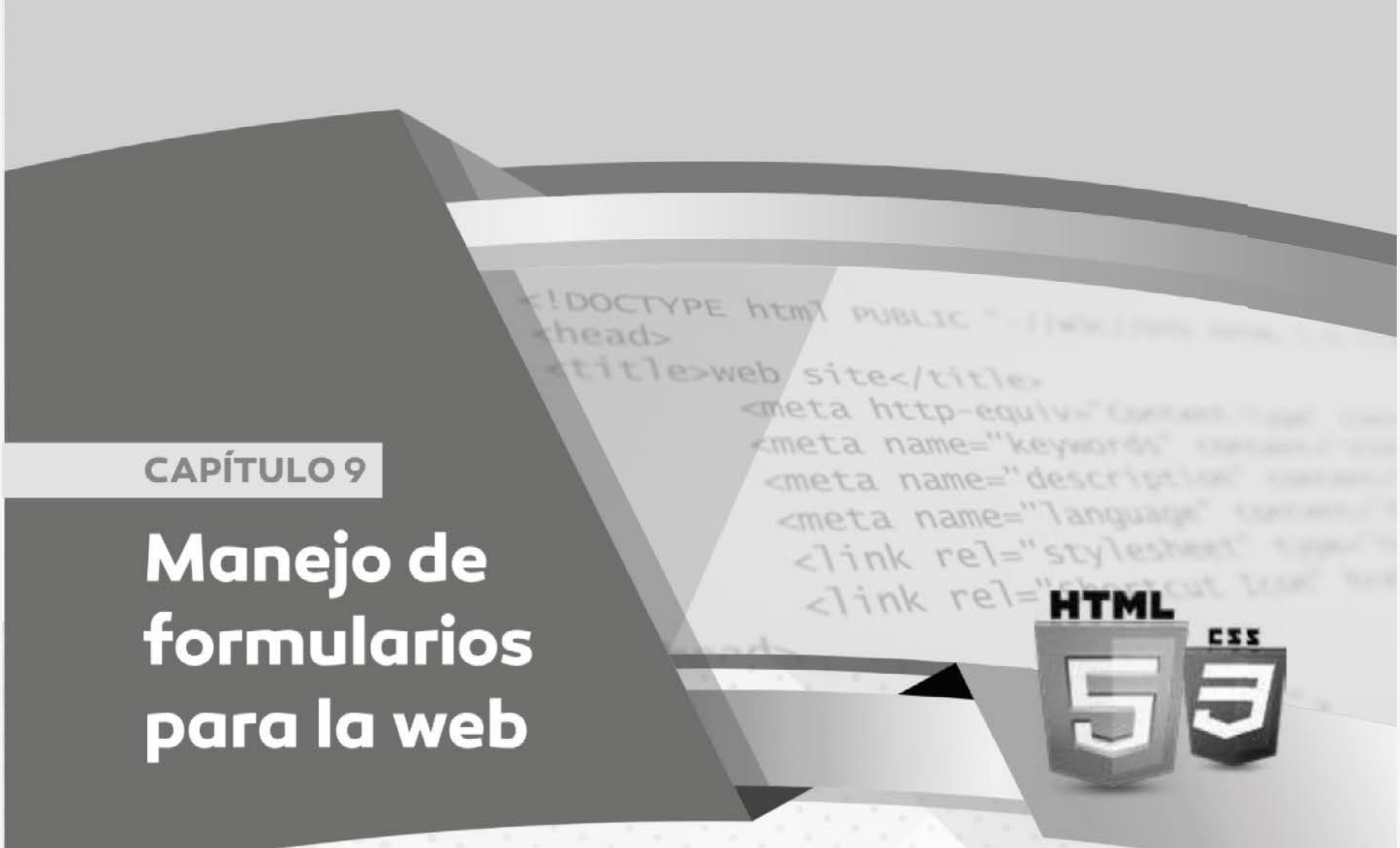
```
body{
   font-family: Tahoma;
   font-size: 16px;
}

caption{
   font-weight: bold;
   font-size: 20px;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
   <head>
      <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
      <meta charset='utf-8'>
         <meta content='width=device-width, initial-scale=1.0' name='viewport'>
         </meta>
      </meta>
   </head>
   <body>
      <table border='1' cellpadding='5' cellspacing='0'>
         <caption>COMPARACIÓN ENTRE NAVEGADORES WEB</caption>
         <colgroup style='background: #bbbbff;'></colgroup>
         <colgroup>
            <col style='background: #aaffaa;'></col>
            <col style='background: #aaffaa;'></col>
         </colgroup>
         <colgroup>
            <col style='background: #ffbbbb;'></col>
            <col style='background: #ffbbbb;'></col>
            <col style='background: #ffbbbb;'></col>
         </colgroup>
         <tr>
            <th>NAVEGADORES WEB</th>
            <th>VERSIÓN</th>
            <th>LICENCIA</th>
            <th>SISTEMA OPERATIVO</th>
            <th>PRECIO</th>
            <th>MEMORIA</th>
         </tr>
         <tr>
            <td>Chrome</td>
            <td>24</td>
            <td>Gratuita</td>
            <td>Windows</td>
            <td>-</td>
            <td>26.50</td>
         </tr>
         <tr>
            <td>Firefox</td>
            <td>26</td>
            <td>Gratuita</td>
            <td>Windows-Mac</td>
            <td>-</td>
            <td>26.20</td>
         </tr>
```

```
            <tr>
                <td>Internet Explorer</td>
                <td>9</td>
                <td>Gratuita</td>
                <td>Windows</td>
                <td>-</td>
                <td>32.40</td>
            </tr>
        </table>
    </body>
</html>
```



CAPÍTULO 9

Manejo de formularios para la web

Competencias

- Distingue los distintos tipos de formularios y conoce el empleo de la etiqueta **<form>** y sus respectivos atributos.
- Crea un formulario de registro en un documento web.

Contenido

9.1 Introducción a los formularios web
9.2 Creando un formulario - etiqueta <form>
9.3 Control de textos para formularios
9.4 Control de botones para formularios
9.5 Control de cuadros de chequeo
9.6 Control de botones de radio
9.7 Control de cuadro combinado
9.8 Control de cuadro de lista
9.9 Etiqueta <datalist>
- **Caso desarrollado:** Formulario de registro
9.10 Control de archivos
9.11 Control de datos ocultos

9.1 Introducción a los formularios web

Los formularios web presentan un mecanismo eficiente para obtener información de los usuarios. Esta capacidad exige que los formularios sean elaborados de la mejor manera posible, ya que los datos que se reciban serán procesados por la organización que los necesite. Véanse algunos casos en los que se recomienda el uso de los formularios web.

9.1.1 *Formulario de contacto*

Este formulario solicita información al usuario y es administrado por la empresa que requiera dicha información. La idea es tener el control de los clientes que visitan la web y saber qué es lo requieren cada vez que le dan clic al botón **Enviar**.



Figura 7.1 Formulario de contacto.
Fuente: Elaboración propia.

9.1.2 *Formulario de encuesta de satisfacción*

Este formulario permite saber qué tanto cumple una empresa con lo que ofrece a sus clientes. Para la empresa puede ser un dato muy importante conocer en qué están fallando o en qué deben mejorar.

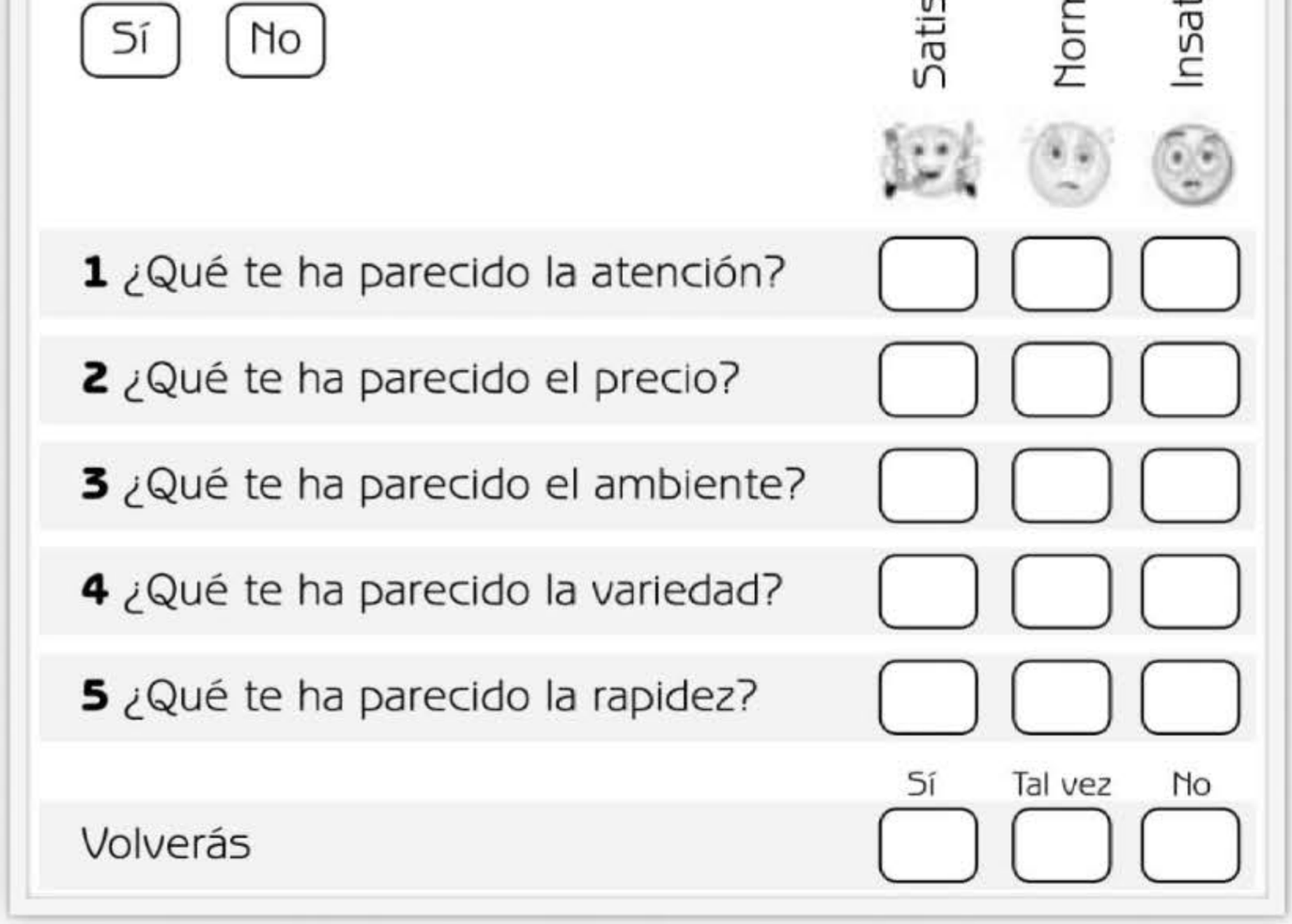


Figura 7.2 Encuesta de satisfacción de los clientes.

Fuente: <http://montarunbarysobrevivir.blogspot.com/2013/01/10-claves-para-realizar-una-encuesta-de.html>.

9.1.3 *Formulario de solicitud de empleo o registro de datos*

Este formulario permite recoger los datos de alguien que desea postular a un determinado puesto de trabajo.



Figura 7.3 Formato de solicitud de empleo
Fuente: <www.formatosexcel.blogspot.com>.

9.1.4 *Formulario de reservación*

Este formulario es el más usado en la actualidad, pues si se quiere reservar una habitación en el hotel de una ciudad distante, se podría realizar mediante un formulario web, como la que se observa a continuación:



Figura 7.4 Formulario de reserva de hoteles.
Fuente: <https://blog.trabber.com/2015/03/descuento-reservando-hotel-hotelopia/>.

Cuando un cliente envía un correo electrónico puede enviar, o no, toda la información que necesita a la empresa. Esto es lo que diferencia a los formularios, pues en ellos se le solicita la información exacta al cliente. Y luego, dicha información será manejada de la mejor manera posible por el webmaster de la empresa.

HTML5 no tiene la capacidad para procesar los datos obtenidos desde los formularios. Aquí es donde se hace uso de los lenguajes de programación y de los administradores de base de datos, ellos nos garantizan el uso adecuado de los datos obtenidos. Véase una lista de los lenguajes que trabajan con HTML5:

- **ASP.NET** (Active Server Pages): Es un *framework* para aplicaciones web desarrollado y comercializado por Microsoft. Es usado por programadores y diseñadores para construir sitios webs dinámicos, aplicaciones webs y servicios webs XML.



- **PHP** (Hypertext Pre-processor): Es un lenguaje de programación de uso general, cuyo código trabaja del lado del servidor, originalmente fue diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos.



- **Java Servlet:** El servlet es una clase en el lenguaje de programación Java, utilizada para ampliar las capacidades de un servidor. Aunque los servlets pueden responder a cualquier tipo de solicitudes, que se utilizan comúnmente para extender las aplicaciones alojadas por servidores web, por lo que se pueden considerar como los applets de Java que se ejecutan en servidores en lugar de los navegadores web. Este tipo de servlets es la contraparte Java de otras tecnologías de contenido dinámico web, como PHP y ASP.NET.



Los formularios están compuestos de controles que permiten la correcta organización de sus datos. Véase la composición en controles de un formulario de registro:

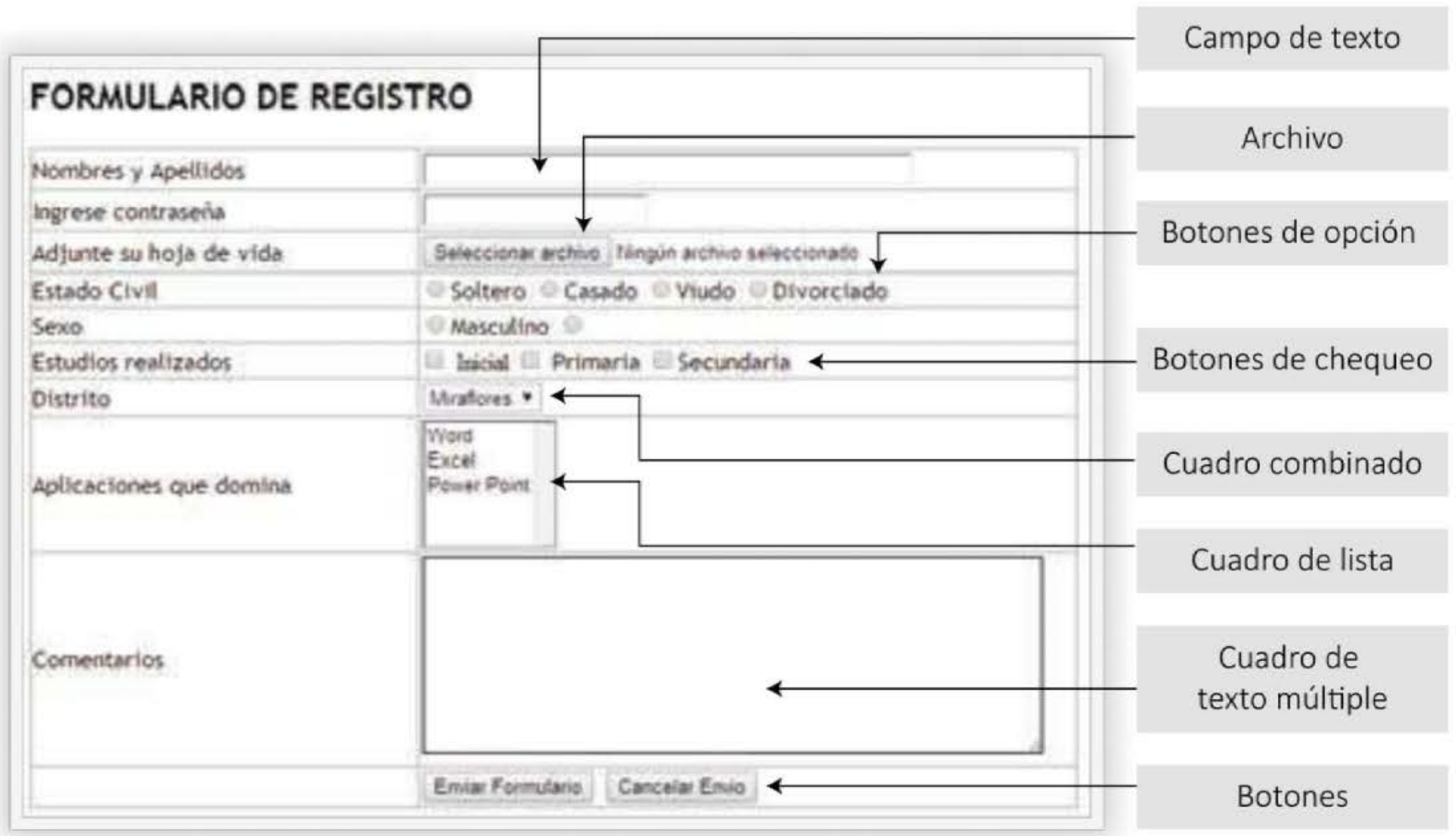


Figura 7.5 Formulario de registro personal.
Fuente: Elaboración propia.

9.2 Creando un formulario - etiqueta <form>

La implementación de un formulario requiere un juego de etiquetas **<form>** y **</form>** con el cual elaborar un contenedor de controles. Su formato básico es:

```
<form
   action='url'
   name='nombre del formulario'
   method='tipo de método'>
   <!- contenido del formulario ->
</form>
```

Donde:

- **action**: Determina qué acción se realizará una vez procesado el formulario.
- **name**: Es el nombre que se le asigna a un formulario.
- **method**: Determina qué método se usará para la comunicación con otros documentos web.

Ejemplo 1

A continuación, se presenta el código para implementar un formulario haciendo uso de los atributos básicos.

```
<form name='frmbasico' method='post' action='basico.php' >
   Escribe tu nombre:
   <input type='text' name='nombre' value="" /><br/>
   <input type='submit' value='enviar' />
</form>
```

Como se observa en el código, se asigna el nombre de **frmbasico** al formulario. Este permitirá registrar a un usuario en línea. El nombre comienza con **frm** por una convención de nombres usada para los controles, pero no es necesario seguir estrictamente este patrón, eso dependerá de cada diseñador. Lo que sí se recomienda es usar nombres que representen al control de manera correcta y entendible.

El método **post** envía información, de forma directa, desde el formulario hacia la página especificada en **action**, es decir, no se mostrará en la URL los valores que transfiere.

El atributo **action** define el destino de los valores obtenidos en el formulario, normalmente es controlado por los archivos de un lenguaje de programación. En este caso, a dicha tarea se le asignó el archivo **basico.php** que pertenece al lenguaje de programación PHP.

9.2.1 *Atributo action*

Este atributo indica qué tipo de acción debe realizar el formulario cuando el cliente presione el botón de envío o **Submit**. Cuando no se especifica ninguna acción, el formulario envía la información obtenida al mismo documento web. Entonces se tiene:

- Si se desea que los datos del formulario se envíen por correo electrónico, la implementación debe realizarse de la siguiente manera:

```
<form action=mailto:direcciondelcorreo@correo.com>
    <!- Implementación de los controles del formulario ->
</form>
```

- Si se desea que los datos del formulario sean enviados a un lenguaje de programación, como PHP, para que este los procese, entonces se debe realizar su implementación de la siguiente manera:

```
<form action='registro.php'>
    <!- Implementación de los controles del formulario ->
</form>
```

9.2.2 *Atributo method*

Este atributo permite indicarle al navegador web cómo serán enviados los datos obtenidos desde el formulario. Los más usados son **get** y **post**, pero existen dos métodos más llamados **put** y **delete**.

A. Método get

Esta definición permite que los valores que sean enviados desde el formulario sean visibles mediante la URL del archivo destino. Asimismo, se caracteriza por colocar un símbolo de interrogación y, seguidamente, los valores que se obtuvieron del formulario mediante una asignación simple. Su implementación se realiza de la siguiente manera:

```
<form action='registro.php' method='get'>
    <!- Implementación de los controles del formulario ->
</form>
```

El formato de la URL destino es el siguiente:

```
http://www.servidor.com/destino?variable1=valor1&variable2=valor2
```

Donde:

- **?:** Es el punto de inicio de los valores con el método **get**.
- **destino**: Es el nombre de la página web que recibirá los valores por medio del método **get**.
- **variable1**: Es la variable que usará la página **destino** para obtener los datos que se necesite del formulario.
- **valor1**: Es el dato asignado a la variable y que puede ser usado por la página web de destino.
- **&:** Sirve para unir dos o más variables cuando estas viajan por el método **get**.

Ejemplo 2

Véase lo siguiente:

```
http://www.servidor.com/registro.php?nombre=luis&apellido=perez
```

En este ejemplo se enviarán los nombres y apellidos por medio del método **get**. Como se puede notar, los valores que fueron obtenidos son mostrados en la URL de la página web de destino. A partir de aquí los datos podrán ser usados dentro del archivo destino. En este caso, **registro.php** podrá usar la información de la manera que crea conveniente.

Cuando no se especifica el atributo **method** en la implementación del formulario, el navegador toma por defecto el método **get**.

B. Método post

A diferencia del método **get**, **post** envía los valores al servidor web para ser obtenidos desde allí por la página de destino. Es importante saber que estos valores no serán visibles por el usuario, es decir, la URL solo mostrará el enlace hacia la página, pero no lo valores obtenidos del formulario. Su implementación debe realizarse de la siguiente manera:

```
<form action='registro.php' method='post'>
   <!- Implementación de los controles del formulario ->
</form>
```

Ejemplo 3

Se verá la misma invocación al archivo **destino** por medio del método **post**:

```
http://www.servidor.com/destino
```

Es recomendable el uso del método **post** cuando se envía información confidencial, como por ejemplo, las claves de correos electrónicos, los números de tarjetas de crédito, etc.

9.2.3 *Atributo id*

Permite asignar un identificador único al elemento asociado al formulario. Se recomienda que este nombre sea exclusivo para todo el documento y que solo pueda ser utilizado para hacer referencia a este elemento en otras instancias.

No es necesario hacer la especificación del atributo **id** si el formulario nunca se asocia a elementos externos. Véase qué usos tiene el atributo **id**:

- Como selector para las hojas de estilo (CSS).
- Como enlace destino para vínculos externos.

- Como medio para referenciar a un elemento particular desde un código script.
- Como nombre de un elemento **object** declarado.

Ahora bien, véase lo siguiente:

```
#estilo{
    background-color:#cfc;
}
```

Si se cuenta con el código CSS anterior y se quiere referenciarlo en un formulario básico, sería de la siguiente manera:

```
<form id='estilo'>
    Escribe tu nombre:
    <input type='text' name='nombre' value="" /><br/>
    <input type='submit' value='enviar' />
</form>
```

9.2.4 *Atributo name*

Permite asignar un nombre al formulario para poder identificarlo y poder acceder a sus elementos cuando se le haga referencia de forma externa. Su formato es:

```
<form id='estilo' name='frmregistro'>
    Escribe tu nombre:
    <input type='text' name='nombre' value="" /><br/>
    <input type='submit' value='enviar' />
</form>
```

Se asignó el nombre **frmregistro** a un determinado formulario. Se recomienda iniciar el nombre con las iniciales **frm** para poder identificar que es un control de tipo formulario.

9.2.5 *Atributo enctype*

Permite configurar la forma en que viajará la información obtenida desde un formulario. Existen varias maneras de realizar aquello. La más usual es enviar la información como texto plano, es decir, de la siguiente forma: **enctype="text/plain"**. Véase la especificación de tipo realizada a un formulario para texto plano:

```
<form
    name='frmregistro'
    action='registro.php'
    method='post'
    enctype='text/plain'>
</form>
```

Ahora se verá una breve explicación de los tipos que se puede configurar desde el atributo **enctype**:

- **application/x-www-form-urlencoded**
  Es el valor por defecto del formulario, es decir, cuando no se especifica este atributo toma dicho valor. Se puede utilizar cuando se envían datos de texto plano. Su implementación se realiza de la siguiente manera:

```
<form
    name='frmregistro'
    action='registro.php'
    method='post'
    enctype='application/x-www-form-urlencoded'>
</form>
```

- **multipart/form-data**
  Es el valor que se usa para archivos diferentes a un texto plano como Microsoft Word, PDF, MP3, etc. Su implementación sería de la siguiente manera:

```
<form
    name='frmregistro'
    action='registro.php'
    method='post'
    enctype='multipart/form-data'>
</form>
```

- **text/plain**
  Es el valor más usado, pues el intercambio de información se realiza estrictamente en texto plano. Su implementación se realiza de la siguiente manera:

```
<form
    name='frmregistro'
    action='registro.php'
    method='post'
    enctype='text/plain'>
</form>
```

9.2.6 *Atributo accept-charset*

Especifica la configuración para el juego de caracteres que usará el formulario, los cuales están en relación con el idioma adoptado por el navegador web. El idioma es identificado mayormente por las tildes o símbolos usados en los textos del formulario.

El uso del atributo **accept-charset** se limita a configurar el juego de caracteres para la información que proviene de los manejadores de base de datos y que va hacia un formulario web. Por ejemplo, sirve

para obtener información desde MySQL y mostrarla en los controles de un formulario. Cuenta con las siguientes configuraciones:

- **UTF-8:** Es el valor por defecto en HTML5.
- **ISO-8859-1:** Es una norma de la ISO (International Organization for Standardization), el cual usa caracteres de la tabla ASCII más 128 caracteres nuevos. En él se encontrará todo el alfabeto latino (a-z, A-Z) y los diacríticos (á-ú, ñ, Ñ, ý, ô, etc).

Su implementación se realiza de la siguiente manera:

```
<form
    name='frmregistro'
    action='registro.php'
    method='post'
    accept-charset='utf-8'>
</form>
```

9.2.7 *Atributo novalidate*

Habilita la no validación de los datos del formulario durante la presentación del mismo. Hay que tener en cuenta que es un atributo lógico, es decir, no tiene asignado un valor, su presencia indica el valor `true` del atributo en el formulario. La implementación de este atributo se realiza de la siguiente manera:

```
<form
    name='frmregistro'
    accept-charset='utf-8'
    action='registro.php'
    method='post'
    enctype='application/x-www-form-urlencoded'
    novalidate>
</form>
```

9.2.8 *Atributo target*

Permite configurar la salida que tendrá el valor devuelto desde un formulario, la cual podría realizarse en una nueva ventana o en la misma. Los valores son similares a los atributos de la etiqueta `a`:

- **_self:** Es el valor por defecto y hace que el resultado se muestre en la misma ventana del formulario.
- **_blank:** El resultado se muestra en una nueva ventana dependiendo del navegador web.

Ejemplo 4

Véase el siguiente caso, donde se configura como valor **target** a **_blank**.

```
<form
    name='frmregistro'
    accept-charset='utf-8'
    action='registro.php'
    method='post'
    enctype='application/x-www-form-urlencoded'
    target='_blank'
    novalidate >
</form>
```

9.2.9 Atributo autocomplete

Muchas veces los datos que se colocan en un formulario se graban automáticamente, de modo que la próxima vez que el usuario los digita estos se autocompletan. Esta situación tiene beneficios, como el hecho de completar valores que se repiten dentro del formulario; y perjuicios, por ejemplo, luego de realizar una compra *online* los datos que se ingresan (como el número de tarjeta o incluso la clave) pueden autocompletarse.



Con el atributo **autocomplete** hay la posibilidad de habilitar e inhabilitar dicha opción dentro de un formulario, su implementación se realiza de la siguiente manera:

```
<form
    name='frmregistro'
    accept-charset='utf-8'
    action='registro.php'
    method='post'
    enctype='application/x-www-form-urlencoded'
    target='_blank'
    autocomplete='off|on'>
</form>
```

Cuando este atributo no es implementado en la construcción del formulario, por defecto, este se habilita y HTML5 le asigna **on** como valor. En el ejemplo anterior, se inhabilita la opción de autocompletar en los valores en el formulario.

9.3 Control de textos para formularios

9.3.1 *La etiqueta <label>*

Representa a un texto simple dentro de un formulario. Es posible colocar componentes propios del formulario dentro de la etiqueta **<label>** con el fin de poder aplicarle un estilo CSS. Su formato es:

```
<label>
    <!- Contenido de la etiqueta->
</label>
```

Ejemplo 5

A continuación, se implementará el encabezado de un formulario de registro de usuarios usando la etiqueta **<label>** para el subtítulo:



Consideraciones:

- Implemente el formulario sin usar tablas para su diseño.
- El título **Formulario de Registro** debe usar la etiqueta de encabezado **h1**.
- Para el subtítulo, use la etiqueta **<label>**.
- Debajo del subtítulo se debe mostrar una línea de color blanco.
- Para todos los casos, use los estilos CSS para dar la mejor presentación posible al formulario de registro.

Códigos:

En el archivo **estilo.css** se coloca el siguiente código:

```
/*estilo para el fondo de la página web*/
body{
    font-family: 'Tahoma';
    font-size:12px;
}
```

```
/*estilo para el formulario*/
#formulario{
    border:solid 2px #b7ddf2;
    background:#b7ddf2;
    margin:0 auto;
    width:400px;
    padding:14px;
}

/*estilo para el titulo*/
h1{
    font-size:16px;
    font-weight:bold;
    margin-bottom:8px;
}

/*estilo para el texto y su linea*/
#linea{
    color:#666666;
    margin-bottom:30px;
    border-bottom:solid 1px #fff;
    padding-bottom:5px;
}

/*estilo para los textos del formulario*/
#label{
    display: block;
    font-weight: bold;
    width: 140px;
    font-size: 12px;
}
```

En el archivo **index.html** se inserta el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <section id='formulario'>
            <form name='frmregistro' method='post' action='principal.html'>
                <h1>Formulario de Registro</h1>
                <label id='label'>Complete el formulario con sus datos</label>
                <p id='linea'> </p>
            </form>
        </section>
    </body>
</html>
```

9.3.2 *Cuadro de texto simple*

En un cuadro de texto simple el usuario podrá interactuar con el formulario, es decir, podrá ingresar datos como: nombres, apellidos, *e-mail* o la clave de acceso para una aplicación web, etc.

La característica principal de este control es recolectar información en una sola línea. En otras palabras, el usuario no podrá presionar **<Enter>** para ingresar valores en la segunda línea. Presionar **<Enter>** en un documento web podría ocasionar una redirección del documento, tenga cuidado cuando lo realice.
Su formato es:

```
<input
   type - 'text'
   name - 'nombre'
   value - 'valor'
   maxlength - 'máxima longitud'
   size - 'tamaño en ancho del control'
   readonly - 'readonly'
   placeholder - 'mensaje de ayuda'
   autofocus
   disabled
/>
```

Donde:

- **type="text":** Indica que la etiqueta **<input>** se comportará como un cuadro de entrada o texto simple.
- **name:** Es el nombre que se le asigna al control de tipo **text.** Se recomienda que las tres primeras letras del nombre sean **txt** para poder diferenciarlo de los demás controles del formulario web. En el siguiente script se implementa el control **text**, y se le asigna el nombre de **txtapellidos** para que el usuario pueda ingresar sus apellidos.

```
<input
   type - 'text'
   name - 'txtapellidos'
/>
```

- **value**: Es el valor que se le puede asignar de forma directa al control de tipo **text**. Normalmente, esto último no debería realizarse, ya que es el usuario quien debe llenar dicho control, pero podría darse el caso en el que se desee mostrar la fecha actual de forma predeterminada, como se muestra a continuación:

```
<input
   type - 'text'
   name - 'txtfecha'
   value - '28 de julio del 2018'
/>
```

- **maxlength:** Especifica el número máximo de caracteres que puede contener el campo de texto. No se debe confundir esto con la configuración de ancho que tiene el mismo control. Cuando no se especifica este atributo, el límite de caracteres que se pueden introducir es infinito. Esta configuración sirve para que el usuario ingrese valores exactos; por ejemplo, como se muestra a continuación, para el número de DNI:

```
<input
type='text'
name='txtdni'
maxlength='8'
/>
```

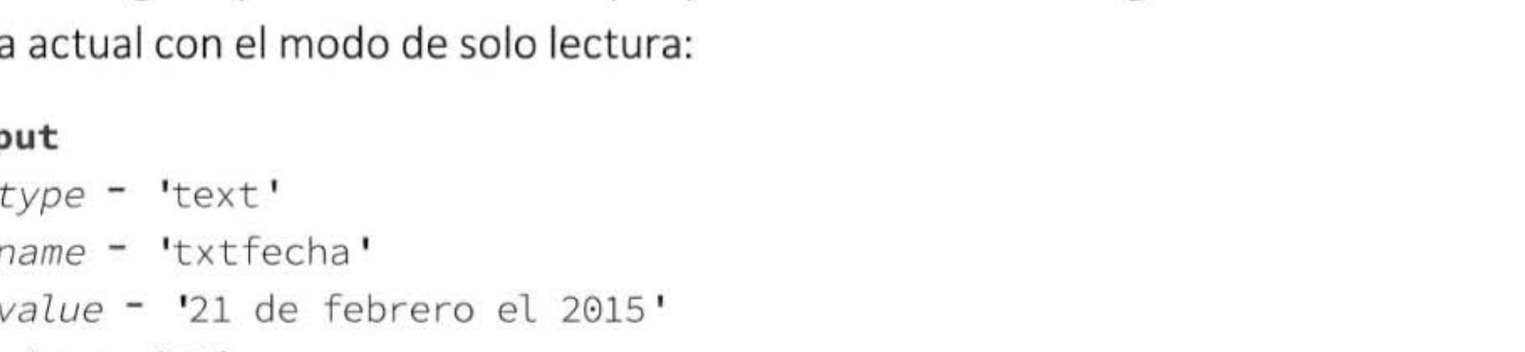
- **size:** Especifica el ancho que puede tener el control `text`. Esta medida se registra en píxeles, no en cantidad de caracteres que puede contener el control. A continuación, la implementación del control `text` para el ingreso de apellidos y nombres del usuario:

```
<input
   type = 'text'
   name = 'txtnombres'
   size = '100'
/>
```

- **readonly:** Aplica la característica de «solo lectura» al control `text`. La idea es mostrar un valor que no admita ningún tipo de modificación por parte del usuario. En seguida se mostrará cómo colocar la fecha actual con el modo de solo lectura:

```
<input
   type = 'text'
   name = 'txtfecha'
   value = '21 de febrero el 2015'
   size = '80'
   readonly = 'readonly'
/>
```

- **placeholder:** Permite mostrar un texto de apoyo que le muestre al usuario un ejemplo del valor correcto que debe ingresar en el control `text` del formulario. Se manifiesta como una anotación de color gris que puede ser modificada por el usuario. A continuación, se muestran los textos de apoyo del DNI y del teléfono:



```
<input
    type='text'
    name='txtdni'
    size='60'
    placeholder='8 caracteres'/>
<input
    type='text'
    name='txtfono'
    size='60'
    placeholder='(123)-123456789'/>
```

› **autofocus:** Permite activar el cursor del mouse sobre el control especificado. Hay que tener en cuenta que solo uno de los controles puede tener el `autofocus`.




```
<input
    type - 'text'
    name - 'txtdni'
    size - '60'
    placeholder - '8 caracteres'
    autofocus
/>
```

› **disabled:** Permite inhabilitar el acceso a los valores mostrados en un control `text`. Normalmente, se utiliza para bloquear el acceso del usuario, y aparece un color gris sobre el control.




```
<input
    type - 'text'
    name - 'txtdni'
    size - '60'
    placeholder - '8 caracteres'
    disabled
/>
```

- **required:** Muestra un mensaje al usuario cuando este no ingresa ningún valor requerido dentro de un control `text`. Esta opción se caracteriza porque obliga al usuario a digitar un valor. Hay que tener en cuenta que el mensaje se mostrará cada vez que se intente enviar el formulario con información incompleta.




```
<input
    type - 'text'
    name - 'txtnombres'
    size - '60'
    placeholder - 'Ingrese datos completos'
    required
/>
```

Adicionalmente se puede agregar un texto al mensaje, como por ejemplo:




```
<input
    type - 'text'
    name - 'txtnombres'
    size - '60'
    placeholder - 'ingrese datos completos'
    title - 'nombres y apellidos'
    required
/>
```

Ejemplo 6

Implemente el siguiente modelo de formulario:



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body {
    font-family: 'tahoma';
    font-size: 14px;
}

form {
    padding: 10px;
    margin: 0;
    width: 250px;
    border: 1px solid darkgray;
}

#entrada {
    font-family: Tahoma;
    font-size: 14px;
    width: 240px;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='utf-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta content='width=device-width, initial-scale=1.0' name='viewport'>
    </head>
    <body>
        <form>
            <table>
                <tr>
                    <td>REGISTRO DE USUARIOS</td>
                </tr>
                <tr>
```

```
            <td>
                <input id='entrada' placeholder='Ingrese su correo electrónico'
                       type='text'></input>
            </td>
        </tr>
        <tr>
            <td><a href='#'>Inscribirse</a></td>
        </tr>
    </table>
</form>
</body>
</html>
```

9.3.3 *Cuadro de texto múltiple*

Permite implementar un cuadro de texto de múltiples líneas, el cual puede contener un número ilimitado de caracteres. Asimismo, el número de líneas puede ser especificado por los atributos del control o también por medio de estilos CSS. Su formato es:

```
<textarea
    name - 'nombre'
    cols - 'valor'
    rows - 'valor'
    readonly - 'readonly'
    maxlength - 'valor'
    placeholder - 'mensaje de ayuda'
    autofocus
    disabled
    required>
</textarea>
```

Donde:

- **cols**: Especifica el ancho visible para el control `<textarea>`.
- **rows**: Determina el número de filas que puede contener el control `<textarea>`. Se debe tener en cuenta que esto generará una barra de desplazamiento en el control.

Ejemplo 7

A continuación se verá un script que permite implementar un control **<textarea>** de 60 de ancho con 10 líneas de filas:




```
<textarea
    name - 'txtreseña'
    cols - '50'
    rows - '10'>
</textarea>
```

Ejemplo 8

Ahora bien, implemente el siguiente modelo de formulario básico:



Códigos:

En el archivo **estilo.css** se agrega el siguiente código:

```
body {
    font-family: 'tahoma';
    font-size: 14px;
}

form {
    padding: 10px;
    margin: 0;
    width: 250px;
    border: 1px solid darkgray;
}

#entrada {
    font-family: Tahoma;
    font-size: 14px;
    width: 240px;
}

textarea {
    width: 240px;
    height: 144px;
    border-style: solid;
    border-width: 1px;
    background-color: #E7E7E7;
}
```

En el archivo **index.html** se coloca el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta charset='UTF-8'>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
    </head>
    <body>
        <form>
            <table>
                <tr>
                    <td>
                        FORMULARIO DE CONTACTO<br><br>
                    </td>
                </tr>
```

```
            <tr>
                <td>
                    Nombre completo<br>
                    <input type='text' id='entrada'
                           placeholder='Ingrese nombre' required>
                </td>
            </tr>
            <tr>
                <td>
                    Correo Electrónico<br>
                    <input type='text' id='entrada'
                           placeholder='Ingrese correo' required>
                </td>
            </tr>
            <tr>
                <td>
                    Teléfono<br>
                    <input type='text' id='entrada'
                           placeholder='Ingrese teléfono'>
                </td>
            </tr>
            <tr>
                <td>
                    Mensaje<br>
                    <textarea></textarea>
                </td>
            </tr>
        </table>
    </form>
</body>
</html>
```

9.3.4 *Cuadro de password*

El control **password** tiene las mismas características del control **text**, pero se diferencia del primero en que el valor ingresado no será visible para el usuario. En lugar de los caracteres comunes, aparecen unos símbolos dependiendo del navegador web. Su formato es:

```
<input
    type - 'password'
    name - 'nombre'
    value - 'valor'
    maxlength - 'máxima longitud'
    size - 'tamaño en ancho del control'
    readonly - 'readonly'
```

```
    placeholder - 'mensaje de ayuda'
    autofocus
    disabled
/>
```

Ejemplo 9

Se verá el script que permite implementar el siguiente registro de una clave personal:




```
<input
    type - 'password'
    name - 'txtclave'
    size - '60'
    placeholder - 'Máximo 6 caracteres'
/>
```

9.3.5 Entrada de teléfono

En HTML5 se ha implementado la etiqueta **tel**, que permite tener un control de los números telefónicos registrados en un formulario. Cuando la página es vista desde un *smartphone*, la pantalla se convierte en un teclado numérico semejante al del teléfono.



Seguidamente, el código de la implementación de la entrada de teléfono:

```
<input
    type - 'tel'
    size - '60'
    name - 'txtfono'
/>
```

9.3.6 *Entrada de correo electrónico*

El tipo **email** permite validar el ingreso de un correo electrónico en un cuadro de texto simple. En los dispositivos móviles es usado con mayor frecuencia. Estos muestran un teclado en el que se agrega el símbolo @.



A continuación, la implementación del correo electrónico en un formulario:




```
<input
    type - 'email'
    size - '60'
    name - 'txtemail'
/>
```

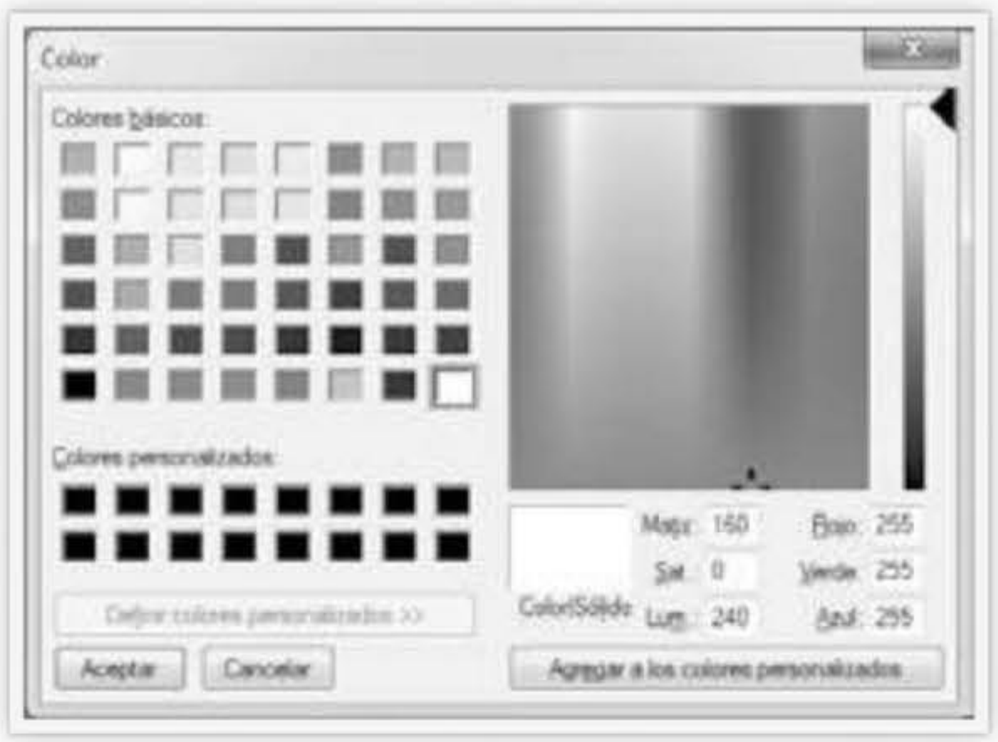
9.3.7 *Entrada de color*

La entrada de color muestra una paleta de colores y permite seleccionar uno de ellos para usarlo en un lenguaje de programación.




```
<input
    type - 'color'
    name - 'txtcolor'
    value - '#ffffff'
/>
```

Al presionar sobre el botón de selección de color aparecerá una ventana de diálogo. En ella podrá seleccionar el color de su preferencia, que se configurará como el color activo del control.



Hay que tener en cuenta que el color inicial es blanco, pero se puede modificar usando la propiedad **value** como se muestra en el script anterior.

9.3.8 *Entrada de fecha*

Esta entrada muestra un selector de fecha y un campo para ingresar texto. Dicho campo contiene separadores para fecha, horas, minutos y segundos. Se debe tener en cuenta que solo se puede seleccionar un solo valor a la vez.



El código será:

```
<input
    type - 'date'
    name - 'txtfecha'
/>
```

El tipo de entrada de fecha tiene las siguientes configuraciones:

› **datetime**: Permite al usuario ingresar la fecha y la hora en un mismo cuadro de texto simple.



Su formato es:

```
<input
    type='datetime-local'
    name='txtfecha'
/>
```

- **month**: Permite seleccionar cualquiera de los meses del año que han sido almacenados, los cuales se muestran como números que van desde el 1 hasta el 12. Cada navegador proporciona distintos mecanismos de selección para dichos meses.



Su formato es:

```
<input
    type - 'month'
    name - 'txtfecha'
/>
```

- **week**: Permite elegir una semana almacenada internamente en el formato 2010-W37 (semana 37 del año 2010).



Su formato es:

```
<input
    type - 'week'
    name - 'txtfecha'/>
```

9.3.9 *La etiqueta <number>*

Permite tener el control de los valores numéricos mediante una barra de desplazamiento que aparece en el lado derecho del control `<input>`. Su formato es:

```
<input
    type - 'number'
    name - 'nombre del control'
    min - 'valor'
    max - 'valor'
    step - 'valor'
    placeholder - 'leyenda del control numerico'
    value - 'valor por defecto'
    disabled
    autofocus
    readonly
    required
/>
```

Donde:

- **type="number":** Define la etiqueta de entrada en forma numérica.
- **min**: Establece el mínimo valor numérico que puede mostrar la etiqueta.
- **max**: define el máximo valor numérico que puede mostrar la etiqueta.
- **step**: Establece la forma de avance entre los números definidos en el mínimo y máximo.

Ejemplo 10

A continuación, se muestra el script HTML5 que permitirá implementar un control **<input>**, en este caso, para seleccionar la edad de una persona. De manera obligatoria, solo se podrá elegir edades en el rango de 18 a 65 años:




```
<form name='frmregistro' method='post' action='principal.html'>
   <label> registre su edad: </label>
   <input name - 'txtedad'
          type - 'number'
          min - '18'
          max - '65'
          value - '18'
          required>
</form>
```

Ejemplo 11

En el siguiente ejemplo se ve el cálculo de la edad una persona, según el año de nacimiento ingresado en un control **<input>** de tipo **number**. El formato es el siguiente:




```
<form oninput='resultado.value=2014-parseInt(txtaño.value)'>
   <label>Año de nacimiento
      <input name='txtaño' type='number' size='60' placeholder='Ingrese año
            de nacimiento' required/>
   </label>
   <label>Su edad es:
      <output name='resultado'></output>
   </label>
</form>
```

Donde:

La etiqueta **<form>** implementa el atributo **<oninput>**, que permite obtener los valores asignados a la etiqueta de tipo **number** y los opera mediante una expresión de cálculo simple. Ahí mismo debe especificarse dónde se mostrará el resultado.

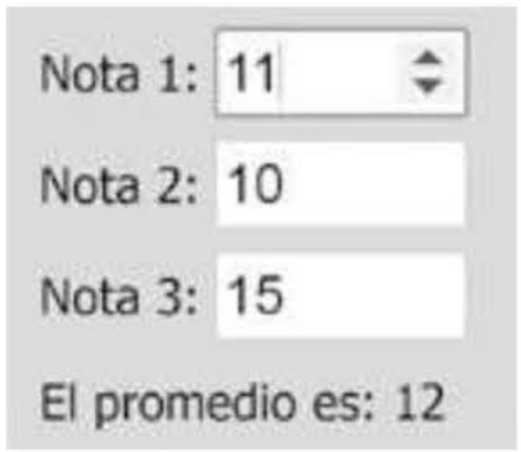
Hay que considerar que, en el formulario, el valor solicitado para el cálculo debe ser numérico entero; para ello se debe usar la función **parseint**, la cual permitirá convertir un valor textual en un número entero.

Finalmente, se usará la etiqueta **<output>** con **name="resultado"**, en la cual se muestra el valor del cálculo en tiempo real, es decir, a medida que se escriban los valores se va obteniendo el resultado.

Debe tenerse en cuenta que la respuesta se visualiza gracias a la definición que se realizó en la etiqueta **<form>**, de lo contrario no funcionaría.

Ejemplo 12

A continuación, el último caso de implementación de **number**, donde se obtendrá el promedio de tres notas. El formato es el siguiente:



```
<form oninput='resultado.value = (parseInt(txtn1.value) + parseInt(txtn2.value)
                + parseInt(txtn3.value))/3'>
   <label id='label'>Nota 1:
      <input name='txtn1' type='number' min='1' max='20' placeholder='n1' />
   </label>
   <label id='label'>Nota 2:
      <input name='txtn2' type='number' min='1' max='20' placeholder='n2' />
   </label>
   <label id='label'>Nota 3:
      <input name='txtn3' type='number' min='1' max='20' placeholder='n3' />
   </label>
   <label id='label'>El promedio es:
      <output name='resultado'></output>
   </label>
</form>
```

9.3.10 *La etiqueta <URL>*

Permite validar el ingreso de una dirección URL correcta. Si el formulario es visualizado en un dispositivo móvil, se mostrará un teclado Querty.



Su formato es:

```
<input
   type='url'
   name='nombre del control'
   placeholder='leyenda del control url'
   value='valor por defecto'
   disabled
   autofocus
   readonly
   required
/>
```

9.3.11 *La etiqueta <range>*

Permite implementar un slider, también llamado "barra con botón deslizante", el cual permite seleccionar un valor que se ubica entre otros dos que han sido previamente especificados. Su formato es:

```
<label>min
   <input
      type='range'
      name='nombre del control range'
      min='minimo valor del rango'
      max='maximo valor del rango'
   />
   max
</label>
```

Ejemplo 13

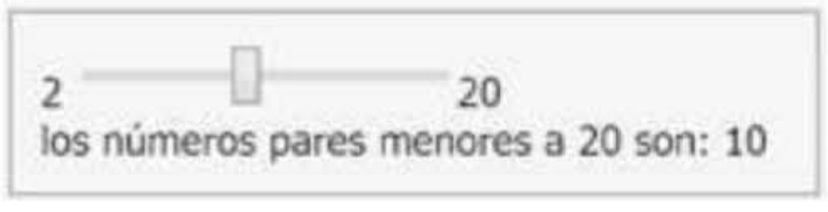
A continuación, se verá cómo se realiza la implementación de un formulario que permite sumar dos números enteros. El primero deberá ser registrado mediante el control **input** tipo **range**; el segundo, con se observa en el siguiente formato:




```
<form oninput='resultado.value = parseInt(ran1.value) + parseInt(txtn.value)'>0
   <input type='range' name='ran1' min='0' max='100'>100 +
   <input type='number' name='txtn' min='0' max='100' value='0'>=
   <output name='resultado'>0</output>
</form>
```

Ejemplo 14

Ahora se verá cómo se realiza la implementación de un formulario que permita, desde control **input** tipo **range**, mostrar los números pares menores a veinte. El formato es el siguiente:




```
<form oninput='resultado.value=ran.value'>2
   <input type='range' name='ran' min='2' max='20' step='2'>20
   <br/>los números pares menores a 20 son:
   <output name='resultado' for='ran'>2</output>
</form>
```

9.4 Control de botones para formularios

Si se tiene en cuenta que el usuario ingresa información por medio de los controles del formulario, entonces el botón es el encargado de enviar la información. Un botón es un control usado que sirve para realizar una acción en el formulario que lo contenga.

Las principales propiedades son:

- **Type**: Es la implementación del tipo de botón que se debe mostrar al usuario. Se pueden dar tres casos:
  - Button
  - Reset
  - Submit
- **Value:** Es el texto que se muestra como contenido del botón. Por cuestiones de comodidad, es importante que dicho texto no sea muy extenso.
  Existen tres tipos básicos de botones:
- **Botón de envío**: Es el botón que muestra información al usuario como resultado de una operación.




```
<form method='post' action='url destino'>
   <input
      type='submit'
      value='texto mostrado'
   />
</form>
```

Aquí debe tenerse en cuenta que el resultado se enviará a una URL destino especificada en la propiedad **action** del formulario; en caso no se especificara el destino del resultado, se enviará al mismo formulario.

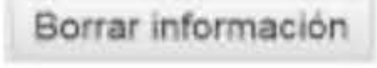
Ejemplo 15

Véase la implementación de un botón que permite enlazar a la página web <http://www.google.com> usando el tipo **submit** con el siguiente formato:



```
<input
    type='submit'
    value='buscar'
    onclick='document.location='http://www.google.com.pe''
/>
```

- **Botón de reinicialización**: Es el botón que permite restablecer todos los controles del formulario, es decir, "limpia" el formulario.



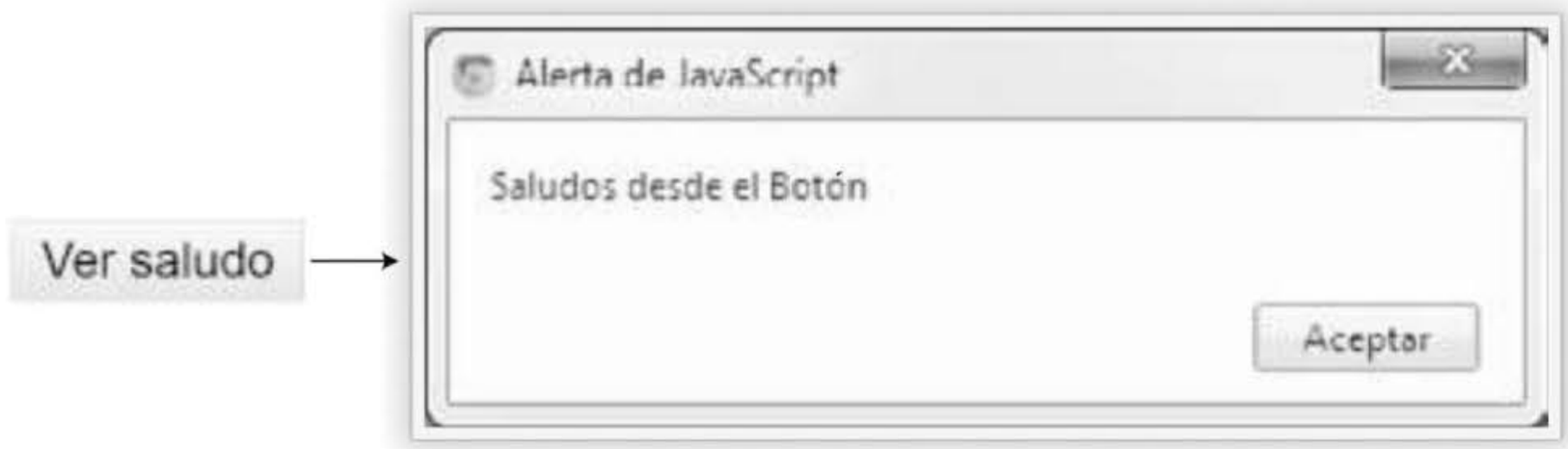

```
<form method='post' action='url destino'>
    <input
        type='reset'
        value='texto mostrado'
    />
</form>
```

- **Botón pulsador**: Es el botón que no tiene un comportamiento predefinido (como el botón de envío o el de reinicialización). Este botón responde al código que se le asigne, ya sea desde Javascript o PHP.

```
<form method='post' action='url destino'>
    <input
        type='button'
        value='texto mostrado'
    />
</form>
```

Ejemplo 16

A continuación, se muestra la implementación de un botón llamado **Ver Saludo** que, al ser presionado, permita mostrar el mensaje **Saludos desde el Botón**. El formato es el siguiente:




```
<input
    type='button'
    onclick='javascript:alert('Saludos desde el Botón')'
    value='Ver saludo'
/>
```

Ejemplo 17

Ahora se implementará un botón que permita enlazar al usaurio a la página web <http://www.google.com>, usando una función Javascript con el siguiente formato:




```
<script>
    function ira() {
        document.location='http://www.google.com.pe';
    }
</script>
<input
    type='button'
    value='buscar'
    onclick='ira();'
/>
```

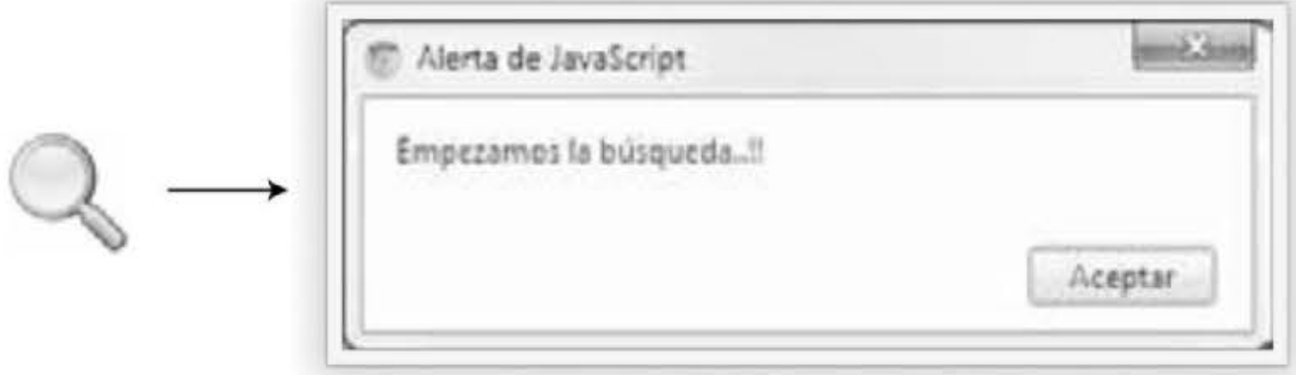
9.4.1 Usando imágenes para botones

A veces los botones suelen ser muy comunes entre los diferentes formularios, lo que se puede hacer es mostrar una imagen en vez de un botón rectangular sin que ello suponga una variación en los atributos. Su formato es:

```
<form method='post' action='url destino'>
    <input
        type='image'
        src='ruta de la imagen'
    />
</form>
```

Ejemplo 18

A continuación, se verá la implementación de un botón de tipo imagen que permita mostrar el archivo **lupa.jpg** guardado en la carpeta **imágenes**. Al presionar el botón de búsqueda deberá mostrar el mensaje **Empezamos la búsqueda**. Para ello, se empleará el código Javascript y el formato es como sigue:




```
<input type='image'
   src='imagenes\indice.jpg'
   onclick='javascript:alert('Empezamos la búsqueda..!!')'
/>
```

9.4.2 La etiqueta <output>

Representa la salida que puede tener un cálculo realizado a través de un script en un determinado formulario. Su formato es:

```
<output
   name='nombre de la variable de salida'
   for 'variable de referencia'>
</output>
```

Donde:

- **name**: Permite asignar un nombre al elemento **<output>**. Este atributo es muy importante, pues de aquí se determinarán los resultados del cálculo. También hay que considerar que las mayúsculas y las minúsculas son diferenciadas unas de las otras.
- **for**: Establece una relación entre el resultado del cálculo y los elementos del mismo. Aquí se debe especificar los nombres de los controles involucrados en el cálculo, considere que las mayúsculas y minúsculas son consideradas como elementos diferentes.

Ejemplo 19

A continuación, se muestra la implementación de una multiplicación entre dos elementos numéricos, para lo cual se usará el elemento **<output>** con el siguiente formato:



```
<form
   oninput='res.value=parseint(op1.value) * parseint(op2.value)'>
   <input name='op1' type='number'> x
   <input name='op2' type='number'> =
   <output
      name='res'
      for 'op1 op2'>
   </output>
</form>
```

9.5 Control de cuadros de chequeo

El control de cuadro de chequeo sirve para seleccionar una o más opciones dentro de un grupo que se presenta en un documento web. Su formato es:

```
<input
   type='checkbox'
   name='nombre del control'
/> mensaje a mostrar
```

Ejemplo 20

A continuación, se verá la implementación de cuatro cuadros de chequeo usando el **input** de tipo **checkbox**. El formato es el siguiente:




```
<label>
   ESTUDIOS REALIZADOS
</label>
<label>
   <input checked='checked' name='chkprimaria' type='checkbox'/> Primaria
</label>
<label>
   <input name='chksecundaria' type='checkbox'/> Secundaria
</label>
<label>
   <input name='chktecnico' type='checkbox'/> Técnico
</label>
<label>
   <input name='chkuniversitario' type='checkbox'/> Universitario
</label>
```

9.6 Control de botones de radio

Un botón es un control usado para realizar una acción con el formulario que lo contenga. Existen tres tipos básicos de botones:

```
<input type='radio'
   name='nombre del botón'
   value='valor del botón'
/> texto mostrado
```

Donde:

- **type="radio":** Es la definición del tipo botón de radio dentro de un formulario.
- **name**: Es el nombre que se le asigna al control de tipo opción. Normalmente, las tres primeras letras son "opt" y en seguida el nombre que representa al control.
- **value**: Es la definición del valor al seleccionar una opción. Este será usado por algún lenguaje de programación.

Ejemplo 21

Seguidamente, se muestra el formato y el script:




```
<label id='label'>Estado civil</label>
<label id='label'>
   <input type='radio' name='btnestado' value='s' checked='checked'/> Soltero
</label>
<label id='label'>
   <input type='radio' name='btnestado' value='c' /> Casado
</label>
<label id='label'>
   <input type='radio' name='btnestado' value='v' /> Viudo
</label>
<label id='label'>
   <input type='radio' name='btnestado' value='d' /> Divorciado
</label>
```

A diferencia de los cuadros de chequeo, el nombre de un grupo de botones siempre será el mismo. En el script anterior se puede apreciar que todos tienen el nombre de **btnestado**, esto se debe a que el usuario puede escoger una sola opción de la lista de botones.

9.7 Control de cuadro combinado

El control de cuadro combinado se caracteriza porque puede presentar muchas opciones en un solo control, a las cuales se puede acceder por medio de una lista desplegable. Su formato es:

```
<select name='nombre del control'>
   <option value='valor 1' selected> opción 1 </option>
   <option value='valor 2'> opción 2</option>
   <option value='valor 3'> opción 3</option>
</select>
```

Donde:

- **name**: Es el nombre que se le da al control de cuadro combinado. Normalmente, se le asigna "sel" como las tres primeras letras ; seguidamente, se escribe nombre que representa.
- **option value**: Es el valor mostrado en una lista desplegable.
- **selected**: Define cuál de los valores se mostrará inicialmente al usuario.

Ejemplo 22

A continuación, se verá la implementación de selección de ciudades del Perú, para lo cual se deberá usar el control **select** con el siguiente formato:




```
<label id='label'>Ciudad</label>
<label id='label'>
   <select name='selciudad'>
      <option value='lima' selected>Lima</option>
      <option value='arequipa'>Arequipa</option>
      <option value='tacna'>Tacna</option>
      <option value='trujillo'>Trujillo</option>
      <option value='huancayo'>Huancayo</option>
   </select>
</label>
```

9.8 Control de cuadro de lista

Este control es parecido al anterior, pero se diferencia en que los elementos que componen la lista son totalmente visibles, no como el cuadro combinado que los oculta y solo muestra el predefinido. Su formato es:

```
<select name='nombre del control' size='valor'>
   <option value='valor 1' selected> opcion 1 </option>
   <option value='valor 2'> opcion 2</option>
   <option value='valor 3'> opcion 3</option>
</select>
```

Donde:

- **size="valor"**: Aquí se especifica el número de líneas que ocupará el control cuadro de lista, el mínimo valor es cuatro. Considere que, al especificar un valor, este será el número de elementos a mostrar en el cuadro de lista, y aparecerá una barra de desplazamiento vertical que permitirá observar los demás elementos.

Ejemplo 23

A continuación, se verá cómo se implementa una selección de ciudades del Perú, para lo cual se deberá usar el control **select** con un tamaño inicial de cuatro elementos. El formato es el siguiente:




```
<label id='label'>Ciudad</label>
<label id='label'>
   <select name='selciudad' size='4'>
      <option value='lima' selected>Lima</option>
      <option value='arequipa'>Arequipa</option>
      <option value='tacna'>Tacna</option>
      <option value='trujillo'>Trujillo</option>
      <option value='huancayo'>Huancayo</option>
   </select>
</label>
```

Si la especificación de tamaño fuera cinco, el diseño sería de la siguiente manera:



9.9 La etiqueta <datalist>

En HTML5 un **<datalist>** permite automatizar el ingreso de datos. Con ello se otorga una nueva opción

para mostrar distintos valores en un mismo control, esta vez usando el control **input** de tipo **text**. Este generará una lista desplegable de opciones para que el usuario realice su selección del mismo modo que con el control de cuadro combinado. Su formato es:

```
<input type='text' name='nombre del control input text' list='nombre de lista'/>
<datalist id='nombre de lista'>
   <option value='valor 1' />
   <option value='valor 2' />
   <option value='valor 3' />
</datalist>
```

En caso de que el navegador web no soportara esta característica propia del HTML5, mostrará un control **input** sencillo, sin desplegar las opciones de la lista. La principal ventaja de este control es que reconoce los primeros caracteres ingresados por el usuario, para autocompletar la información y así poder seleccionar una de las opciones.

Ejemplo 24

A continuación, se verá cómo se implementa una selección de ciudades del Perú, para lo cual se deberá usar la etiqueta **<datalist>** con el siguiente formato:




```
<label id='label'>Provincia</label>
<input type='text' name='txtprovincia' list='provincias'/>
<datalist id='provincias'>
   <option value='Lima' />
   <option value='Arequipa' />
   <option value='Tacna' />
   <option value='Trujillo' />
   <option value='Huancayo' />
</datalist>
```

Caso desarrollado
Formulario de registro

Implemente el siguiente formulario de registro usando todos los controles desarrollados.



Consideración:

No use la tabla para la tabulación de los elementos, sino los estilos CSS.

Solución:

1. Cree el archivo **estilo.css** con el siguiente código:

```css
/*Estilo para el fondo de la página web*/
body {
    font-family: 'tahoma';
    font-size: 12px;
}

/*Estilo para el formulario*/
#formulario {
    border: solid 2px #b7ddf2;
    background: #b7ddf2;
    margin: 0 auto;
    width: 420px;
    padding: 10px;
}
```

```
/*Estilo para el titulo*/
h1 {
    font-size: 16px;
    font-weight: bold;
    margin-bottom: 2px;
}

/*Estilo para el texto y su linea*/
#linea {
    color: #666666;
    margin-bottom: 5px;
    border-bottom: solid 1px #FFF;
    padding-bottom: 5px;
}

/*Estilo para los textos del formulario*/
#label {
    display: block;
    font-weight: bold;
    width: 240px;
    font-size: 12px;
}

/*Estilo para las cajas de texto simple*/
#caja {
    padding: 4px;
    border: solid 1px #aacfe4;
    margin: 2px 0 5px 0;
}
/*Estilo para el botón Registrar*/
#boton {
    clear: both;
    margin-left: 150px;
    width: 125px;
    height: 31px;
    background: #666666;
    text-align: center;
    line-height: 31px;
    color: #FFFFFF;
    font-size: 11px;
    font-weight: bold;
}
```

2. Cree el documento web **index.html** con el siguiente código:

```
<!DOCTYPE html>
<html>
    <head>
    <meta charset='UTF-8'>
        <link href='css/estilo.css' rel='stylesheet' type='text/css'/>
        <meta name='viewport' content='width=device-width, initial-scale=1.0'>
```

```
</head>
<body>
   <section id='formulario'>
      <form name='frmregistro' method='post' action='principal.html'>
         <h1>FORMULARIO DE REGISTRO</h1>
         <label id='label'>Complete el formulario con sus datos</label>
         <p id='linea'> </p>
         <label id='label'>Nombres y apellidos
            <input name='txtnombres' type='text' id='caja' size='60'
                   placeholder='ingrese datos completos' required/>
         </label>
         <label id='label'>DNI
            <input type='text' id='caja' name='txtdni'
                   size='60' placeholder='8 caracteres' disabled/>
         </label>
         <label id='label'>Telefono
            <input type='tel' id='caja' size='60' name='txtfono'/>
         </label>
         <label id='label'>Estudios realizados</label>
         <label id='label'>
            <input type='checkbox' name='chkprimaria' checked='checked'/> Primaria
         </label>
         <label id='label'>
            <input type='checkbox' name='chksecundaria' /> Secundaria
         </label>
         <label id='label'>
            <input type='checkbox' name='chktecnico' /> Técnico
         </label>
         <label id='label'>
            <input type='checkbox' name='chkuniversitario' /> Universitario
         </label>
         <br/>
         <label id='label'>Estado civil</label>
         <label id='label'>
            <input type='radio' name='btnestado' value='s'
                   checked='checked'/> Soltero
         </label>
         <label id='label'>
         <input type='radio' name='btnestado' value='c' /> Casado
         </label>
         <label id='label'>
         <input type='radio' name='btnestado' value='v' /> Viudo
         </label>
         <label id='label'>
            <input type='radio' name='btnestado' value='d' /> Divorciado
         </label>
         <br/>
         <label id='label'>Provincia</label>
         <input type='text' name='txtprovincia' list='provincias'/>
```

```
<datalist id='provincias'>
   <option value='Lima' />
   <option value='Arequipa' />
   <option value='Tacna' />
   <option value='Trujillo' />
   <option value='Huancayo' />
</datalist>
<label id='label'>Correo electrónico
   <input type='email' id='caja' size='60' name='txtemail'/>
</label>
<label id='label'>Clave personal
   <input type='password' id='caja' name='txtclave'
          size='60' placeholder='Máximo 6 caracteres '/>
</label>
<label id='label'>Color de preferencia
   <input type='color' name='txtcolor'/>
</label>
<label id='label'>Fecha de nacimiento
   <input type='datetime-local' name='txtfecha'/>
</label>
<input type='submit' value='registrar'>
<input type='reset' value='limpiar'>
</form>
</section>
</body>
</html>
```

9.10 Control de archivos

Permite seleccionar cualquier tipo de archivo usando como fuente las unidades reconocidas por el sistema operativo del cliente. Normalmente, es usado para seleccionar algún archivo que será enviado por medio de un lenguaje de programación hacia el servidor. Su formato es:

```
<input
   type='file'
   multiple
/>
```

Ejemplo 25

Se implementará un formulario que permita seleccionar archivos de diferentes tipos desde la unidad de una computadora. Dicho formulario, además, debe mostrar los nombres de los archivos seleccionados, el tipo de archivo y el tamaño. Para ello se usará la etiqueta **input** de tipo **file**. Inicialmente, el formulario deberá mostrarse de la siguiente manera:



Los archivos seleccionados desde su dispositivo deberán mostrarse de la siguiente manera:

Elegir archivos 5 archivos

- Ordenamiento y filtros.xlsx–(**application/vnd.openxmlformats-officedocument.spreadsheetml.sheet**) ->7867
- Introducción.avi–(**video/avi**) ->0
- index.html–(**text/html**) ->510
- salir.gif–(**image/gif**) ->3560
- MasterIn.ppt–(**application/vnd.ms-powerpoint**) ->203264

El script HTML5 que permite crear los controles es:

```
<section>
   <input type='file' id='files' multiple>
   <article id='archivos'>
      <ul></ul>
   </article>
<section>
<script>
   var insertar_en = document.querySelector('#archivos ul');
   file_in = document.querySelector('#files')
   file_in.onchange = function(e){
      var files = e.target.files;
      for(var i=0,f;f = files[i];++i){
         var archivo = document.createElement('li');
         archivo.innerHTML = f.name + '–(<b>' + f.type + '</b>) ->' + f.size;
         insertar_en.appendChild(archivo);
      }
   }
</script>
```

9.11 Control de datos ocultos

Este control es parecido al **input** de tipo **text**, pero no es visible para el usuario. Es un dato que se mantiene oculto a la vista, pero que igualmente se transmite al servidor, junto con los otros campos del formulario.

Se usa generalmente cuando se necesita enviar información de un documento web a otro sin que sea visible por el usuario. Esto ayudará a pasar información que será de vital importancia en los subsiguientes documentos web. Su formato es:

```
<input
   type='hidden'
   name='txtoculto'
/>
```

Referencias bibliográficas



Van, L. (2013). *HTML5 Y CSS3. Domine los estándares de las aplicaciones web*. 2.ª edición. Barcelona: ENI.

Goldstein, A. *et al*. (2011). *HTML5 y CSS3. Manual Imprescindible.* Madrid: Anaya Multimedia.

Meloni, J. (2012). *HTML5, CSS3 y JavaScript. Programación*. Madrid: Anaya Multimedia.

Impreso en los talleres gráficos de



EDITORIAL
MACRO®

Surquillo

Abril 2018