

**Desarrollo de aplicaciones web con**

# **PHP**



---

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



## **Desarrollo de aplicaciones web con PHP**

Autor: Manuel Ángel Torres Remon

© Derechos de autor registrados:

Empresa Editora Macro EIRL

© Derechos de edición, arte gráfico y diagramación reservados:

Empresa Editora Macro EIRL

Coordinación de edición:

Cynthia Arestegui Baca

Diseño de portada:

Alejandro Marcas León

Corrección de estilo:

Milton A. Gonzales M.

Diagramación:

Lizbeth R. Eufracio Quispe

Edición a cargo de:

© Empresa Editora Macro EIRL

Av. Paseo de la República N.° 5613 , Miraflores, Lima, Perú

☎ Teléfono: (511) 748 0560

✉ E-mail: [proyecto@editorialmacro.com](mailto:proyecto@editorialmacro.com)

🌐 Página web: [www.editorialmacro.com](http://www.editorialmacro.com)

Primera edición: diciembre de 2014

Tiraje: 1000 ejemplares

**Impresión**

Talleres gráficos de la Empresa Editora Macro EIRL

Jr. San Agustín N.° 612-624, Surquillo, Lima, Perú

ISBN N.° 978-612-304-248-6

Hecho el depósito legal en la Biblioteca Nacional del Perú N.° 2014-16719

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

## Manuel Ángel Torres Remon

El licenciado Manuel Ángel Torres Remon estudió en el Instituto de Educación Superior Manuel Arévalo Cáceres y en la Universidad Alas Peruanas. Actualmente se desempeña como consultor tecnológico de empresas como TopyTop y Nestlé. Asimismo, imparte los cursos de Programación, Análisis y Diseño de Sistemas y Base de Datos en instituciones superiores, como el instituto Manuel Arévalo Cáceres, Cibertec y Unimaster, de la Universidad Nacional de Ingeniería.

Ha publicado los libros de programación *Desarrollo de aplicaciones con Java*, *Programación orientada a objetos con Visual Basic 2012*, *Fundamentos de programación con Visual Basic 2012*, *Programación Transact con SQL Server 2012* y *Diseño web con HTML5 y CSS3*.

Para cualquier duda o sugerencia sobre el material, le puede escribir al siguiente correo electrónico: manuel.torresr@hotmail.com.



## Agradecimientos

Cada libro desarrollado es una labor en equipo que a veces lo asume una sola persona, sacrificando su tiempo con la familia, horas de sueño y muchas otras cosas más, que luego tienen su recompensa cuando el material preparado es por fin una realidad.

Me complace compartir con ustedes los casos expuestos, los cuales además facilitarán mi desempeño como docente; por ello, en primer lugar, este agradecimiento, es para la familia MACRO, por confiar nuevamente en mi persona para la realización del presente libro.

En segundo lugar, deseo expresar todo mi agradecimiento a las personas que he privado de un tiempo que siempre les perteneció, como son mis pequeñas hijas Ángela Victoria y Fernanda Ximena Torres Lázaro y mi esposa Luz. Ellas, con mucha paciencia, me han brindado el tiempo necesario para la elaboración de este material. Siempre les estaré agradecido por ese gesto que han tenido conmigo.

## Dedicatoria

Este libro está dedicado a mis pequeñas Ángela y Fernanda, que son y seguirán siendo mi fuente de inspiración y mucho más. Y a mi esposa Luz por comprenderme en todo lo que me propongo.

# Índice

Introducción.....	13
-------------------	----

## Capítulo 1

Introducción al HTML5.....	15
1.1 Definiciones básicas.....	17
1.1.1 <i>Software</i> .....	17
1.1.2 <i>Software</i> libre.....	18
1.1.3 Ventajas del <i>software</i> libre.....	19
1.1.4 Desventajas del <i>software</i> libre.....	19
1.1.5 <i>Software</i> propietario.....	20
1.1.6 <i>Freeware</i> .....	20
1.1.7 <i>Shareware</i> .....	21
1.1.8 GNU.....	21
1.1.9 Lenguaje interpretado.....	21
1.1.10 Lenguaje compilado.....	22
1.2 Introducción al HTML5.....	23
1.2.1 Concepto.....	23
1.2.2 Nuevos conceptos.....	24
1.2.3 Estructura de una etiqueta HTML5.....	25
1.2.4 Etiquetas obsoletas para HTML5.....	26
1.2.5 Etiquetas HTML5 que cambian su significado.....	26
1.2.6 Atributos de una etiqueta HTML5.....	27
1.2.7 Identificación de los atributos en una etiqueta HTML5.....	28
1.2.8 Especificación DOCTYPE.....	29
1.2.9 La etiqueta HEAD.....	29
1.2.10 La etiqueta BODY.....	30
1.2.11 ¿Qué elementos podemos colocar dentro del BODY?.....	30
1.2.12 Comentarios en HTML5.....	32
1.3 Funcionamiento de un servidor web.....	32
1.4 Introducción al Apache.....	32
1.4.1 Descargar servidor Apache.....	33
1.4.2 Instalación del servidor Apache.....	34
1.4.3 Pruebas del servidor Apache.....	37
1.5 Casos desarrollados de script HTML5 ejecutados desde el servidor Apache.....	37
• Caso desarrollado 1: Menú de opciones vertical simple.....	37
• Caso desarrollado 2: Menú de opciones horizontal.....	40
• Caso desarrollado 3: Menú de opciones vertical con resaltado desde el puntero del mouse.....	41
• Caso desarrollado 4: Sección con HTML5.....	43
• Caso desarrollado 5: Artículo con HTML5.....	44
• Caso desarrollado 6: Pie de página con HTML5.....	45
• Caso desarrollado 7: Compra de productos con tablas.....	46
• Caso desarrollado 8: Formulario de registro de usuarios.....	48
• Caso desarrollado 9: Carga de archivos.....	52

## Capítulo 2

<b>Introducción al PHP</b> .....	55
2.1 Definición de PHP.....	57
2.2 Usos de PHP.....	58
2.3 Evolución de PHP.....	60
2.4 Novedades de la última versión de PHP.....	60
2.5 Introducción al WAMP Server.....	61
2.5.1 Descargar WAMP Server.....	62
2.5.2 Instalación del servidor WAMP Server.....	63
2.5.3 Pruebas del servidor WAMP.....	66
2.5.4 Anomalías en la prueba del servidor WAMP.....	66
2.6 Instalación de Netbeans para PHP.....	68
2.6.1 Paquete de aplicaciones JDK.....	68
2.6.2 IDE Netbeans.....	70
2.7 Cuestiones posteriores a la instalación del Netbeans.....	73
• Cuestión 1: Crear un proyecto en Netbeans.....	73
• Cuestión 2: Agregar un archivo HTML5 al proyecto.....	75
• Cuestión 3: Agregar la paleta con etiquetas HTML5.....	76
• Cuestión 4: Agregar y ejecutar un archivo PHP.....	76
• Cuestión 5: Agregar y ejecutar un archivo de página web PHP.....	77
• Cuestión 6: Agregar un archivo CSS al proyecto.....	78
• Cuestión 7: Modificar la fuente y tamaño del código mostrado en el editor de Netbeans.....	79
• Cuestión 8: Modificar las ubicaciones de los proyectos al ejecutarlos.....	80
• Cuestión 9: Modificar el navegador predeterminado.....	81

## Capítulo 3

<b>Lenguaje PHP</b> .....	83
3.1 Integrar código PHP en HTML5.....	85
3.2 Salida de información con PHP.....	86
3.2.1 Función echo.....	86
3.2.2 Función printf.....	87
3.2.3 Comentarios PHP.....	88
3.3 Página estática versus página dinámica.....	88
3.4 Manejo de literales de programación.....	91
3.5 Manejo de operadores.....	91
3.5.1 Operadores aritméticos.....	92
3.5.2 Operadores de cadena de caracteres.....	92
3.5.3 Orden de prioridad de los operadores.....	92
3.6 Manejo de variables.....	93
3.7 Tipos de datos usados en PHP.....	94
3.8 Manejo de constantes.....	96
3.9 Casos desarrollados.....	96
• Caso desarrollado 1: Diferencia entre echo y printf.....	96

• Caso desarrollado 2: Manejo de variables y operadores.....	99
• Caso desarrollado 3: Manejo de constantes.....	102

## Capítulo 4

<b>Estructuras condicionales .....</b>	<b>105</b>
4.1 Definición de lógica booleana .....	107
4.2 Estructurar una condición lógica en PHP.....	109
4.2.1 Operadores de comparación.....	110
4.2.2 Operadores lógicos .....	110
4.2.3 Estructurar bloques de código .....	110
4.2.4 Control de errores.....	111
4.3 Estructura condicional If simple .....	112
4.4 Estructura condicional If doble.....	113
4.5 Estructura condicional If doblemente enlazada .....	115
4.6 Estructura condición switch .....	117
4.7 Casos desarrollados.....	121
• Caso desarrollado 1: Salario de empleados usando condicional simple.....	121
• Caso desarrollado 2: Obsequio a clientes usando condicional simple.....	126
• Caso desarrollado 3: Venta de productos usando condicional doble.....	132
• Caso desarrollado 4: Control de mensualidad usando condicional doblemente enlazada.....	137
• Caso desarrollado 5: Venta de entradas usando condicional múltiple con switch.....	144

## Capítulo 5

<b>Estructuras repetitivas .....</b>	<b>149</b>
5.1 Operadores de conteos y acumulaciones.....	151
5.1.1 Operadores de incremento y decremento.....	151
5.1.2 Operadores complejos.....	151
5.2 Contadores .....	152
5.3 Acumuladores .....	153
5.4 Estructura while.....	154
5.4.1 Ciclo de repeticiones while con cero iteración .....	156
5.4.2 Ciclo de repeticiones while infinito.....	156
5.4.3 Uso de la instrucción break en la estructura while.....	157
5.4.4 Uso de la instrucción continue en la estructura while.....	157
5.4.5 Anidamiento de ciclos while .....	158
5.5 Estructura for .....	158
5.5.1 Analogías entre while y for .....	161
5.5.2 Uso de la instrucción break en la estructura for .....	161
5.5.3 Uso de la instrucción continue en la estructura for .....	162
5.5.4 Anidamiento de ciclos for .....	162
5.6 Estructura do... while.....	163
5.6.1 Analogías entre while, for y do...while.....	164
5.7 Casos desarrollados.....	164
• Caso desarrollado 1: Venta de productos usando while.....	165
• Caso desarrollado 2: Pago de préstamo usando for.....	171

## Capítulo 6

<b>Funciones</b> .....	177
6.1 Funciones para variables .....	179
6.1.1 Función <code>isset</code> .....	179
6.1.2 Función <code>unset</code> .....	181
6.1.3 Función <code>gettype</code> .....	182
6.1.4 Función <code>settype</code> .....	183
6.1.5 Función <code>empty</code> .....	184
6.1.6 Función <code>is_integer</code> .....	186
6.1.7 Función <code>is_double</code> .....	188
6.1.8 Función <code>is_string</code> .....	190
6.1.9 Función <code>var_dump</code> .....	192
6.2 Funciones de cadena .....	192
6.2.1 Función <code>strlen</code> .....	193
6.2.2 Función <code>strpos</code> .....	195
6.2.3 Función <code>strcmp</code> .....	196
6.2.4 Función <code>strstr</code> .....	196
6.2.5 Función <code>substr</code> .....	197
6.2.6 Funciones <code>ltrim</code> , <code>rtrim</code> , <code>chop</code> y <code>trim</code> .....	198
6.2.7 Función <code>str_replace</code> .....	198
6.2.8 Funciones <code>strtolower</code> y <code>strtoupper</code> .....	199
6.2.9 Función <code>preg_match</code> .....	199
6.2.10 Función <code>explode</code> .....	203
6.2.11 Función <code>strrev</code> .....	205
6.2.12 Función <code>str_repeat</code> .....	205
6.2.13 Función <code>str_pad</code> .....	205
6.3 Funciones numéricas.....	206
6.3.1 Función <code>abs</code> .....	206
6.3.2 Función <code>ceil</code> .....	207
6.3.3 Función <code>exp</code> .....	207
6.3.4 Función <code>floor</code> .....	207
6.3.5 Función <code>getrandmax</code> .....	208
6.3.6 Función <code>max</code> .....	208
6.3.7 Función <code>min</code> .....	208
6.3.8 Función <code>mt_rand</code> .....	209
6.3.9 Función <code>pi</code> .....	209
6.3.10 Función <code>pow</code> .....	209
6.3.11 Función <code>round</code> .....	210
6.3.12 Función <code>sqrt</code> .....	210
6.4 Funciones de fecha y hora .....	211
6.4.1 Función <code>date</code> .....	211
6.4.2 Función <code>time</code> .....	212
6.4.3 Función <code>checkdate</code> .....	213
6.4.4 Función <code>getdate</code> .....	214
6.5 Funciones implementadas por el usuario .....	215
6.5.1 Definición y usos.....	215
6.5.2 Implementación de una función .....	216

6.5.3	Llamando a una función.....	218
6.5.4	Implementación de una función con parámetros .....	218
6.5.5	Implementación de una función con parámetros y con valor por defecto .....	219
6.5.6	Implementación de una función sin valor de retorno.....	220
6.5.7	Implementación de una función con múltiples valores de retorno .....	220
6.5.8	Implementación de funciones anónimas (lambda en PHP) .....	221
6.6	Funciones include y require .....	223
6.6.1	Función include.....	223
6.6.2	Función require.....	228
6.7	Casos desarrollados .....	232
	• Caso desarrollado 1: Funciones de cadena - Registro de empleado.....	232
	• Caso desarrollado 2: Funciones numéricas - Promedio de notas.....	240
	• Caso desarrollado 3: Funciones implementadas por el usuario - Venta de productos....	246
	• Caso desarrollado 4: Funciones anónimas - Pago de estudiantes.....	255

## Capítulo 7

Arreglos .....	261	
7.1	Introducción.....	263
7.1.1	Tipos de arreglos.....	264
7.2	Estructura repetitiva foreach.....	266
7.3	Administrar elementos de un arreglo.....	269
7.3.1	Insertar elementos.....	269
7.3.2	Insertar elementos numéricos mediante una función.....	271
7.3.3	Recorrer los elementos por índice. ....	272
7.3.4	Recorrer por elementos asociativos .....	274
7.3.5	Modificar elementos.....	275
7.3.6	Extrayendo un rango de elementos con array_slice .....	277
7.3.7	Avanzar y retroceder por elementos .....	277
7.3.8	Eliminar elementos.....	282
7.4	Métodos de un arreglo.....	285
7.4.1	Ordenamiento de elementos .....	285
7.4.2	Convertir un arreglo en una lista de variables .....	289
7.4.3	Convertir cadena de caracteres en array .....	293
7.4.4	Eliminando elementos repetidos en un array .....	294
7.4.5	Invertir los elementos de un arreglo.....	295
7.5	Arreglos multidimensionales.....	296
7.6	Casos desarrollados.....	299
	• Caso desarrollado 1: Arreglo indexado - Informe de notas.....	299
	• Caso desarrollado 2: Arreglo asociativo - Informe de notas.....	306
	• Caso desarrollado 3: Arreglo indexado - Manejo de imágenes.....	312
	• Caso desarrollado 4: Arreglo indexado - Paginación de productos.....	315
	• Caso desarrollado 5: Arreglo indexado - Paginación de imágenes.....	318
	• Caso desarrollado 6: Manejo de la función include - Listado de productos.....	322
	• Caso desarrollado 7: Manejo de la función require - Control de pago.....	326
	• Caso desarrollado 8: Manejo de la función require - Control de facturas.....	329

## Capítulo 8

<b>Archivos</b> .....	335
8.1 Manejo de archivos .....	337
8.1.1 Función file_exist .....	337
8.1.2 Función fopen. ....	338
8.1.3 Función fclose .....	339
8.1.4 Función fwrite .....	340
8.1.5 Función fread .....	340
8.1.6 Función fgets .....	341
8.1.7 Función fputs. ....	342
8.1.8 Función rewind .....	343
8.1.9 Función filectime .....	343
8.1.10 Función file .....	344
8.1.11 Función file_put_contents .....	345
8.2 Manejo de archivos y carpetas .....	346
8.2.1 Función scandir .....	346
8.2.2 Función unlink .....	347
8.2.3 Función rename .....	347
8.3 Casos desarrollados .....	348
• Caso desarrollado 1: Verificar la existencia de un archivo .....	348
• Caso desarrollado 2: Contador de visita básico .....	352
• Caso desarrollado 3: Contador de visitas de forma gráfica .....	354
• Caso desarrollado 4: Control de registro de clientes .....	357

## Capítulo 9

<b>Sesiones</b> .....	365
9.1 Introducción .....	367
9.2 Definición de sesiones .....	368
9.3 Funciones de session .....	368
9.3.1 Función session_start() .....	368
9.3.2 Función session_id() .....	369
9.3.3 Función session_name() .....	370
9.3.4 Función session_unset() .....	370
9.3.5 Función session_destroy() .....	370
9.4 Escritura y lectura de una variable de session .....	371
9.5 Escritura y lectura de un arreglo unidimensional en la session .....	372
9.6 Escritura y lectura de un arreglo asociativo en la session .....	373
9.7 Casos desarrollados .....	374
• Caso desarrollado 1: Verificación de la session .....	375
• Caso desarrollado 2: Uso de colores desde la session .....	376
• Caso desarrollado 3: Encuesta de inseguridad .....	379
• Caso desarrollado 4: Login de usuario .....	392
• Caso desarrollado 5: Votación de candidatas .....	399
• Caso desarrollado 6: Carrito de compras básico - Venta de productos .....	405
• Caso desarrollado 7: Registro de nuevos productos .....	416

<b>Bibliografía</b> .....	423
---------------------------	-----

## Introducción

PHP es uno de los lenguajes de programación web más usados en la actualidad, que combina su código con HTML5, como Visual o Java en sus aplicaciones web, implementando aplicaciones dinámicas de manera profesional. Entre sus características más relevantes, podemos afirmar que se trata de un lenguaje multiplataforma que puede ser ejecutado en cualquier tipo de dispositivo con suficiente capacidad para conectarse a cualquier base de datos; además, posee una buena fuente de documentación en su sitio web oficial. Asimismo, es considerado como un *software* libre, ya que puede ser usado en cualquier ámbito. Muchas compañías han optado por el uso de PHP por su versatilidad y su fácil uso, como Yahoo INC, Wikipedia.org, Facebook.com, Sourceforge.org, Flickr.com, Joomla Wordpress, Drupal, Moodle, etc.

El libro está dividido en nueve capítulos. En el capítulo uno se hace un reconocimiento de las etiquetas HTML5, ya que cuando PHP implementa sus aplicaciones se integra al HTML, y este a su vez al CSS3. Cabe mencionar que a partir del capítulo tres se presentan casos desarrollados que incluyen ambas tecnologías. En el capítulo dos se expone una introducción al lenguaje de programación PHP y se explica la instalación de las aplicaciones necesarias para poder desarrollar otras con PHP desde un servidor Apache. En el capítulo tres se analiza el lenguaje PHP, como la declaración de variables, constantes y, principalmente, la integración entre PHP y HTML5. Asimismo, se presentan casos desarrollados paso a paso. En el capítulo cuatro se reconocen todas las estructuras condicionales que maneja PHP, como el if simple, doble, doblemente enlazado; y condicionales múltiples, como el manejo del switch. En el capítulo cinco se utilizan las estructuras repetitivas de PHP, las cuales permiten crear aplicaciones de manera dinámica, proponiendo casos desarrollados que usen for y while. En el capítulo seis se hace referencia a las funciones en PHP, pues resulta importante implementar aplicaciones con programación modular usando las principales funciones que presenta PHP, como funciones de cadenas, fechas, etc. En el capítulo siete se trata el tema de los arreglos y la diferencia que presentan estos con los lenguajes tradicionales, integrando a la vez el tema de funciones. En el capítulo ocho se desarrolla el tema de archivos para implementar aplicaciones que manejen información permanente, para lo cual se manejará nuevas funciones en el manejo de los archivos desde PHP. Finalmente, en el capítulo nueve se desarrolla el tema de sesiones, que permitirá crear aplicaciones web que sean capaces de fusionar las funciones y arreglos en una misma aplicación.

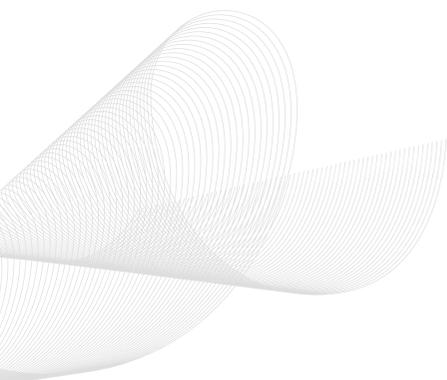




**CAP.**

**1**

# ***Introducción al HTML5***





PHP es considerado como un lenguaje de programación de código abierto, es muy usado por los desarrolladores de páginas web, también llamados *webmaster*. Así como PHP, existen muchos lenguajes que permiten implementar aplicaciones web, como Visual Studio con su ASP, o Java con sus Java Servlets y JavaServer Pages (JSP). Desde aquí se desprende el término código abierto, sobre el cual también se expondrá en este capítulo, ya que resulta importante diferenciar los tipos de *software* que hay en el mercado, y cuál de ellos implementa PHP. El común denominador entre todos los lenguajes mencionados es la integración de su lenguaje con las etiquetas HTML; en nuestro caso, lo realizaremos usando la tecnología propuesta por HTML5, para ello mencionaremos en este libro las principales etiquetas y su trabajo embebido con PHP.

Una de las ventajas del desarrollo de aplicaciones con PHP es que resulta muy sencillo de entender su sintaxis, pues solo se requiere tener alguna experiencia en cualquier lenguaje de programación y un tiempo para leer la sintaxis propia de PHP; y todo ello se puede encontrar en Internet, tanto en español (<http://www.php.net/manual/es/>) como en inglés (<http://www.php.net/manual/en/>).

También debemos considerar la importancia que tiene el lenguaje HTML5 en la elaboración de aplicaciones web con PHP, pues resulta de vital importancia su integración, por tal motivo hemos preparado en este capítulo un resumen de las principales etiquetas HTML5 con casos desarrollados, lo cual nos servirá para realizar aplicaciones web con PHP de forma profesional.

Hoy en día se ha vuelto muy importante la interacción entre el usuario y las aplicaciones web; prácticamente se ha vuelto una necesidad el uso de las aplicaciones, pero cada vez con menos asistencia o capacitación; es decir, el usuario deberá usar su sentido común para poder interactuar de la mejor manera posible frente a las aplicaciones web, ya que estas no solo están en las computadoras personales sino también en dispositivos móviles, como celulares o tabletas; o más cerca aun, con los dispositivos de uso doméstico, como son los aparatos electrónicos, por ejemplo, la televisión *smart* o el juego de consola PlayStation. Es así como PHP se convierte en una herramienta importante para la implementación de aplicaciones web.

## 1.1 DEFINICIONES BÁSICAS

Antes de comenzar con el desarrollo de aplicaciones con PHP debemos tener en cuenta algunos conceptos básicos que serán de vital importancia para nuestro desempeño en el uso del lenguaje PHP.

### 1.1.1 Software

El concepto varía dependiendo del entorno donde nos encontremos, pero veamos la definición expuesta en IEEE Software Engineering Standard: Glossary of Software Engineering Terminology. IEEE Computer Society Press, 1993: «Es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación».

O también podríamos decir que el *software* es un: «Conjunto de programas, instrucciones y reglas informáticas que permiten ejecutar distintas tareas en una computadora», según la Real Academia Española (RAE).

Finalmente, podríamos decir que el concepto de *software* abarca a todas las aplicaciones informáticas que tienen un objetivo claro y específico, como lo es PHP, que es considerado como *software* libre. Este concepto lo veremos en el siguiente tema. Veamos algunas imágenes de los programas más conocidos:



Fuente: <<http://windows.microsoft.com/>>



Fuente: <<http://www.linux.org/>>



Fuente: <<http://www.apache.org/>>



Fuente: <<http://www.php.net/>>

### 1.1.2 Software libre

Hoy en día es muy usado el término *software* libre y su concepto resulta algo predecible, pues el término «libre» indica que se le otorga a los usuarios toda la libertad sobre el *software*. Se podría decir que el *software* se puede adecuar a una determinada situación; si no cumple con esta premisa, entonces no es considerado *software* libre. Así, podríamos tomar el siguiente concepto de *software* libre como válido: «Es el *software* que respeta la libertad de los usuarios y la comunidad. En grandes líneas, significa que los usuarios tienen la libertad para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el *software*»<sup>1</sup>.

Eso quiere decir que para considerar a un *software* realmente libre, este deberá venir con su código fuente y que los usuarios podrán modificarlos de acuerdo a lo que convengan. Hay que tener en cuenta que existe una gran diferencia entre *software* libre y *software* gratuito. Este último hace referencia a un costo cero por el *software*; pero no se considera como *software* libre, pues el término «libertad» no necesariamente es sinónimo del término «precio». Veamos algunos ejemplos de *software* libre:

Navegadores web:

- **Mozilla Firefox:** Navegador web considerado como un fuerte competidor de Internet Explorer. Una de las características principales es la rapidez al mostrar información, su entorno bastante amigable y su costo totalmente gratuito.
- **Google Chrome:** Navegador web de rápido servicio y cada vez más popular entre usuarios de todo el mundo. No solo trabaja en computadoras personales, ahora está disponible en dispositivos móviles, como celulares, tabletas y televisores inteligentes, y su distribución es gratuita.
- **Opera:** Navegador web considerado como el más completo y rápido. Cuenta con dos versiones, una propietaria y otra gratuita.

Clientes para protocolos de transferencia de archivos (FTP):

- **FileZilla:** Permite hacer transferencia de archivos desde una máquina local a un servidor web, de tal manera que podríamos subir nuestros documentos web directamente al servidor, sin necesidad de hacer un envío uno a uno de los archivos.

<sup>1</sup> Concepto tomado de: <https://www.gnu.org/philosophy/free-sw.es.html>.

Editores de páginas web:

- **Amaya:** Es el editor oficial del W3C llamado también consorcio internacional de la web. Este permite implementar documentos web como si estuviera trabajando en Dreamweaver.
- **Nvu:** Es un editor de documentos web de forma visual, muy poco usado en nuestro medio, pero su característica principal es que en su mayor parte está basado en el motor de Mozilla.

### 1.1.3 Ventajas del software libre

Veamos algunas ventajas que puede presentar un *software* definido como libre:

- No permitirá la adquisición de licencias ni las generará, ya que de lo contrario no sería considerado como *software* libre.
- Puede representar un beneficio tecnológico para la sociedad, ya que es de uso libre e ilimitado.
- Evita la distribución con beneficio, ya que siempre será considerado como *software* libre en todos sus aspectos.
- Cumple con todos los estándares establecidos para un *software* y se encuentra actualizado, ya que existen muchos colaboradores en el mundo que se encuentran dispuestos a colaborar con el proyecto.
- Se caracteriza por ser diverso y no centrarse en un tema particular.

Muchos piensan que el *software* libre se caracteriza por ser inseguro, inestable y con poca tendencia a crecer, algo así como un temor de llevar todos los procesos que vienen funcionando correctamente a *software* libre. Pero hoy en el Perú está ocurriendo un crecimiento en el uso de este tipo de *software*, que es impulsado generalmente por el estado peruano, tal es así que los organismos públicos ya han comenzado a usarlo; dentro de los cuales se encuentran: la Superintendencia Nacional de Aduanas y de Administración Tributaria-Sunat, Ministerio de Educación, Poder Judicial, Ministerio del Trabajo, Ministerio Público, entre otros que ya operan sus procesos de negocios mediante el uso de *software* libre.

### 1.1.4 Desventajas del software libre

Veamos algunas desventajas que presenta el uso de un *software* que es considerado como libre:

- Se tiene que contar con una persona capacitada en el tema de *software* libre y con conocimientos de programación, ya sea PHP, JAVA, PERL o PYTHON.
- El *software* libre no ofrece ningún tipo de garantía sobre el uso o administración del mismo, ya que no cuenta con una empresa que respalda el uso de dicho *software*.
- La interfaz gráfica del usuario (GUI) recién está tomando un aspecto atractivo para el usuario final, asumimos que seguirá mejorando en los años siguientes.

### 1.1.5 Software propietario

También es conocido como *software* no libre, *software* privativo, *software* privado, *software* con propietario o simplemente *software* de propiedad. Una de las características principales del *software* propietario es que no se tiene el código fuente del mismo; por lo tanto, no se puede adaptar a las necesidades de una determinada organización. Un *software* propietario pertenece a una determinada empresa, es ella la que mantiene los derechos de uso, alteración o distribución. Es aquí donde se desprende el término «licencia», el cual garantiza una actualización, capacitación e incluso un soporte técnico en beneficio del usuario.



Fuente: <<http://www.omicrono.com/2012/10/microsoft-office-2013-llegara-a-ios-y-android-en-marzo/>>



Fuente: <<http://www.unocero.com/2013/09/13/sale-visual-studio-2013-release-candidate/>>

### 1.1.6 Freeware

Viene del inglés y significa «*software* gratis», el cual presenta características como la distribución en forma gratuita, ya sea por medio de copias o descargas desde algún servidor. Es totalmente libre de uso; es decir, no se necesita licencia alguna y finalmente el tiempo de uso del freeware es limitado. En otros términos, el usuario decidirá hasta cuándo debe o no usar dicho *software*.

Podríamos nombrar algunos ejemplos de freeware, como el reproductor multimedia Windows Media Player, el reproductor iTunes, el navegador Internet Explorer, el navegador Opera Browser, Ares, Atube catcher, etc.



Fuente: <<http://diymediahome.org/outputting-wmp-videos-to-second-display/?lang=es>>



Fuente: <http://www.izalmusic.com/tienda>

### 1.1.7 Shareware

Significa «compartir por partes», en la cual un determinado *software* puede ser distribuido entre los usuarios de forma gratuita, estos a su vez evaluarán dicho producto en un tiempo determinado; claro está que dicho *software* contará con las limitaciones puestas por el propietario.

Muchos propietarios de *software* han tomado este término como la capacidad de distribuir *software* de prueba, en lo cual se denota las limitaciones del *software* hasta que el usuario decida comprar una licencia del producto. Podemos observar *software* de distribución shareware en los antivirus que nos ofrecen usar su producto por un número determinado de días, al cabo de estos se solicita la adquisición del *software* a través de mensajes enviados desde el mismo *software* instalado.

### 1.1.8 GNU

GNU (Unix-like) es un sistema operativo implementado por el proyecto GNU, que fue iniciado por Richard Stallman, un programador estadounidense que es conocido por implantar un marco legal para el movimiento del *software* libre en el mundo. Gracias a él, hoy en día podemos hacer uso del *software* libre sin problemas de licencias, pero con ciertas limitaciones.

GNU está formado en su totalidad por *software* libre, pero la diferencia con el sistema operativo Unix es la libertad de su uso, y su carencia de código de Unix.

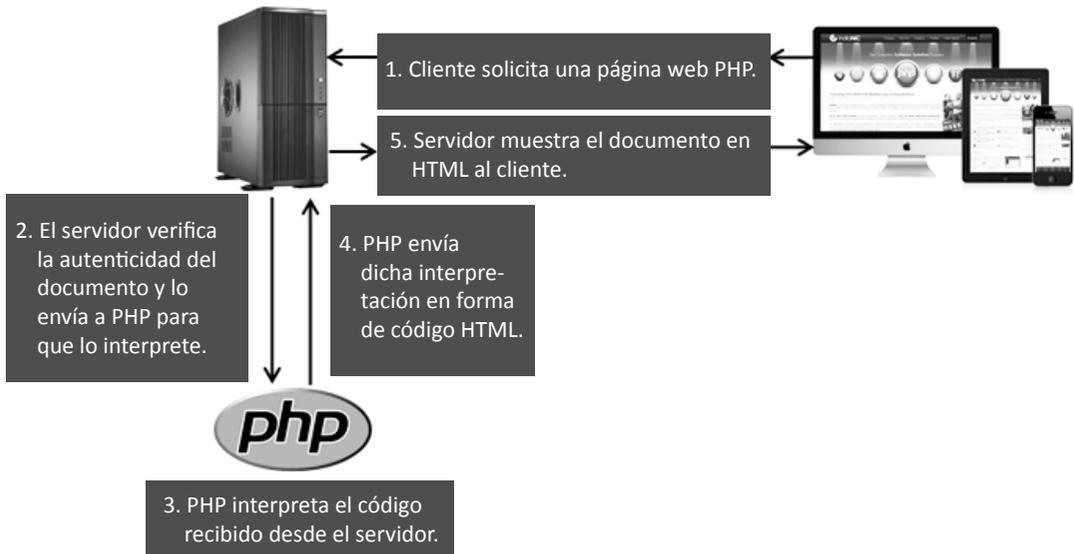
El sistema GNU se lanzó bajo una licencia denominada «copyleft», que representa sin autoría; por lo tanto, puede ser distribuido de manera que cualquier usuario es capaz ejecutar, copiar, modificar o distribuir el sistema sin ningún problema. Dicha licencia está contenida en la Licencia General Pública de GNU, más conocida como GPL.

Richard Stallman además creó Free Software Foundation –también llamado «Fundación para el Software Libre» o simplemente FSF– para proveer soporte logístico, legal y financiero al proyecto GNU que tanto esperaba el mundo.

Es así que años más tarde, en 1991, un ingeniero de *software* llamado Linus Torvalds comenzó a implementar un núcleo que lo llamó Linux, y lo distribuyó bajo licencia GPL. Finalmente en 1992 es combinado con el sistema GNU, lo cual tuvo resultados nunca antes esperados; es así que se rompe el monopolio Microsoft y nace un pequeño monstruo llamado GNU/Linux, o simplemente el sistema operativo Linux.

### 1.1.9 Lenguaje interpretado

Se conoce como lenguaje interpretado cuando el código desarrollado es traducido por un intérprete a un lenguaje que puede ser entendido por la máquina, este proceso de interpretación se repetirá cada vez que se ejecute una determinada aplicación. Este lenguaje ha tenido un alto crecimiento en el desarrollo de aplicaciones web, como PHP, Ruby, Phyton y otros lenguajes interpretados.



En la imagen observamos cómo se ejecuta una petición desde el cliente hacia el intérprete de PHP.

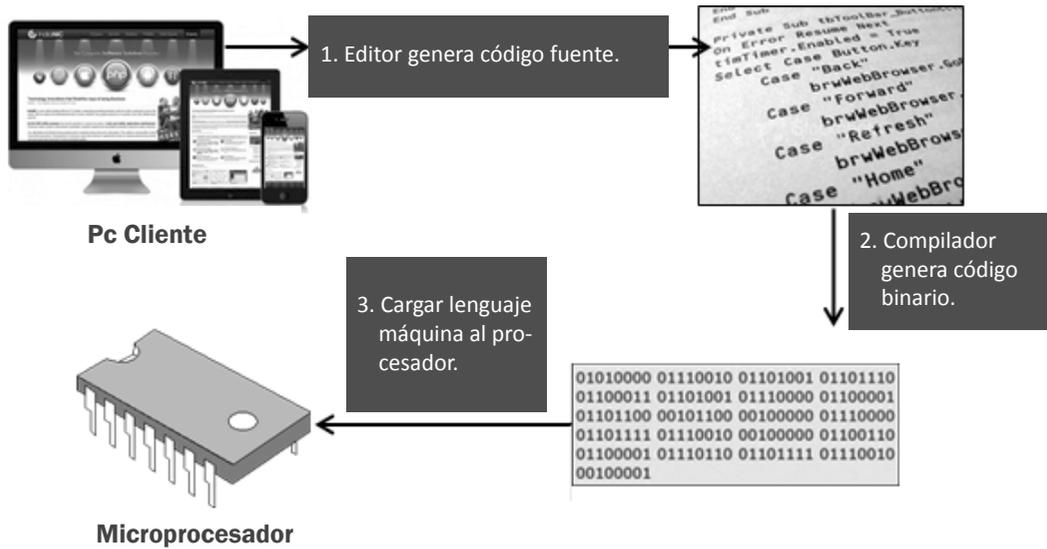
Veamos algunas características básicas:

- Una variable puede almacenar un valor y comportarse como el tipo de datos que el valor le provee, es por eso que no es necesario realizar declaraciones de variables, ya que son declaradas como tipo de datos dinámicos.
- Posee independencia con respecto a la plataforma de trabajo; es decir, no tiene un *software* asociado al lenguaje; esto permite implementar dicho lenguaje en cualquier editor, como es el caso de PHP, el cual se puede codificar en Bloc de notas o Netbeans.
- La independencia de plataforma conviene al momento de comparar el tamaño de las aplicaciones, es por eso que los lenguajes interpretados mayormente son usados por aplicaciones web.
- Mayor consumo de tiempo porque necesita ser interpretado, en eso radica la diferencia con los lenguajes compilados que veremos más adelante.

### 1.1.10 Lenguaje compilado

Un lenguaje compilado hace que el código sea traducido por un compilador a un archivo ejecutable, entendible por una máquina desde cualquier plataforma de trabajo. Una de las grandes ventajas de este lenguaje compilado es que consume menos tiempo al ejecutar una aplicación, pues como el archivo ya se encuentra compilado ya no se ejecuta el proceso de interpretación a cada momento.

Entre los lenguajes de programación considerados como compilados, tenemos al lenguaje C con sus variaciones C++, Objective C, C#; y, por otro lado, Visual Basic. Este último es muy usado por programadores por ser un lenguaje compilado y muy sencillo de programar.



En la imagen podemos observar cómo generar un código binario desde el compilador, que luego será cargado en el procesador de la computadora, para una mayor agilidad en el proceso de ejecución.

## 1.2 INTRODUCCIÓN AL HTML5



Fuente: <<http://www.w3.org/html/logo/>>



Fuente: <<http://www.html5xcss3.com/2013/06/zcorporate-responsive-html5-theme.html>>

### 1.2.1 Concepto

HTML5 es un lenguaje de etiquetas que permite diseñar documentos web estáticos; a diferencia de las versiones anteriores al HTML5, esta ofrece un conjunto de funciones que permitirá dar una nueva experiencia en el diseño web, veamos algunas de estas funciones:

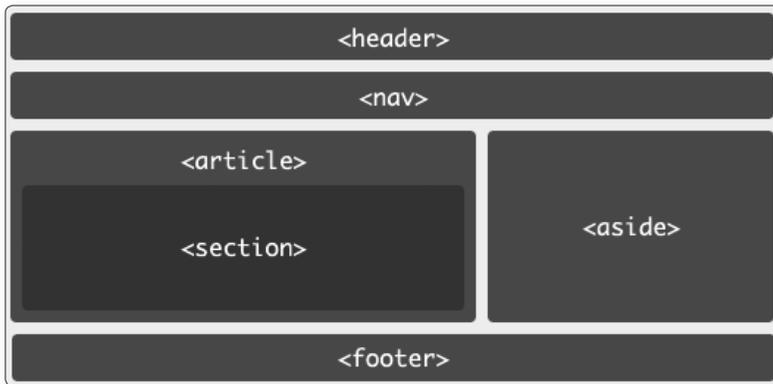
- **Semántica:** Mejora en la distribución de los elementos web, lo que da precisión al contenido.
- **Conectividad:** Permite comunicarse con un servidor web de manera ágil, sin consumir muchas capacidades.
- **Servicio local:** Permite navegar por un sitio web sin necesidad de estar en línea, también es llamado ejecución local desde el lado cliente.
- **Multimedia:** Permite asignar archivos de videos o música con una sola etiqueta de manera sencilla.

- **Alto rendimiento:** Proporciona una mayor optimización con respecto a la velocidad usada en el *hardware* del equipo cliente.
- **Multiplataforma:** Permite visualizar las páginas web en diferentes dispositivos con la misma calidad.

### 1.2.2 Nuevos conceptos

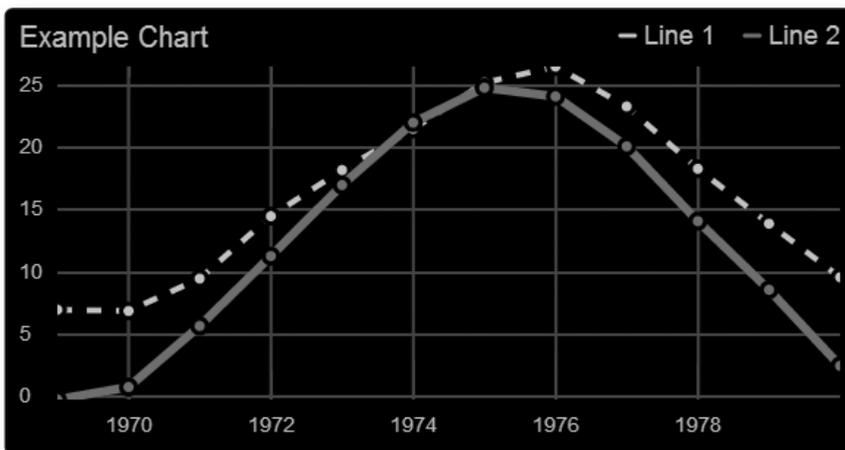
- **Con respecto a su estructura:**

HTML5 presenta nuevas etiquetas que mejorarán la experiencia en el desarrollo de documentos web; eso quiere decir que las etiquetas anteriores al HTML5 han sido reformuladas en algunos casos, y otros se mantienen tal como se iniciaron. Esta nueva versión presenta un nuevo orden en los elementos; basado en seis etiquetas como su columna vertebral: header, nav, article, section, aside y footer, logrando a su vez un aumento en el dinamismo de la página web.



- **Gráficos canvas:**

Canvas ya estaba implementado en las versiones anteriores al HTML5, pero esta vez mejora en el uso y propone nuevas funciones, siempre teniendo como base el script de Javascript.



Fuente: <http://www.html5canvastutorials.com/meteorcharts/html5-canvas-meteorcharts-series-styling/>

- **Audio y video:**

HTML5 incorpora un reproductor multimedia para la ejecución de videos y música en línea; a diferencia de las versiones anteriores al HTML5, esta funcionalidad presenta menos código para la implementación de audio y video en la web, haciéndolo sencillo y amigable. Además, no necesita la instalación de ningún *plugin* para su uso.



Fuente: <<http://pixelcoblog.com/6-recursos-para-video-html5/>>

- **Geolocalización:**

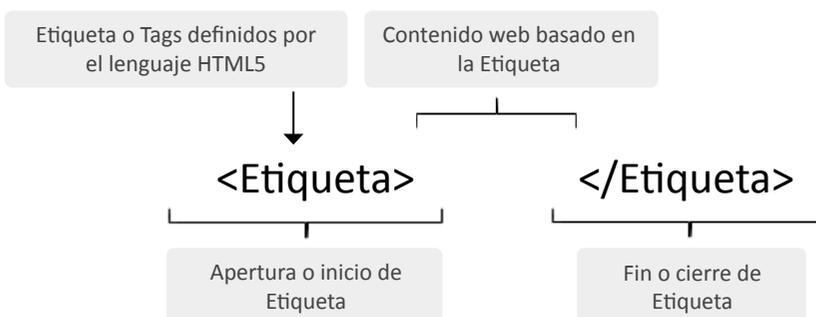
La geolocalización permitirá determinar desde qué lugar se está visualizando un sitio web, algo que ya se viene manejando en la red social Facebook. Y todo eso gracias a los sistemas de referenciación, como el GPS. Esto permitirá al usuario tener una rápida ubicación de la empresa que promociona la web, lo que dará confianza al cliente al momento de realizar una compra en línea.



Fuente: <<http://www.cristalab.com/blog/como-funciona-la-geolocalizacion-por-wifi-c1076771/>>

### 1.2.3 Estructura de una etiqueta HTML5

La estructura de las etiquetas HTML5 no ha sufrido cambios con respecto a las versiones anteriores, su formato es:



Características de las etiquetas:

- Toda etiqueta en HTML5 se encuentra estandarizada por la W3C, que se considera como una comunidad internacional que desarrolla estándares que aseguran el crecimiento de la web.
- No es considerado *case sensitive*; es decir, no distingue la mayúscula de la minúscula, por lo tanto no genera error cuando se escribe de una u otra forma.
- Casi siempre se usan dos etiquetas; la primera indicará el inicio de la etiqueta, mientras que la otra especificará el final de la misma.
- Las etiquetas obsoletas provenientes de las versiones anteriores no han sido deshabilitadas por los navegadores web actuales; es así que podemos seguir usándolas, pero recuerde que HTML5 promueve tanto el uso de nuevas etiquetas como el de estilos CSS3.

#### 1.2.4 Etiquetas obsoletas para HTML5

Veamos una lista de las principales etiquetas que ya no serán usadas por HTML5:

ETIQUETA OBSOLETA	DESCRIPCIÓN	ETIQUETA EN HTML5
<b>applet</b>	Permite insertar un script realizado por algún lenguaje de programación, como por ejemplo Java e integrarlo a html.	Object
<b>basefont</b>	Permite determinar el tamaño de fuente predeterminado.	Uso de CSS3
<b>big</b>	Permite mostrar un texto en tamaño grande.	Uso de CSS3
<b>center</b>	Permite centrar un elemento específico.	Uso de CSS3
<b>dir</b>	Permite insertar una lista de directorios.	UL
<b>font</b>	Permite establecer el estilo de fuente de un texto.	Uso de CSS3
<b>frame</b>	Permite insertar un marco web.	-
<b>frameset</b>	Permite insertar un grupo de frames o marcos web.	-
<b>strike</b>	Permite mostrar el texto que contiene estilo tachado.	Uso de CSS3
<b>tt</b>	Permite mostrar el texto como 'teletype' o espaciado simple.	Uso de CSS3
<b>u</b>	Permite mostrar el texto como subrayado.	Uso de CSS3
<b>xmp</b>	Permite definir un texto preformateado.	Pre

El uso de estilos CSS3 ha hecho que muchas etiquetas HTML ya no sean aplicables; es decir, perdieron su sentido de uso. El CSS3 presenta de manera profesional el mismo aspecto con ciertas ventajas; por ello, usted no tendrá problemas en adaptarse al cambio.

#### 1.2.5 Etiquetas HTML5 que cambian su significado

En HTML5 se han reacomodado la funcionalidad de unas pocas etiquetas, la mayoría de ellas trabaja como en las versiones anteriores. Veamos la lista de las etiquetas que modifican su funcionalidad:

- **Etiqueta small:** En HTML5, la etiqueta **small** se usa para hacer referencia a condiciones legales, informaciones de *copyright*, etc.

```
<small>
  Todos los derechos reservados-Copyright @2015.
</small>
```

- **Etiqueta b:** En HTML5 la etiqueta **b** se empleará cuando deseemos resaltar algún aspecto de un párrafo. La norma indica que la etiqueta **b** debe usarse como último recurso, y es mejor emplear encabezados (h1... h6), em, strong, mark o estilos CSS3 en función del tipo de información que se desea resaltar.

```
<p>
  Solo rendirán examen aquellos alumnos que hayan entregado la boleta de pago en
  <b>Counter de Secretaría Académica</b> en las fechas indicadas.
</p>
```

- **Etiqueta i:** En HTML5 la etiqueta **i** permite aplicar el estilo cursiva a un texto. Se recomienda su uso para resaltar nombres propios, una frase en otro idioma, una idea o un pensamiento, en otros casos usar estilos CSS3.

```
<p>
  Solo rendirán examen aquellos alumnos que hayan entregado la boleta de pago en
  <i>Counter de Secretaría Académica</i> en las fechas indicadas.
</p>
```

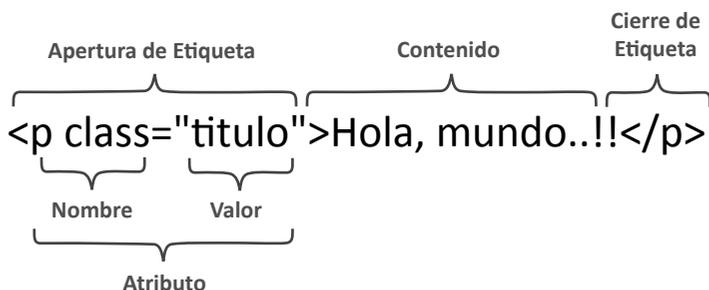
- **Etiqueta s:** En HTML5 la etiqueta **s** aplica un estilo tachado a un determinado texto el cual podría indicar que un párrafo no es recomendado o ya quedó obsoleto.

```
<p>Visítenos</p>
<p><s>dirección anterior: Av. Lima 549</s></p>
<p><strong>Nueva dirección: Jr. Cuzco 120</strong></p>
```

### 1.2.6 Atributos de una etiqueta HTML5

Se le denomina «atributos» a todas las opciones adicionales que puede tener una etiqueta; por ejemplo asignar un estilo o definir un nombre a la etiqueta, etc.

Componente de un atributo:



### 1.2.7 Identificación de los atributos en una etiqueta HTML5

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

#### ◦ Atributo id

Este atributo permite asignar un identificador único a un elemento contenido en el script de una página web, lo cual es analizado por el Standard Generalized Mark-up Language (SGML) es muy parecido a la asignación de nombres que se le da a los objetos en un lenguaje de programación. Por ejemplo:

```
<p id="email">mtorres@queue-system.com</p>
<p id="movil">(051) 982-948948</p>
```

#### ◦ Atributo class

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. Por ejemplo:

##### 1.- Implementamos un estilo

```
<style>
  p.info      { color: green }
  p.advertencia { color: blue }
  p.error     { color: red }
</style>
```

##### 2.- Aplicamos el estilo

```
<p class="info"> Mensaje de Información </p>
<p class="advertencia"> Mensaje de Advertencia </p>
<p class="error"> Mensaje de Error </p>
```

#### ◦ Atributo title

Permite establecer un mensaje a un determinado elemento web; es decir, mostrará alguna información de apoyo al usuario sobre un determinado elemento, claro está que solo será visible cuando el usuario posicione el puntero del mouse sobre dicho elemento. Por ejemplo:

```
<a href="mailto:comentarios@queue-system.com"
  title="Servicio de atención al cliente">
  Remítanos sus comentarios
</a>
```

### 1.2.8 Especificación DOCTYPE

La declaración DOCTYPE le informa al navegador web con qué versión de HTML está escrita dicha página, esto lo realiza basándose en la especificación DTD, más conocida como Document Type Definition; esta define las reglas usadas por HTML5 en un documento web y será interpretado correctamente por el servidor web. Su formato es:

```
<!DOCTYPE html>
```

Las versiones anteriores al HTML5 contaban con el siguiente formato:

- **Doctype HTML 4.01 strict**

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

- **Doctype HTML 4.01 transitional**

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

### 1.2.9 La etiqueta HEAD

La etiqueta HEAD permite definir la cabecera del documento web y normalmente contiene información que no es visible para el usuario. HEAD pertenece a la estructura principal de un documento web; por lo tanto, debe ser colocado antes de la etiqueta <body> y dentro de <html>.

El formato completo de un documento web en HTML5 es como se muestra en el siguiente script:

```
<!DOCTYPE html>
<html>
  <head>
    Cabecera del documento
  </head>
  <body>
    Cuerpo del Documento.
  </body>
</html>
```

Lo que podemos asignar en el bloque HEAD puede ser:

```
<TITLE> Libro HTML5 </TITLE>
<META charset="utf-8" />
<LINK rel="stylesheet" href="estilo.css" />
```

Donde:

- **Title:** Define el título de la ventana al ejecutar el documento web.
- **Meta charset="utf-8":** Define el uso de tildes y eñes en el documento web.

- **Link:** Permite asociar un archivo al documento web actual, por ejemplo, para asociar un documento web a un archivo CSS, tendríamos el siguiente código:

```
<link href="estilo.css" rel="stylesheet" >
```

### 1.2.10 La etiqueta BODY

La etiqueta BODY es un espacio de trabajo que contiene todo lo visible en 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. 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 PHP, Java, ASP, etc.

### 1.2.11 ¿Qué elementos podemos colocar dentro del BODY?

El contenido de BODY ha sido mejorado para HTML5 haciéndolo semántico; es decir, permite programar en HTML5 con sentido semántico y entendible. Veamos las etiquetas relevantes de HTML5:



- **HEADER**

Es la etiqueta que conforma la cabecera de una sección o artículo de un documento web basado en HTML5.

- **NAV**

Esta etiqueta sirve para incluir menús de navegación en diferentes partes de nuestro sitio.

- **SECTION**

Permite crear secciones dentro de las demás etiquetas, organizando de la mejor manera la información que se muestre en un documento web.

- **ARTICLE**

Un artículo es parecido a una sección en la cual podríamos colocar noticias, entradas del blog, trabajo de un portafolio o un contenido dinámico agregado con cierta frecuencia.

- **ASIDE**

Es similar a una sección con la diferencia que se puede implementar como una barra lateral.

- **FOOTER**

Es el pie de página de una sección, artículo, etc.

Veamos el script de una página web básica con HTML5:

```
<!DOCTYPE html>
<html lang="es">
<HEAD>
  <meta charset="utf-8" />
  <title>Estructura básica de una web - HTML5</title>
</HEAD>

<BODY>
  <HEADER>
    <NAV>
      <a href="opcion1.html"> Opcion 1 </a>
      <a href="opcion2.html"> Opcion 2 </a>
      <a href="opcion3.html"> Opcion 3 </a>
      <a href="opcion4.html"> Opcion 4 </a>
    </NAV>
  </HEADER>
  <ASIDE>
    <p> Contenido aside </p>
  </ASIDE>
  <SECTION>
    <p> División de contenidos </p>
    <ARTICLE>
      <h1>Nombre del Articulo</h1>
      <p> Contenido del Articulo </p>
    </ARTICLE>
  </SECTION>
  <FOOTER>
    <p>Derechos reservados</p>
  </FOOTER>
</BODY>
</html>
```

### 1.2.12 Comentarios en HTML5

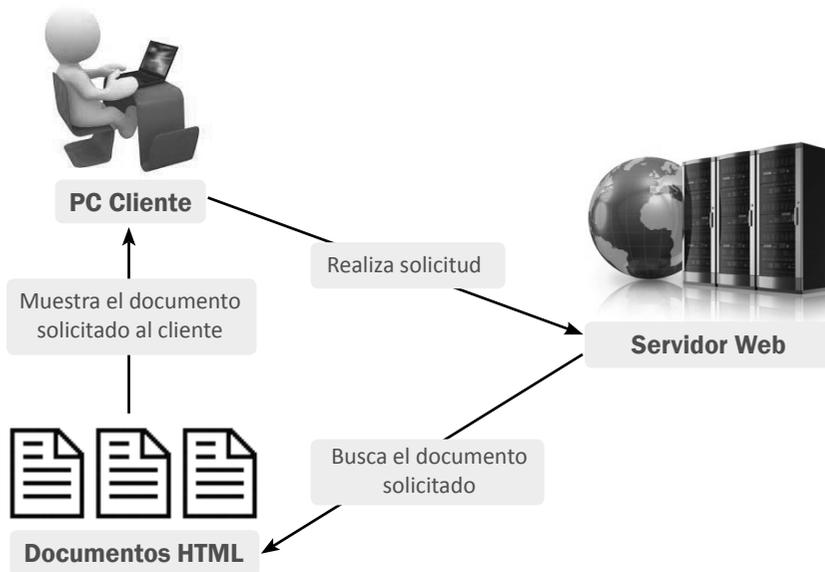
Un comentario permite asignar mensajes en un documento web con el fin de recordar algunos detalles del código, o simplemente mostrar un texto en cualquier parte del documento HTML5. El formato para asignar un comentario es:

```
<!-- Comentario de una línea-->
```

```
<!-- Comentario
de
varias
líneas
-->
```

## 1.3 FUNCIONAMIENTO DE UN SERVIDOR WEB

Un servidor web es aquel que presta servicios a los clientes, una de sus funciones principales es almacenar archivos pertenecientes a un sitio web y mostrarlo por la red, y así poder ser visitado por los usuarios en el mundo. Tal como se muestra en la siguiente imagen:



## 1.4 INTRODUCCIÓN AL APACHE

El servidor Apache es considerado un servidor web de código abierto y de libre distribución; que puede ser usado en sistemas como Windows, Linux, Macintosh y otros.

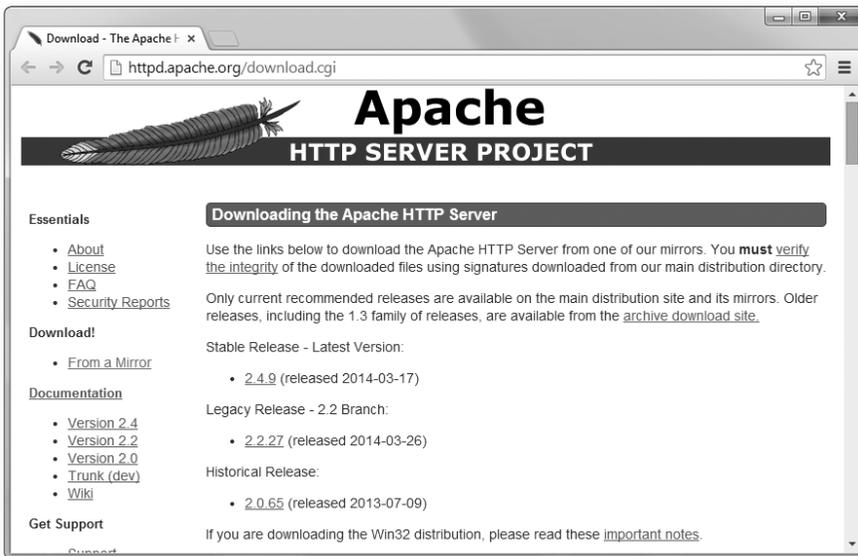
Apache es una aplicación que permite implementar un servidor web en su computadora personal, asignándole un nivel de servidor local no importando el sistema operativo donde se encuentra, ya que tiene compatibilidad abierta. Su más cercana competencia es el Internet Information Server, más conocido como IIS, que pertenece a Microsoft y tiene las mismas funcionalidades de apache, pero es considerado un *software* propietario; por lo tanto solo funciona para sistemas Microsoft.

Apache tiene una fuerte afinidad con el lenguaje de programación PHP por eso tiene librerías que soportan al PHP. Entre las características principales que presenta podemos mencionar:

- Un completo soporte para el lenguaje de programación PHP.
- Incorpora módulos de autenticación web, como el `mod_access`, `mod_auth` y `mod_digest`.
- Presenta un soporte para certificados SSL y TLS.
- Permite la configuración de mensajes de errores personalizados y negociación de contenido.
- Permite autenticación de base de datos basada en SGBD.

#### 1.4.1 Descargar servidor Apache

Para iniciar la descarga del servidor Apache, debemos ingresar a la siguiente URL <http://httpd.apache.org/download.cgi> tal como se muestra en la siguiente imagen:



Debemos considerar que el servidor Apache cuenta con licencia GPL; por lo tanto la descarga es *freeware*, y no estaríamos incurriendo en faltas.

#### Apache HTTP Server 2.0.65 Final is also available 2013-07-09

Apache 2.0.65 is the final historical release of the 2.0 series, and is recommended over any previous 2.0 release. No further releases will occur, and all users are directed to install stable 2.4 or legacy 2.2 releases instead. This release fixes a few potential security vulnerabilities.

For details see the [Official Announcement](#) and the [CHANGES\\_2.0](#) and [CHANGES\\_2.0.65](#) lists.

Apache 2.0 add-in modules are not compatible with Apache 2.2 modules. If you are running third party add-in modules, you will need to obtain modules compiled for or compatible with Apache 2.0 from that third party, before you attempt to use this specific release.

- Source: [httpd-2.0.65.tar.gz](http://httpd-2.0.65.tar.gz) [ PGP ] [ MD5 ]
- Source: [httpd-2.0.65.tar.bz2](http://httpd-2.0.65.tar.bz2) [ PGP ] [ MD5 ]
- Win32 Source: [httpd-2.0.65-win32-src.zip](http://httpd-2.0.65-win32-src.zip) [ PGP ] [ MD5 ]
- Win32 Binary without crypto (no mod\_ssl) (MSI Installer): [httpd-2.0.65-win32-x86-no\\_ssl.msi](http://httpd-2.0.65-win32-x86-no_ssl.msi) [ PGP ] [ MD5 ] [ SHA1 ]
- Win32 Binary including OpenSSL 0.9.8y (MSI Installer): [httpd-2.0.65-win32-x86-openssl-0.9.8y.msi](http://httpd-2.0.65-win32-x86-openssl-0.9.8y.msi) [ PGP ] [ MD5 ] [ SHA1 ]
- NetWare Binary: [apache\\_2.0.65-netware.zip](http://httpd-2.0.65-netware.zip) [ PGP ] [ MD5 ] [ SHA1 ]
- [Security and official patches](#)
- [Other files](#)

En la página de descargas del servidor Apache debemos buscar la sección **Apache HTTP Server 2.0.65 Final is also available** y seleccionaremos `httpd-2.0.65-win32-x86-no_ssl.msi`, tal como se muestra a continuación:

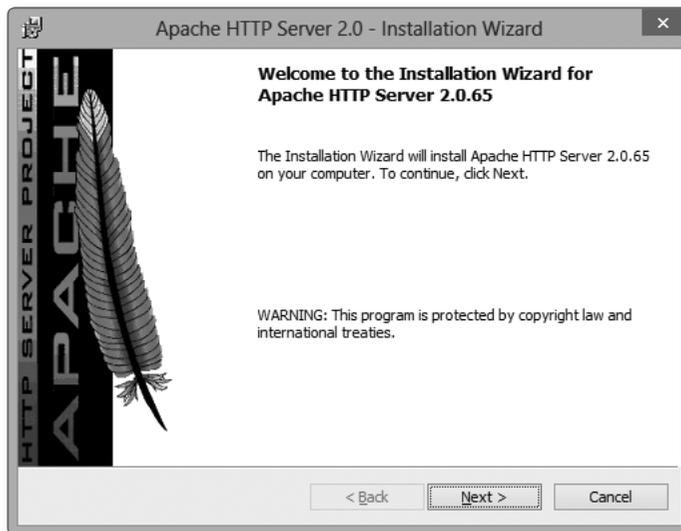
Win32 Binary without crypto (no mod\_ssl) (MSI Installer): [httpd-2.0.65-win32-x86-no\\_ssl.msi](#) [ PGP ] [ MD5 ] [ SHA1 ]

Finalmente la descarga se inicia, luego obtendremos el instalador del servidor Apache tal como se muestra en la siguiente imagen:



### 1.4.2 Instalación del servidor Apache

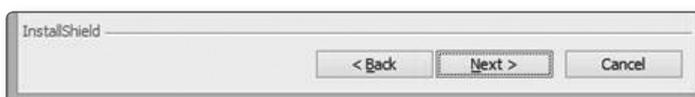
Vamos a iniciar el proceso de instalación del servidor Apache, la versión 2.0.65 ha sido probada para esta material usando el sistema operativo Windows 8.



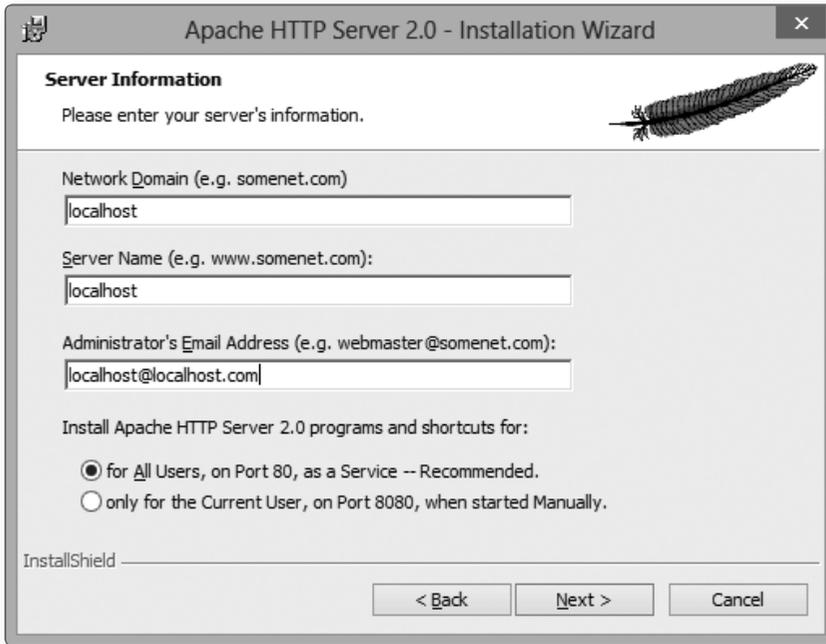
**Paso 1:** Debemos seleccionar el botón **Next>** de la ventana de bienvenida al asistente de instalación de Apache.



**Paso 2:** Seleccionar la opción «I accept the terms in the license agreement» y presionar el botón **Next>**.



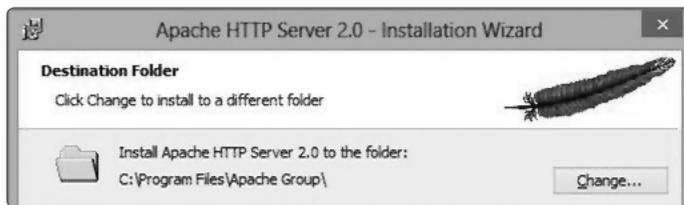
**Paso 3:** En esta ventana solo presionar Next>.



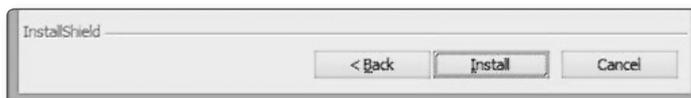
**Paso 4:** Debemos brindar información solicitada por el servidor de la siguiente manera: **Network Domain** colocar localhost, **Server Name** colocar localhost y **Administrator's Email Address** también asignar localhost@localhost.com.



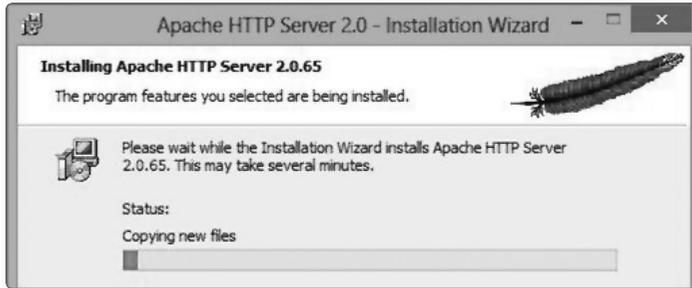
**Paso 5:** Ahora nos toca seleccionar el tipo de instalación, para lo cual elegiremos **Typical**. No es necesario hacer una personalización en la instalación.



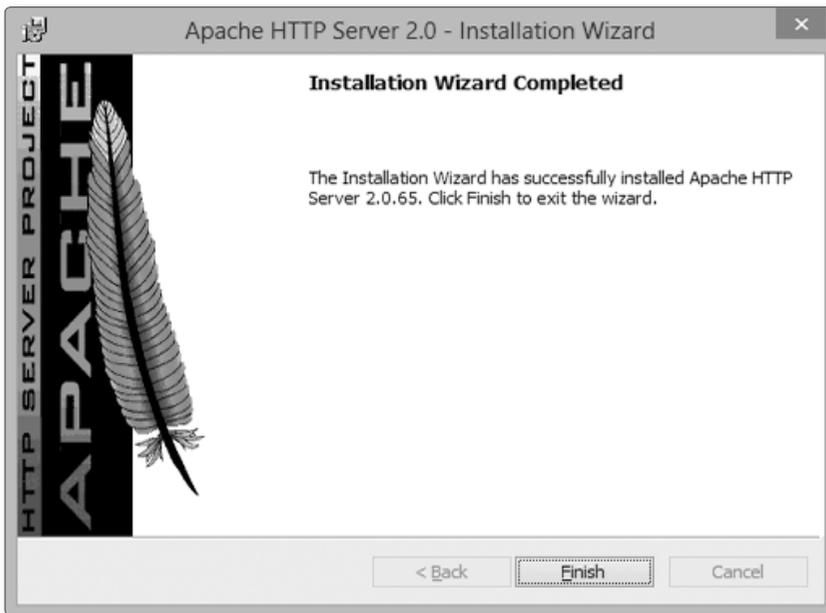
**Paso 6:** Seleccionaremos el lugar destino de la instalación, para ello notará que el asistente de instalación toma como carpeta predeterminada a **c:\Program files\Apache Group**; dejaremos dicha ruta como se muestra en la imagen, a menos que usted decida cambiarlo, para finalmente dar clic en el botón **Next>**.



**Paso 7:** Estamos listos para iniciar la instalación del servidor luego de haber seguido los pasos del asistente.



**Paso 8:** Vemos cómo se inicia el proceso de instalación del servidor Apache en el sistema operativo Windows 8.



**Paso 9:** Para finalizar la instalación debemos seleccionar el botón **Finish** de la ventana «Installation Wizard Completed».

### 1.4.3 Pruebas del servidor Apache

Para comprobar si el servidor trabaja correctamente, debemos ingresar a un navegador web, como Internet Explorer, Mozilla Firefox o Google Chrome y colocar la siguiente **URL > localhost**, si todo es correcto usted deberá estar visualizando la siguiente imagen.



En caso no se muestre la imagen anterior, debemos revisar que otro servidor no se encuentre activo en nuestro sistema, como es el IIS (Internet Information Server) de Windows; y que para que funcione correctamente el Apache debemos inhabilitar el servicio IIS desde el panel de control de Windows.

## 1.5 CASOS DESARROLLADOS DE SCRIPT HTML5 EJECUTADOS DESDE EL SERVIDOR APACHE

Implementaremos casos prácticos de documentos web en HTML5 y CSS3 usando como servidor a Apache.

### • Caso desarrollado 1: Menú de opciones vertical simple

Implemente una página web con HTML5 que permita mostrar un menú de opciones para el sistema de control de cliente, tal como se muestra en la siguiente imagen:



**Consideraciones:**

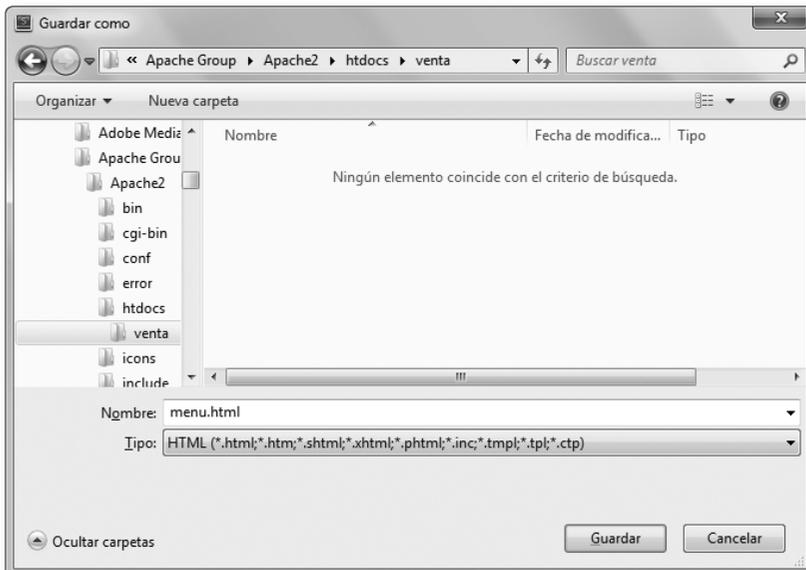
- ◆ Las opciones del menú deben presentarse en una lista desordenada.
- ◆ Las opciones del menú deben presentarse de forma vertical.
- ◆ El enlace de cada menú será a un archivo con el mismo nombre, por ejemplo si la opción del menú es **Listado de clientes**, su enlace sería **listado.html**.

**Solución:**

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
menu.html
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <link rel="stylesheet" href="estilo.css" type="text/css">
5 <meta charset="utf-8"/>
6 <title>Sistema de Control de Clientes</title>
7 </head>
8 <body>
9 <HEADER id="encabezado">
10 <h1>Sistema de control de clientes</h1>
11 <nav>
12 <ul>
13 <li><a href="principal.html">Principal</a></li>
14 <li><a href="listado.html">Listado de clientes</a></li>
15 <li><a href="registro.html">Registro del nuevo cliente</a></li>
16 <li><a href="actualiza.html">Actualización de datos</a></li>
17 </ul>
18 </nav>
19 </HEADER>
20 </body>
21 </html>
22
```

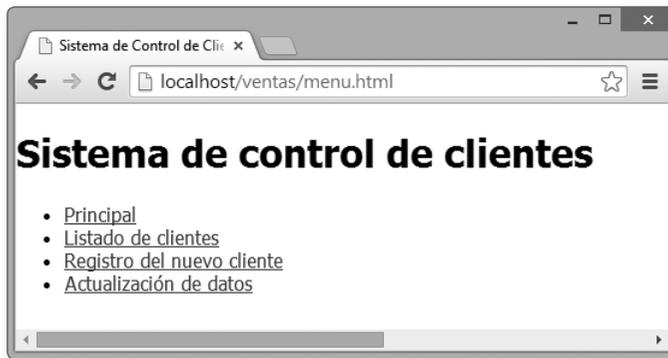
Le asignaremos el nombre de **menu.html** y lo registraremos en la carpeta **htdocs** de Apache, el cual se encuentra ubicado en **Archivos de Programa > Apache Group > Apache2 > htdocs**. Es recomendado crear carpetas para una mejor administración de los documentos web, tal como se muestra en la siguiente imagen:



**Paso 2:** Una vez colocado el archivo en la carpeta `htdocs`, procederemos a ejecutar el documento web, para lo cual debemos ingresar a un navegador web como podría ser Google Chrome, Mozilla Firefox o Internet Explorer y colocar el siguiente URL:

**`http://localhost/venta/menu.html`**

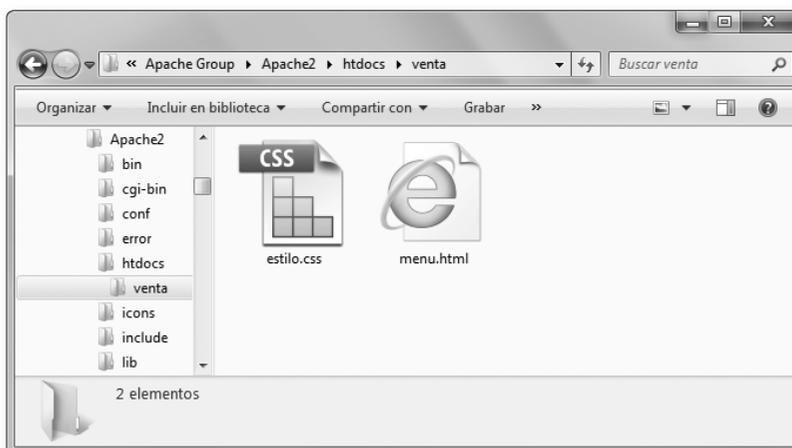
La siguiente imagen muestra el resultado de invocar al archivo `menu.html` ejecutado desde el servidor Apache.



**Paso 3:** De la misma manera podríamos crear un archivo llamado `estilo.css` que formateará los textos mostrados en el archivo `menu.html`; el script sería de la siguiente manera:

```
menu.html  estilo.css
1  body
2  {
3      width:960px;
4      margin:2px auto;
5      font-family: Tahoma;
6  }
7
8  #encabezado
9  {
10     padding:0;
11     margin: 0 2px 0 0;
12 }
```

No olvide guardar el archivo `estilo.css` en la carpeta `venta` del Apache; es decir, ahora tendremos dos archivos, tal como se muestra en la siguiente imagen:



## ◉ Caso desarrollado 2: Menú de opciones horizontal

Implemente una página web con HTML5 que permita mostrar un menú de opciones para el Sistema de control de cliente, tal como se muestra en la siguiente imagen:



### Consideraciones:

- ◊ Las opciones del menú deben presentarse en una lista desordenada.
- ◊ Las opciones del menú deben presentarse de forma horizontal.
- ◊ Las opciones del menú presentan un aspecto aplicado con estilo en la cual se configura color de fondo y color al texto.
- ◊ El enlace de cada menú será dirigido a un archivo con el mismo nombre, por ejemplo si la opción del menú es **Listado de clientes** su enlace sería **listado.html**.

### Solución:

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
menuh.html  estiloh.css
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <link type="text/css" href="estiloh.css" rel="stylesheet">
5     <meta charset="utf-8"/>
6     <title>Sistema de Control de Clientes</title>
7   </head>
8   <body>
9     <header>
10      <nav id="menu">
11        <ul>
12          <li><a href="principal.html">Principal</a></li>
13          <li><a href="listado.html">Listado de clientes</a></li>
14          <li><a href="registro.html">Registro del nuevo cliente</a></li>
15          <li><a href="actualiza.html">Actualización de datos</a></li>
16        </ul>
17      </nav>
18    </header>
19  </body>
20 </html>
```

Le asignaremos el nombre de **menue.html** y lo registraremos en la carpeta venta de **htdocs**, el cual se encuentra ubicado en **Archivos de Programa > Apache Group > Apache2 > htdocs > venta**.

**Paso 2:** Ejecutar el archivo **menue.html** colocando el siguiente URL en el navegador:

**<http://localhost/venta/menuh.html>**

**Paso 3:** No debemos olvidarnos de dar el formato adecuado para la presentación del menú horizontal usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
menuh.html  estiloh.css
1 #menu li {
2     display: inline-block;
3 }
4
5 #menu a {
6     display: block;
7     float: left;
8     margin-right: 0.12em;
9     padding: 1em 1.5em;
10    background: #178EAE;
11    box-shadow: inset 0px -2px 0px 0px rgba(0,0,0,0.25);
12    text-decoration: none;
13    text-shadow: 1px 1px 0px #0C749C;
14    text-transform: uppercase;
15    font-size: .90em;
16    color: #FFFFFF;
17 }
18
19 body {
20    margin: 0em;
21    padding: 0em;
22    background: #1EB0D7;
23    font-family: "Tahoma";
24    font-size: 10pt;
25    color: #525252;
26 }
```

Le llamaremos **estiloh.css** y lo guardaremos en el mismo lugar que el archivo **menuh.html**; es decir, en la carpeta **venta** del servidor Apache.

### • Caso desarrollado 3: Menú de opciones vertical con resaltado desde el puntero del mouse

Implemente una página web con HTML5 que permita mostrar un menú de opciones para el sistema de control de cliente, tal como se muestra en la siguiente imagen:



#### Consideraciones:

- ◆ Las opciones del menú deben presentarse en una lista desordenada.
- ◆ Las opciones del menú deben presentarse de forma vertical.
- ◆ Al pasar por las opciones del menú, cambiará de aspecto.
- ◆ Las opciones del menú presentan un aspecto aplicado con estilo, en el cual se configura el color de fondo y el color del texto.
- ◆ El enlace de cada menú será dirigido a un archivo con el mismo nombre, por ejemplo, si la opción del menú es **Listado de clientes** su enlace sería **listado.html**.

**Solución:**

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
menue.html x  estiloe.css x
1 <!DOCTYPE html>
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <link rel="stylesheet" href="estiloe.css">
5 <meta charset="utf-8" />
6 <title>Sistema de Control de Clientes</title>
7 </head>
8 <body>
9 <aside>
10 <NAV class="aside-nav">
11 <ul>
12 <li><a href="principal.html">Principal</a></li>
13 <li><a href="listado.html">Listado de clientes</a></li>
14 <li><a href="registro.html">Registro del nuevo cliente</a></li>
15 <li><a href="actualiza.html">Actualización de datos</a></li>
16 </ul>
17 </NAV>
18 </aside>
19 </body>
20 </html>
```

Le asignaremos el nombre de **menue.html** y lo registraremos en la carpeta **venta** de **htdocs**, el cual se encuentra ubicado en **Archivos de Programa > Apache Group > Apache2 > htdocs > venta**.

**Paso 2:** Ejecutar el archivo **menue.html** colocando el siguiente URL en el navegador:

**<http://localhost/ventas/menue.html>**

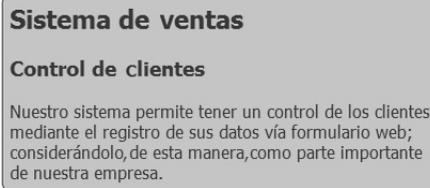
**Paso 3:** No debemos olvidarnos de dar formato adecuado para la presentación del menú usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
menue.html x  estiloe.css x
1 /* Estilo de Fondo */
2 body {
3     background: #d3d3d3;
4     font-size: 100%;
5     margin: 0;
6     font-family:tahoma;
7     color: #444;
8 }
9
10 /* Estilo para la lista */
11 .aside-nav ul {
12     background: #e8e8e8;
13     padding: 0;
14     width: 25%;
15 }
16
17 /* Estilo para los elementos de la lista */
18 .aside-nav li a {
19     display: block;
20     padding: 1.25em;
21     border-bottom: 1px solid #D2D2D2;
22 }
23
24 /* Estilo al posicionar el puntero encima del enlace*/
25 .aside-nav a:hover {
26     color: #fff;
27 }
```

Le llamaremos **estiloe.css** y lo guardaremos en el mismo lugar que el archivo **menue.html**; es decir, en la carpeta **venta** del servidor Apache.

#### ◦ Caso desarrollado 4: Sección con HTML5

Implemente una página web con HTML5 que permita mostrar un comentario sobre el sistema de ventas, tal como se muestra en la siguiente imagen:



#### Consideraciones:

- ◊ Debe usar la etiqueta `section` del HTML5.
- ◊ Los textos del título, subtítulo y contenido presentan estilos CSS.

#### Solución:

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
seccion.html x estilo_seccion.css x
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <link rel="stylesheet" href="estilo_seccion.css">
5 <meta charset="utf-8" />
6 <title>Manejo de la etiqueta SECTION</title>
7 </head>
8 <body>
9 <SECTION>
10 <header>
11 <h1>Sistema de Ventas</h1>
12 </header>
13 <article>
14 <h3>Control de Clientes</h3>
15 <p>
16 <span style="display: block; padding-left: 20px;">Nuestro sistema permite tener un control de los clientes mediante el
registro de sus datos vía formulario web; considerándolo de esta manera
como parte importante de nuestra empresa.
17 </span>
18 </p>
19 </article>
20 </SECTION>
21 </body>
22 </html>
```

Le asignaremos el nombre de `seccion.html` y lo registraremos en la carpeta `venta` de `htdocs`, el cual se encuentra ubicado en `Archivos de Programa > Apache Group > Apache2 > htdocs > venta`.

**Paso 2:** Ejecutar el archivo `seccion.html` colocando el siguiente URL en el navegador:

<http://localhost/venta/seccion.html>

**Paso 3:** No debemos olvidarnos de dar formato adecuado para la presentación de la sección usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
seccion.html x estilo_seccion.css x
1 /* Estilo de Fondo */
2 body {
3     background: #d3d3d3;
4     font-size: 100%;
5     margin: 10px;
6     font-family:tahoma;
7     color: #444;
8 }
9 /* Estilo para la seccion */
10 section{
11     max-width:400px;
12 }
```

Le llamaremos `estilo_seccion.css` y lo guardaremos en el mismo lugar que el archivo `seccion.html`; es decir, en la carpeta `venta` del servidor Apache.

#### ◦ Caso desarrollado 5: Artículo con HTML5

Implemente una página web con HTML5 que permita mostrar un comentario realizado por un cliente sobre el sistema de ventas, tal como se muestra en la siguiente imagen:

#### Foro de comentarios

##### Cliente opina:

"El servicio prestado por la empresa es totalmente profesional, recomiendo realizar sus compras con total confianza...!"

15 de abril 2015

#### Consideraciones:

- ◊ Debe usar la etiqueta `article` del HTML5.
- ◊ Debe usar la etiqueta `time` del HTML5 para mostrar la fecha actual.
- ◊ Los textos del título, subtítulo y contenido presentan estilos CSS.

#### Solución:

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
articulo.html x estilo_articulo.css x
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <link rel="stylesheet" href="estilo_articulo.css">
5     <meta charset="utf-8" />
6     <title>Manejo de la etiqueta ARTICLE</title>
7 </head>
8 <body>
9     <ARTICLE>
10     <header>
11         <h1>Foro de comentarios</h1>
12         <h3>Cliente opina:</h3>
13     </header>
14     <section>
15         <p>
16             "El servicio prestado por la empresa es totalmente profesional, recomiendo realizar
17             sus compras con total confianza...!"
18         </p>
19     </section>
20     <p><time datetime="2015-04-15">15 de abril 2015</time></p>
21 </ARTICLE>
22 </body>
23 </html>
```

Le asignaremos el nombre de `articulo.html` y lo registraremos en la carpeta `venta` de `htdocs`, el cual se encuentra ubicado en `Archivos de Programa > Apache Group > Apache2 > htdocs > venta`.

**Paso 2:** Ejecutar el archivo `articulo.html` colocando el siguiente URL en el navegador:

**`http://localhost/venta/articulo.html`**

**Paso 3:** No debemos olvidarnos de dar formato adecuado para la presentación del artículo usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
articulo.html x estilo_articulo.css x
1 /* Estilo de Fondo */
2 body {
3     background: #d3d3d3;
4     font-size: 100%;
5     margin: 10px;
6     font-family:tahoma;
7     color: #444;
8 }
9 /* Estilo para el articulo */
10 article{
11     max-width:400px;
12 }
```

Le llamaremos `estilo_articulo.css` y lo guardaremos en el mismo lugar que el archivo `articulo.html`; es decir, en la carpeta `ventas` del servidor Apache.

#### • Caso desarrollado 6: Pie de página con HTML5

Implemente una página web con HTML5 que permita mostrar el pie de página tal como se muestra en la siguiente imagen:

Copyright © 2015 – Todos los derechos reservados - Editorial Macro

Diseñado por Lic. Manuel Torres R.

#### Consideraciones:

- ◆ Debe usar la etiqueta `footer` del HTML5.
- ◆ Debe controlar la alineación del texto en la misma línea, usando estilos CSS.
- ◆ Todos los textos presentan estilos CSS.

#### Solución:

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
pie.html x estilo_pie.css x
1 <!DOCTYPE html>
2 <head>
3     <link href="estilo_pie.css" rel="stylesheet">
4     <meta charset="utf-8" />
5     <title>Manejo de FOOTER</title>
6 </head>
7 <body>
8     <FOOTER>
9         <p class="fl_left">Copyright &copy; 2015 -
10             Todos los derechos reservados -
11             <a href="http://www.editorialmacro.com">Editorial Macro</a></p>
12         <p class="fl_right">Diseñado por Lic. Manuel Torres R.</p>
13     </FOOTER>
14 </body>
15 </html>
```

Le asignaremos el nombre de `pie.html` y lo registraremos en la carpeta `venta` de `htdocs`, el cual se encuentra ubicado en `Archivos de Programa > Apache Group > Apache2 > htdocs > venta`.

**Paso 2:** Ejecutar el archivo `pie.html` colocando el siguiente URL en el navegador:

**`http://localhost/venta/pie.html`**

**Paso 3:** No debemos olvidarnos de dar el formato adecuado para la presentación del pie usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```

pie.html x estilo_pie.css x
1 /* Estilo para la alineacion de los textos*/
2 .fl_left{
3     float:left;
4 }
5 .fl_right{
6     float:right;
7 }
8
9 /* Estilo para todo el documento */
10 body{
11     font-size:13px;
12     font-family:Tahoma, Geneva, sans-serif;
13     color:#FFF;
14     background-color:#000;
15 }
16
17 /* Estilo para los enlaces */
18 a:link{ color:#FF0; }
19 a:visited{ color:#FF0; }
20
21 /* Estilo para el pie */
22 footer {
23     display:block;
24     margin:10px;
25 }

```

Le llamaremos `estilo_pie.css` y lo guardaremos en el mismo lugar que el archivo `pie.html`; es decir, en la carpeta `venta` del servidor apache.

### ◦ Caso desarrollado 7: Compra de productos con tablas

Implemente una página web con HTML5 que permita mostrar el resumen de compras realizada por un determinado cliente, tal como se muestra en la siguiente imagen:

PRODUCTOS	DESCRIPCIÓN	PRECIO	CANTIDAD	TOTAL
	Television 42 pulgadas de alta resolución COD: 2612215412	S/. 1699.00	01 ACTUALIZAR	S/. 1699.00
	BlueRay SONY COD: 2065150948198	S/. 399.00	01 ACTUALIZAR	S/. 399.00
			SUBTOTAL	S/. 2098.00
<div style="display: flex; align-items: center;"> <span style="background-color: #333; color: white; padding: 5px 10px; border-radius: 5px;">Continuar</span> <span style="margin-left: 10px;">▶</span> </div>				

### Consideraciones:

- ◊ Debe usar tablas para la presentación de los productos.
- ◊ Los botones **actualizar** y **continuar** los puede obtener de la galería de imágenes del [www.google.com](http://www.google.com).
- ◊ Todos los textos presentan estilos CSS.



**Paso 3:** No debemos olvidarnos de dar el formato adecuado para la presentación de la tabla usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
1 BODY{
2     font-family:Tahoma, Geneva, sans-serif;
3     font-size:12px;
4 }
5 #CENTRADO{
6     text-align:center;
7     color:#000;
8 }
9 #COLORFILAS{
10     background-color:#999;
11 }
```

Le llamaremos `estilo_tabla.css` y lo guardaremos en el mismo lugar que el archivo `tabla.html`; es decir, en la carpeta `venta` del servidor Apache.

### ◦ Caso desarrollado 8: Formulario de registro de usuarios

Implemente una página web con HTML5 que permita mostrar un formulario de registro de usuario, tal como se muestra en la siguiente imagen:



Formulario de registro de usuarios  
Complete el formulario con sus datos

**Nombres y apellidos**  
ingrese datos completos

**DNI**  
8 caracteres

**Teléfono**

**Estudios realizados**  
 Primaria  
 Secundaria  
 Técnico  
 Universitario

**Estado civil**  
 Soltero  
 Casado  
 Viudo  
 Divorciado

**Provincia**

**Correo electrónico**

**Clave personal**  
máximo 6 caracteres

**Color de preferencia**

**Fecha de nacimiento**  
dd/mm/aaaa --:--:--

REGISTRAR LIMPIAR CONTROLES

### Consideraciones:

- ◊ Debe usar la etiqueta `form` como plataforma del modelo.
- ◊ Todos los elementos son controles de formularios como caja de texto, botones, etc.
- ◊ Todos los textos y controles de formularios presentan estilos CSS.

**Solución:**

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```
formulario.html  estilo_formulario.css  x
1  <!doctype html>
2  <html>
3  <head>
4      <title>formulario de registro de usuarios</title>
5      <link href="estilo_formulario.css" rel="stylesheet">
6      <meta charset="utf-8">
7  </head>
8  <body>
9  <section id="formulario">
10     <form name="frmregistro" method="post" action="principal.html">
11         <h1>Formulario de registro de usuarios</h1>
12         <label id="label">Complete el formulario con sus datos</label>
13         <p id="linea">&nbsp;</p>
14
15         <label id="label">Nombres y apellidos
16             <input name="txtnombres" type="text" id="caja" size="60"
17                 placeholder="ingrese datos completos" required/>
18         </label>
19
20         <label id="label">DNI
21             <input type="text" id="caja" name="txtdni" Size="60"
22                 placeholder="8 caracteres" disabled/>
23         </label>
24         <label id="label">Telefono
25             <input type="tel" id="caja" size="60" name="txtfono"/>
26         </label>
27         <label id="label">Estudios realizados </label>
28         <input type="checkbox" name="chkprimaria"
29             checked="checked"/> Primaria
30         </label>
31         <label id="label">
32             <input type="checkbox" name="chksecundaria" /> Secundaria
33         </label>
34         <label id="label">
35             <input type="checkbox" name="chktecnico" /> Técnico
36         </label>
37
38         <input type="checkbox" name="chkuniversitario" /> Universitario
39         </label>
40         <br/>
41         <label id="label">Estado civil</label>
42         <label id="label">
43             <input type="radio" name="btneestado" value="s"
44                 Checked="checked"/> Soltero
45         </label>
46         <label id="label">
47             <input type="radio" name="btneestado" value="c" /> Casado
48         </label>
49         <label id="label">
50             <input type="radio" name="btneestado" value="v" /> Viudo
51         </label>
52         <label id="label">
53             <input type="radio" name="btneestado" value="d" /> Divorciado
54         </label>
55     <br/>
56
57     <label id="label">Provincia</label>
58     <input type="text" name="txtprovincia" list="provincias"/>
59     <datalist id="provincias">
60         <option value="Lima" />
61         <option value="Arequipa" />
62         <option value="Tacna" />
63         <option value="Trujillo" />
64         <option value="Huancayo" />
65     </datalist>
```

```
66 <label id="label">Correo electronico
67 <input type="email" id="caja" size="60" name="txtemail"/>
68 </label>
69
70 <label id="label">Clave personal
71 <input type="password" id="caja"
72 name="txtclave" size="60"
73 Placeholder="mximo 6 caracteres "/>
74 </label>
75 <label id="label">Color de preferencia
76 <input type="color" name="txtcolor"/>
77 </label>
78 <label id="label">Fecha de nacimiento
79 <input type="datetime-local" name="txtfecha"/>
80 </label>
81 <input type="submit" value="REGISTRAR">
82 <input type="reset" value="LIMPIAR CONTROLES">
83 </form>
84 </section>
85 </body>
86 </html>
```

Le asignaremos el nombre de `formulario.html` y lo registraremos en la carpeta `venta` de `htdocs`, el cual se encuentra ubicado en `Archivos de Programa > Apache Group > Apache2 > htdocs > venta`.

**Paso 2:** Ejecutar el archivo `formulario.html` colocando el siguiente URL en el navegador:

<http://localhost/venta/formulario.html>

**Paso 3:** No debemos olvidarnos de dar el formato adecuado para la presentaci3n del formulario usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente c3digo:

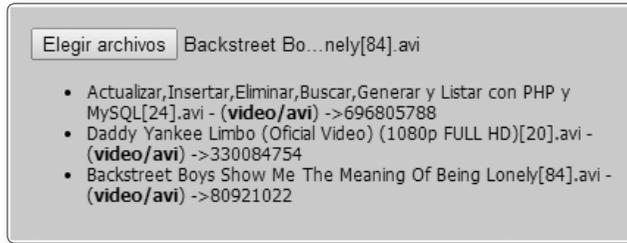
```
formulario.html  estilo_formulario.css
1 /*Estilo para el fondo de la pagina web*/
2 body{
3   font-family:"tahoma";
4   font-size:12px;
5 }
6
7 /*Estilo para el formulario*/
8 #formulario{
9   border:solid 2px #b7ddf2;
10  background:#b7ddf2;
11  margin:0 auto;
12  width:400px;
13  padding: 10px;
14 }
15
16 /*Estilo para el titulo*/
17 h1{
18   font-size:16px;
19   font-weight:bold;
20   margin-bottom:2px;
21 }
22
23 /*Estilo para el texto y su linea*/
24 #linea{
25   color:#666666;
26   margin-bottom:5px;
27   border-bottom:solid 1px #FFF;
28   padding-bottom:5px;
29 }
```

```
formulario.html x estilo_formulario.css x
1 /*Estilo para el fondo de la pagina web*/
2 body{
3     font-family:"tahoma";
4     font-size:12px;
5 }
6
7 /*Estilo para el formulario*/
8 #formulario{
9     border:solid 2px #b7ddf2;
10    background:#b7ddf2;
11    margin:0 auto;
12    width:400px;
13    padding: 10px;
14 }
15
16 /*Estilo para el titulo*/
17 h1{
18     font-size:16px;
19     font-weight:bold;
20     margin-bottom:2px;
21 }
22
23 /*Estilo para el texto y su linea*/
24 #linea{
25     color:#666666;
26     margin-bottom:5px;
27     border-bottom:solid 1px #FFF;
28     padding-bottom:5px;
29 }
30
31 /*Estilo para los textos del formulario*/
32 #label{
33     display: block;
34     font-weight: bold;
35     width: 240px;
36     font-size: 12px;
37 }
38 /*Estilo para la cajas de texto simple*/
39 #caja{
40     padding:4px;
41     border:solid 1px #aacfe4;
42     margin:2px 0 5px 0;
43 }
44
45 /*Estilo para el boton Registrar*/
46 #boton{
47     clear:both;
48     margin-left:150px;
49     width:125px;
50     height:31px;
51     background:#666666;
52     text-align:center;
53     line-height:31px;
54     color:#FFFFFF;
55     font-size:11px;
56     font-weight:bold;
57 }
```

Le llamaremos `estilo_formulario.css` y lo guardaremos en el mismo lugar que el archivo `formulario.html`; es decir, en la carpeta `venta` del servidor Apache.

### ◦ Caso desarrollado 9: Carga de archivos

Implemente una página web con HTML5 que permita mostrar los datos de los archivos que se cargan a un servidor web, tal como se muestra en la siguiente imagen:



#### Consideraciones:

- ◊ Use formulario para el registro de los archivos.
- ◊ Use código Javascript para obtener información de los archivos cargados en el servidor.
- ◊ Todos los textos presentan estilos CSS.

#### Solución:

**Paso 1:** Veamos el siguiente script implementado en la aplicación Sublime Text 2.0:

```

carga.html x  estilo_carga.css x
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Registro de carga de archivos</title>
5      <link href="estilo.css" rel="stylesheet">
6      <meta charset="utf-8">
7  </head>
8  <body>
9      <section id="formulario">
10         <input type="file" id="files" multiple>
11         <ARTICLE id="archivos">
12             <ul></ul>
13         </ARTICLE>
14     </SECTION>
15     <script>
16         var insertar_en = document.querySelector("#archivos ul");
17         file_in = document.querySelector("#files")
18         file_in.onchange = function(e){
19             var files = e.target.files;
20             for(var i=0,f= files[i];++i){
21                 var archivo = document.createElement("li");
22                 archivo.innerHTML = f.name + " - (<b>" + f.type + "</b>)" + f.size;
23                 insertar_en.appendChild(archivo);
24             }
25         }
26     </script>
27 </body>
28 </html>

```

Le asignaremos el nombre de `carga.html` y lo registraremos en la carpeta `ventas` de `htdocs`, el cual se encuentra ubicado en `Archivos de Programa > Apache Group > Apache2 > htdocs > venta`.

**Paso 2:** Ejecutar el archivo `carga.html` colocando el siguiente URL en el navegador:

**`http://localhost/venta/carga.html`**

**Paso 3:** No debemos olvidarnos de dar el formato adecuado para la presentación de la carga de archivos usando estilos, para esto debemos crear un archivo CSS que contenga el siguiente código:

```
carga.html x estilo_carga.css x
1  /*Estilo para el fondo de la pagina web*/
2  body{
3      font-family:"tahoma";
4      font-size:12px;
5  }
6
7  /*Estilo para el formulario*/
8  #formulario{
9      border:solid 2px #b7ddf2;
10     background:#b7ddf2;
11     margin:0 auto;
12     width:400px;
13     padding: 10px;
14 }
15
16 /*Estilo para el titulo*/
17 h1{
18     font-size:16px;
19     font-weight:bold;
20     margin-bottom:2px;
21 }
22
```

Le llamaremos `estilo_carga.css` y lo guardaremos en el mismo lugar que el archivo `carga.html`; es decir en la carpeta `venta` del servidor Apache.



Impreso en los talleres gráficos de



EDITORIAL  
**MACRO**®

Surquillo