



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

Universidad Nacional Autónoma de
México

FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN

***“MANUAL Y PRÁCTICAS PARA DESARROLLO
CON TECNOLOGÍAS WEB”***

TRABAJO MONOGRÁFICO
QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN:

PRESENTA:

GARCÍA SÁNCHEZ TANIA

ASESOR:

ING. ANTONIA NAVARRO GONZALEZ



FES Aragón

SAN JUAN DE ARAGÓN, ESTADO DE MÉXICO 2009



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Mi tesis la dedico con todo mi amor y cariño.

A Dios que me dio la oportunidad de vivir y regalarme una familia maravillosa.

Con mucho cariño principalmente a mis padres que me dieron la vida y han estado conmigo en todo momento. Gracias por todo papá y mamá por darme una carrera para mi futuro y por creer en mí, aunque hemos pasado momentos difíciles siempre han estado apoyándome y brindándome todo su amor, cariño y comprensión, por todo esto les agradezco de todo corazón y este trabajo que me llevo unos meses hacerlo es para ustedes, por ser su hija aquí esta lo que ustedes me brindaron solamente les estoy devolviendo lo que ustedes me dieron en un principio.

A mis hermanos Carlos Ignacio y Wendy gracias por estar conmigo y apoyarme siempre, yo se que cuento con ustedes siempre, los quiero mucho.

A mi abuelita Amparo por encomendarme siempre con Dios para que saliera adelante. Yo se que tus oraciones fueron escuchadas.

Agradecimientos.

A mis padres que me han motivado a seguir este camino, por sus enseñanzas desvelos, regaños...Se que sin ustedes no sería la persona que soy.

Mamá gracias porque tú eres quien me ha enseñado la motivación para no darme por vencida, siempre has estado ahí para escucharme y darme tu apoyo incondicional, en todas mis decisiones, muchas gracias.

Papá gracias porque tú eres quien me ha enseñado a tener objetivos principales en la vida y saber conseguirlos a base de esfuerzo y dedicación, se que una de tus satisfacciones es que obtuviera el título profesional, espero haber cumplido tus expectativas y que siempre estés orgulloso de mi.

A mi abuelita Amparo por que siempre ha estado apoyándonos y brindándonos todo su amor y cariño, muchas gracias.

A mis amigos Arturo, Erik, Carlos Alberto, muchas gracias por estar conmigo en todo este tiempo donde he vivido momentos felices y tristes, donde compartimos tantas aventuras, experiencias, desveladas y triunfos, gracias por ser mis amigos y recuerden que siempre los llevaré en mi corazón.

Gracias a cada uno por hacer que mi estancia en la FES Aragón fuera muy divertida.

A mi amigo Arturo por su apoyo incondicional, confianza y sobre todo por su sincera amistad, muchas gracias.

A mi directora de tesis Profesora Antonia Navarro González por su calidez, sugerencias, confianza, apoyo incondicional, su tiempo y sobre todo su sincera amistad.

A mi profesor y tutor de beca el Ing. Raúl Vera Barrón, por enseñarme que no hay límites, que lo que me proponga lo puedo lograr y que solo depende de mí.

A mis revisores Israel Nava, Ramón Patiño, Arcelia Bernal, quienes me brindaron su apoyo y confianza, en la realización de este trabajo.

Y a mis profesores por confiar en mí, por tenerme la paciencia necesaria, por su disposición y ayuda brindadas, gracias por apoyarme en momentos difíciles, por sus enseñanzas las cuales lograron en mí una formación, para enfrentarme hoy a los retos de mi profesión, muchas gracias. Agradezco el haber tenido unos profesores tan buenos como lo son ustedes. Nuca los olvidaré.

No me puedo ir sin antes decirles, que sin ustedes a mi lado no lo hubiera logrado, tantas desveladas sirvieron de algo y aquí está el fruto. Les agradezco a todos ustedes con toda mi alma el haber llegado a mi vida y el compartir momentos agradables y momentos tristes, pero esos momentos son los que nos hacen crecer y valorar a las personas que nos rodean. Los quiero mucho y nunca los olvidaré.

Donde una puerta se cierra, otra se abre.

(Miguel de Cervantes)

La gota abre la piedra, no por su fuerza sino por su constancia.

(Ovidio)

Si una persona es perseverante, aunque sea dura de entendimiento, se hará inteligente; y aunque sea débil se transformará en fuerte.

(Leonardo Da Vinci)

Lo que un hombre piensa de sí mismo, es lo que determina, o más bien indica, su destino.

(Henry David Thoreau)

ÍNDICE.

Introducción	1
Capítulo I HTML	Pág.
I.1 Estructura del documento	5
I.2 Estructura de un archivo HTML	6
I.3 Salto de Línea	8
I.4 Párrafos	9
I.5 Color de Fondo	10
I.6 Imagen de Fondo	11
I.7 Color de Texto	12
I.8 Comentarios	14
I.9 Marquesinas	15
I.10 Cómo cambiar la apariencia	16
I.11 Cómo cambiar el comportamiento	17
I.12 Títulos	18
I.13 Sangría etiqueta <blockquote> </blockquote>	19
I.14 Texto en Negritas	20
I.15 Texto en Cursiva	21
I.16 Texto Subrayado	22
I.17 Tamaño del Texto	22
I.18 Color del Texto	24
I.19 Cambiar del Tipo de Fuente	25
I.20 Alineación del Texto	26
I.21 Inserción de Imágenes	27
I.22 Líneas Horizontales	31
I.23 Listas	33
I.24 Hipervínculo	36
I.25 Frames	43
I.26 Multimedia	57
I.27 Creación de Tablas	62
I.28 Formularios	75
Capítulo II CSS	
II.1 Formas de Estructurar un CSS	94
II.2 Color de Fondo	98
II.3 Color de Títulos y Textos	98
II.4 Imagen de Fondo	99
II.5 Repetir Imagen de Fondo	100
II.6 Fondo de Imagen Fija a Desplazada	104
II.7 Ubicación de la Imagen de Fondo	108
II.8 Familias de Fuentes	112
II.9 Tamaño de la Fuente	112
II.10 Colocación de Sangrías	114
II.11 Alineación del Texto	115
II.12 Decoración del Texto	118
II.13 Enlaces	122

II.14 Visualización en Mayúsculas	127
II.15 Uso de Clases en CSS	127

Capítulo III JavaScript

III.1 Vincular Programas Javascript en HTML	132
III.2 Inserción de Código Javascript en un documento HTML	132
III.3 Objetos en Javascript	134
III.4 Propiedades	134
III.5 Métodos de Javascript	134
III.6 Estructura de un Programa Javascript	135
III.7 Definir Funciones	135
III.8 Variables y Constantes en Javascript	136
III.9 Operadores de Comparación	139
III.10 Operadores de Cálculo	140
III.11 Operadores de Incremento y Disminución	141
III.12 Operadores Lógicos	142
III.13 Prioridades entre Operadores	143
III.14 Estructuras de Control	143
III.15 Expresiones de Condición	143
III.16 Estructura IF	144
III.17 Ciclo For	145
III.18 Ciclo While	146
III.19 Sentencia do...while	148
III.20 Sentencia Switch	148
III.21 Comentarios	150
III.22 Utilización del Objeto Document	151
III.23 Utilización del Objeto Window	152
III.24 Movimiento del Título	153
III.25 Consultar los Valores de Entrada en Javascript	154

Capítulo IV HTML Dinámico

IV.1 Desactivar las Barras de Desplazamiento	157
IV.2 Trabajar con Eventos	157
IV.3 Vincular un Evento a un Procedimiento	157
IV.4 Barra de Estado Desplazable (scrollbar) a través del evento timer	158
IV.5 Barra de Estado Parpadeante	160
IV.6 Ajustar el Valor de un Atributo	161
IV.7 Posicionar Imagen por detrás del texto	163
IV.8 Definir una Imagen y un Texto de Fondo	164
IV.9 Textos Sombreados	165
IV.10 Textos con Efectos de Brillo	166
IV.11 Textos con Efectos de Sombra	167
IV.12 Filtro con Efecto de Olas	168
IV.13 Modificar el Estilo del Cursor	169

Capítulo V PHP y MySQL	
V.1 Programación con PHP	172
V.2 Separación de Instrucciones	173
V.3 Comentarios de PHP	174
V.4 Variables	174
V.5 Tipos de Datos	171
V.6 Caracteres Protegidos	176
V.7 Operadores de Comparación	177
V.8 Operadores Lógicos	178
V.9 Operadores de Asignación	178
V.10 Expresiones de Condición	179
V.11 Estructura IF	180
V.12 Condición IF...ELSEIF...ELSE	181
V.13 Sentencia Switch	181
V.14 Sentencia While	183
V.15 Sentencia Do While	183
V.16 Ciclo For	184
V.17 Funciones	185
V.18 Formularios desde PHP	186
V.19 MySQL	196
V.20 PHP y MySQL	201
V.21 Agrupando PHP y MySQL	203
Conclusiones	210
Anexo I Instalación de WAMP y MySQL	213
Anexo II Utilización de WAMP y MySQL	229
Anexo III Colores en Hexadecimal	246
Anexo IV Sitios Web más Destacados	250
Bibliografía	257
Webgrafía	257
Enlaces Sugeridos	257
Sitios de Alumnos Sobresalientes	258

ÍNDICE DE FIGURAS.

I.1 Visualización del Archivo.....	5
I.2 Visualización de la Página.....	6
I.3 Página con estructura de HTML.....	8
I.4 Actualización de la Página.....	8
I.5 Salto de Línea.....	9
I.6 Color de Fondo.....	11
I.7 Imagen de Fondo.....	12
I.8 Color de Fondo.....	13
I.9 Comentarios.....	14
I.10 Colocación de Marquesinas.....	16
I.11 Altura y Ancho de la Marquesina.....	16
I.12 Margen de Marquesinas.....	17
I.13 Títulos de Página.....	18
I.14 Sangría de la Página.....	19
I.15 Texto en Negritas.....	20
I.16 Texto en Cursiva.....	21
I.17 Texto en Cursiva.....	21
I.18 Texto Subrayado.....	22
I.19 Tamaño del Texto.....	23
I.20 Color del Texto.....	25
I.20 Tipo de Fuente.....	26
I.21 Alineación del Texto.....	27
I.22 Inserción de Imágenes.....	28
I.23 Alineación de Imágenes con Texto.....	29
I.24 Colocación de Texto Explicativo.....	30
I.25 Cambio del Tamaño de Imagen.....	30
I.26 Colocación del Borde.....	31
I.27 Colocación de Líneas Horizontales.....	32
I.28 Lista no Numerada.....	33
I.29 Lista Numerada.....	34
I.30 Lista de Glosario.....	36
I.31 Colocación de Hipervínculo a un buscador.....	37
I.32 Hipervínculo direccionado a un archivo.....	38
I.33 Descarga de Archivo.....	38
I.34 Guardar Archivo.....	39
I.35 Abrir Archivo.....	39
I.36 Hipervínculo a otra página.....	40
I.37 Hipervínculo a Correo Electrónico.....	41
I.38 Hipervínculo dentro de la misma página.....	42
I.39 División de Página por medio de Frames.....	45
I.40 División de Ventanas Verticalmente.....	46
I.41 División de Ventana Horizontalmente.....	47
I.42 Creación de dos o más frames horizontales.....	48
I.43 Creación de dos o más frames verticales.....	49
I.44 Hipervínculo dentro de un frame.....	49
I.45 Hipervínculo con Propiedades Predefinidas.....	51

I.46 Tamaño del Borde del Frame	54
I.47 Color del Borde del Frame	55
I.48 Imagen del Fondo del Frame	56
I.49 Imagen de Fondo del Frame con Etiquetas HTML	56
I.50 Página con Fondo Musical	58
I.51 Enlace a un Archivo Musical	59
I.52 Página con Fondo Musical y sus Propiedades Predefinidas	60
I.53 Integración de Video Externo	61
I.54 Creación de Tabla	63
I.55 Alineación de Tabla	65
I.56 Visualización de Tabla	66
I.57 Colocación del Título a la Tabla	67
I.58 Color en Tabla	68
I.59 Color en una Sola Fila de la Tabla	69
I.60 Color en un Sola Celda de la Tabla	70
I.61 Color en una Sola Celda de la Tabla	71
I.62 Uso del Atributo rules	72
I.63 Tabla con Imagen de Fondo	73
I.64 Imagen de Fondo para una Celda	74
I.65 Tabla con Imagen de Fondo para una Celda	75
I.66 Cuadro de Texto	77
I.67 Cuadro de Texto con Tamaño Específico	78
I.68 Cuadro de Texto con Número Específico de Caracteres	80
I.69 Cuadro con Petición de Contraseña	81
I.70 Cuadro de Texto Multilínea	82
I.71 Casillas de Verificación	83
I.72 Botones de Radio	84
I.73 Botón	85
I.74 Botón de Borrado	86
I.75 Cuadro de Lista Despegable	88
I.76 Ventana de Selección	89
I.77 Tabla dentro del Formulario	90
I.78 Tabla dentro del Formulario	92
II.1 Método en Línea	95
II.2 Método Interno	96
II.3 Método Externo	97
II.4 Color de Fondo	99
II.5 Imagen de Fondo	100
II.6 Repetir Imagen de Fondo Verticalmente	101
II.7 Repetir Imagen de Fondo Horizontalmente	102
II.8 Repetir Imagen de Fondo en toda la Página	103
II.9 No Repetir la Imagen de Fondo	104
II.10 Imagen de Fondo Repetida	106
II.11 Imagen de Fondo Fija	108
II.12 Ubicación de la Imagen de Fondo	109
II.13 Ubicación de Imagen de Fondo Centrada	110
II.14 Ubicación de la Imagen de Fondo a la Izquierda	111
II.15 Ubicación de Imagen de fondo a la derecha	112

II.16 Tamaño de la Fuente.....	113
II.17 Colocación de Sangrías	115
II.18 Alineación del Texto a la Derecha.....	116
II.19 Alineación del Texto a la Izquierda.....	117
II.20 Alineación del Texto Justificado.....	118
II.21 Decoración del Texto.....	119
II.22 Decoración del Texto.....	121
II.23 Texto Parpadeando.....	122
II.24 Enlaces.....	124
II.25 Enlace Activo.....	125
II.26 Enlace Decorado.....	126
II.27 Uso de Clases.....	130
III.1 Inserción de JavaScript	133
III.2 Uso de Funciones.....	136
III.3 Definición de Variables.....	138
III.4 Operadores de Comparación.....	140
III.5 Operadores de Cálculo.....	141
III.6 Estructura IF.....	145
III.7 Ciclo For	146
III.8 Ciclo While.....	147
III.9 Sentencia Do While.....	150
III.10 Sentencia Do While.....	150
III.11 Comentarios.....	151
III.12 Objeto Document.....	152
III.13 Objeto Window.....	153
III.14 Movimiento del Título.....	154
III.15 Valores de Entrada.....	154
IV.1 Vincular un Evento a un Procedimiento.....	158
IV.2 Barra de Estado Despegable.....	160
IV.3 Barra de Estado Parpadeante.....	161
IV.4 Ajuste de Valores.....	163
IV.5 Imagen Detrás del Texto.....	164
IV.6 Texto e Imagen de Fondo.....	165
IV.7 Textos Sombreados.....	166
IV.8 Textos con Efectos de Brillo.....	167
IV.9 Filtros con Efectos de Sombras.....	168
IV.10 Filtros con Efectos de Olas.....	169
IV.11 Modificación del Estilo del Cursor.....	170
V.1 Programación en PHP.....	173
V.2 Separación de Instrucciones.....	174
V.3 Utilización de Variables.....	176
V.4 Operadores de Asignación	179
V.5 Estructura IF.....	181
V.6 Estructura IF ELSEIF ELSE.....	181
V.7 Sentencia Switch.....	182
V.8 Sentencia While.....	183

V.9 Sentencia Do While	184
V.10 Ciclo For	184
V.11 Uso de Funciones	186
V.12 Uso de Formularios y PHP	187
V.13 Uso de Formularios y PHP	187
V.14 Uso de Caja de Texto	188
V.15 Uso de Casillas de Radio	189
V.16 Uso de Casillas Checkbox	190
V.17 Uso de Botones	190
V.18 Localizar un Archivo	191
V.19 Controles Ocultos	192
V.20 Introducción de Password	193
V.21 Casilla de Selección	194
V.22 Casilla de Múltiple Selección	195
V.23 Caja de Texto Multilínea	195
V.24 Agrupación de PHP y MySQL en un Catálogo de Productos	204
V.25 Inserción de Productos	205
V.26 Mensaje de Aceptación del Producto	206
V.26 Mensaje de Actualización del Producto	207
V.27 Mensaje de Actualización del Producto	207
V.28 Mensaje de Eliminación del Producto	208
Al.1 Contenido del comprimido mysql-5.0.13-rc-win32	213
Al.2 Comienzo de la Instalación	213
Al.3 Presentación de Instalación	213
Al.4 Selección de Instalación	214
Al.5 Proceso de Instalación	214
Al.6 Creación de Cuenta	214
Al.7 Finalización de MySQL	215
Al.8 Configuración de MySQL	215
Al.9 Tipo de Configuración	215
Al.10 Tipo de Servidor	216
Al.11 Selección de la Base de Datos	216
Al.12 Asignación de Usuarios para Utilizar el Software	216
Al.13 Asignación del Puerto a Utilizar	217
Al.14 Selección del Idioma del Software	217
Al.15 Incluir Consola de Comandos MySQL	217
Al.16 Colocación de Contraseña	218
Al.17 Autorización para Instalar las Propiedades del Software	218
Al.18 Finalización Correcta de la Instalación de Propiedades	219
Al.19 Contenido del Comprimido Mydqlcc-0.9.4-win32	219
Al.20 Instalación de MySQL	219
Al.21 Aceptar Licencia	220
Al.22 Términos y Condiciones	220
Al.23 Ubicación donde se instala el programa	221
Al.24 Instalación de los Controles de MySQL	221
Al.25 Confirmación de Instalación	221
Al.26 Proceso de Instalación	222
Al.27 Fin de Instalación	222

Al.28 Inicio de la Instalación de WAMP 5	223
Al.29 Aceptación de Licencia	223
Al.30 Ubicación de Instalación del Programa	224
Al.31 Tipo de Servidor que Utilizará	224
Al.32 Inicio Automático de WAMP 5	224
Al.33 Instalación de WAMP 5	225
Al.34 Proceso de Instalación	225
Al.35 Selección de Carpeta para Guardar los Programas Elaborados	226
Al.36 Directorio Raíz de Ejecución de Programas	226
Al.37 Selección del Explorador a Utilizar	227
Al.38 Fin de la Instalación	227
All.1 Captura de Programa	229
All.2 Guardar con Extensión .php	229
All.3 Directorio de WAMP	230
All.4 Carpeta de Alojamiento de Programas	230
All.5 Ubicación de Localhost dentro del Directorio	231
All.6 Servidor de PHP	231
All.7 Dirección de Localhost	232
All.8 Colocación del Nombre de la Página al Compilar	232
All.9 Visualización de la Página	232
All.10 Icono de MySQL	232
All.11 Vista de Aplicación	233
All.12 Registro de Servidor	233
All.13 Llenado de Ventana	234
All.14 Opción de Introducción de Datos Correctos	234
All.15 Conexión Exitosa	235
All.16 Servidor Instalado	235
All.17 Conexión del Servidor	236
All.18 Vista del Servidor en Funcionamiento	236
All.19 Crear Base de Datos	237
All.20 Nombrar Base de Datos	237
All.21 Inserción de Base de Datos	238
All.22 Conexión de la Base de Datos	238
All.23 Base de Datos Conectada	239
All.24 Crear Tabla de Datos	239
All.25 Inserción de Atributos a la Tabla	240
All.26 Salvar Tabla	240
All.27 Asignar Nombre a la Tabla	241
All.28 Visualización de la Tabla dentro del Servidor	241
All.29 Editor de Comandos de SQL	241
All.30 Vista del Editor de Comandos de SQL	242
All.31 Inserción de Código MySQL	242
All.32 Ejecutar Código	242
All.33 Selección de Código a Ejecutar	243
All.34 Mensaje de Aceptación de Comandos	243
All.35 Vista Completa del Compilador	243
All.36 Mensaje de Introducción Errónea de Comandos	244

AIV.1 Sitio de Música	250
AIV.2 Sitio de Grupo Musical Nirvana	250
AIV.3 Sitio de Consolas de Juegos de Videos	251
AIV.4 Sitio de Anime Candy Candy	251
AIV.5 Sitio de Harry Potter	252
AIV.6 Sitio de Automóviles	252
AIV.7 Sitio de Tome Raider	253
AIV.8 Sitio de One Peace	253
AIV.9 Sitio de Música DJ	254
AIV.10 Sitio de Ángeles	254
AIV.11 Sitio de Vacaciones en el Cervantino	255
AIV.12 Sitio de Flores	255
AIV.13 Sitio del Sistema Solar	256
AIV.14 Sitio Oficial del Curso de Desarrollo Web	256

ÍNDICE.

Introducción	1
Capítulo I HTML	Pág.
I.1 Estructura del documento	4
I.2 Estructura de un archivo HTML	5
I.3 Salto de Línea	7
I.4 Párrafos	8
I.5 Color de Fondo	9
I.6 Imagen de Fondo	10
I.7 Color de Texto	11
I.8 Comentarios	13
I.9 Marquesinas	14
I.10 Cómo cambiar la apariencia	15
I.11 Cómo cambiar el comportamiento	16
I.12 Títulos	17
I.13 Sangría etiqueta <code><blockquote></code> <code></blockquote></code>	18
I.14 Texto en Negritas	19
I.15 Texto en Cursiva	20
I.16 Texto Subrayado	21
I.17 Tamaño del Texto	21
I.18 Color del Texto	23
I.19 Cambiar del Tipo de Fuente	24
I.20 Alineación del Texto	25
I.21 Inserción de Imágenes	26
I.22 Líneas Horizontales	30
I.23 Listas	32
I.24 Hipervínculo	35
I.25 Frames	42
I.26 Multimedia	54
I.27 Creación de Tablas	61
I.28 Formularios	74
Capítulo II CSS	
II.1 Formas de Estructurar un CSS	92
II.2 Color de Fondo	96
II.3 Color de Títulos y Textos	96
II.4 Imagen de Fondo	97
II.5 Repetir Imagen de Fondo	98
II.6 Fondo de Imagen Fija a Desplazada	102
II.7 Ubicación de la Imagen de Fondo	106
II.8 Familias de Fuentes	110
II.9 Tamaño de la Fuente	110
II.10 Colocación de Sangrías	112
II.11 Alineación del Texto	113
II.12 Decoración del Texto	116
II.13 Enlaces	120
II.14 Visualización en Mayúsculas	125
II.15 Uso de Clases en CSS	125

Capítulo III JavaScript

III.1 Vincular Programas Javascript en HTML	129
III.2 Inserción de Código Javascript en un documento HTML	129
III.3 Objetos en Javascript	131
III.4 Propiedades	131
III.5 Métodos de Javascript	131
III.6 Estructura de un Programa Javascript	132
III.7 Definir Funciones	132
III.8 Variables y Constantes en Javascript	133
III.9 Operadores de Comparación	136
III.10 Operadores de Cálculo	137
III.11 Operadores de Incremento y Disminución	138
III.12 Operadores Lógicos	139
III.13 Prioridades entre Operadores	140
III.14 Estructuras de Control	140
III.15 Expresiones de Condición	140
III.16 Estructura IF	141
III.17 Ciclo For	142
III.18 Ciclo While	143
III.19 Sentencia do....while	145
III.20 Sentencia Switch	145
III.21 Comentarios	147
III.22 Utilización del Objeto Document	148
III.23 Utilización del Objeto Window	149
III.24 Movimiento del Título	150
III.25 Consultar los Valores de Entrada en Javascript	151

Capítulo IV HTML Dinámico

IV.1 Desactivar las Barras de Desplazamiento	153
IV.2 Trabajar con Eventos	153
IV.3 Vincular un Evento a un Procedimiento	153
IV.4 Barra de Estado Desplazable (scrollbar) a través del evento timer	154
IV.5 Barra de Estado Parpadeante	156
IV.6 Ajustar el Valor de un Atributo	157
IV.7 Posicionar Imagen por detrás del texto	159
IV.8 Definir una Imagen y un Texto de Fondo	160
IV.9 Textos Sombreados	161
IV.10 Textos con Efectos de Brillo	162
IV.11 Textos con Efectos de Sombra	163
IV.12 Filtro con Efecto de Olas	164
IV.13 Modificar el Estilo del Cursor	165

Capítulo V PHP y MySQL

V.1 Programación con PHP	167
V.2 Separación de Instrucciones	168
V.3 Comentarios de PHP	169
V.4 Variables	169
V.5 Tipos de Datos	171
V.6 Caracteres Protegidos	172
V.7 Operadores de Comparación	173

V.8 Operadores Lógicos	173
V.9 Operadores de Asignación	174
V.10 Expresiones de Condición	175
V.11 Estructura IF	176
V.12 Condición IF...ELSEIF....ELSE	176
V.13 Sentencia Switch	178
V.14 Sentencia While	178
V.15 Sentencia Do While	178
V.16 Ciclo For	179
V.17 Funciones	180
V.18 Formularios desde PHP	181
V.19 MySQL	191
V.20 PHP y MySQL	196
V.21 Agrupando PHP y MySQL	198
Conclusiones	204
Anexo I Instalación de WAMP y MySQL	206
Anexo II Utilización de WAMP y MySQL	221
Anexo III Colores en Hexadecimal	237
Anexo IV Sitios Web más Destacados	240
Bibliografía	247
Webgrafía	247
Enlaces Sugeridos	247
Sitios de Alumnos Sobresalientes	248

Introducción.

En la actualidad la Web ha cambiado nuestra forma de comunicación. Hoy en día es muy difícil, ver a una persona, que no acceda a internet, por lo que se ha convertido en el medio de comunicación más eficiente, además de su velocidad en transmitir la información, así como también una muy buena herramienta para el comercio, en la cual se ofrecen productos, y estar seguros de que millones de personas los verán.

Muchas de las personas que acceden a las páginas que nos presenta internet no tienen ninguna noción de cómo se realizan este tipo de páginas, y no tienen idea de que existen programadores y diseñadores gráficos que realizan todo este trabajo. Uno de los objetivos de este trabajo, es hacer que el lector pueda realizar sus propias páginas Web, sin la necesidad de solicitar estos servicios.

En el primer capítulo se explican las bases mínimas que se deben tener a la hora de realizar una página Web, por medio del lenguaje HTML¹, este capítulo consta de 26 apartados. El primer y segundo apartado sirven de introducción al lector en la realización de su primera página. Al terminar los apartados del 3 al 4, el lector tendrá clara la estructura en la realización de páginas Web, sabrá cómo colocar saltos de línea y la utilización de párrafos. En el apartado 5 y 6, el lector aprenderá la forma de colocar diferentes tipos de fondo a la página. En los apartados de 7 al 18, se explican los diferentes formatos que puede tener el texto, así como la animación que le podemos dar algunos de ellos son: cambio de fuente, color del texto, tamaño de la fuente, marquesinas, títulos, sangrías. En apartado 19 y 20 se explica la forma de colocar una imagen y una línea vertical en la página. En el apartado 21 se explica cómo crear listas. En el apartado 22 se explican los hipervínculos o enlaces, los cuales pueden direccionar a otras páginas. En el apartado 23 el lector aprenderá a crear y manejar frames, esto es páginas web las cuales muestran simultáneamente varias páginas Web. En el apartado 24 se explica cómo incrustar objetos multimedia a la página, como: Sonido, video, archivos a descargar. En el apartado 25 se explica la creación de tablas con sus diferentes formatos. Este capítulo cierra con el apartado 26, en el cual se explica la creación de formularios, en páginas Web, esto va más aplicado a la realización de algún formato específico, que debe llevar la página, en cuanto a la presentación de la misma, algunos de los formularios que se explican son: colocación de botones, cuadros de texto, casillas de selección, etc.

En el segundo capítulo se presenta el concepto de hojas de estilo (CSS²), las cuales se emplean para dos cuestiones: separar las propiedades de decoración de la página (color, fuente de letra, alineación de texto, etc.) de la información que contenga, y utilizarlos como base en HTML Dinámico. Este capítulo consta de 15 apartados. En el apartado número 1, presenta al lector el concepto de hoja de estilo, indicando las ventajas de su uso, así como las posibilidades que ofrece. Además se muestra la forma en que se debe manejar una hoja de estilo, con los tres diferentes métodos para crearlas y sus reglas. En el apartado 2 se explica cómo debemos manejar el color de fondo de una hoja de estilo y

¹ HTML: Hypertext Markup Language(Lenguaje de Marcas de Texto)

² CSS: Cascading Style Sheets(Hojas de Estilo en Cascada)

las características generales que deberán manejar dentro de CSS y HTML. En el apartado 3, se explica el formato que deben tener los títulos. En los apartados del 4 al 7 se explican los diferentes formatos, que se tiene para colocar una imagen de fondo, como: su ubicación dentro de la página, si la imagen permanece fija o con movimiento o si la imagen se repite dentro de la página varias veces o solo una. En los apartados del 8 al 12 se explican las diferentes propiedades que podemos colocarle al texto desde el tamaño de la fuente, colocación de sangrías, alineación del texto o decoración del texto. En el apartado 13 se explica cómo colocar los enlaces dentro de la hoja de estilo, así como sus diferentes presentaciones que puede tener dentro de la página. En el apartado 14 podemos ver cómo se puede visualizar todo el texto en mayúsculas. Este capítulo cierra con el apartado 15, donde se explica el empleo de clases, las cuales pueden ser de mucha ayuda, a la hora de hacer páginas Web, con muchos formatos diferentes.

En el tercer capítulo se presenta al lector el lenguaje Javascript, desde los conceptos más básicos hasta el uso de objetos; así mismo explica la forma de utilizar este lenguaje en las páginas Web mediante el uso del modelo de objetos del navegador y el empleo de eventos. En el apartado 1 se explica la forma de introducir el código de Javascript dentro de la página de HTML. En los apartados del 2 al 5 se explican los objetos y propiedades y métodos que tiene este lenguaje. En los apartados del 6 al 8 se explica la estructura del programa, definir funciones, variables, constantes, caracteres especiales para cadenas. En los apartados del 9 al 13 se explican los operandos del comparación, operadores de cálculo, operadores de incremento y disminución, operadores lógicos, y las prioridades entre los operadores, los cuales sirven de mucha ayuda a la hora de tomar una decisión o realizar un cálculo. En los apartados del 14 al 20 se explican las estructuras de control y condición con las cuales se pueden tomar algunas decisiones, dentro de las expresiones de control se encuentran la estructura if, el ciclo for, ciclo while, sentencia do while, sentencia whitch, las cuales pueden ayudar de manera optima a la hora de realizar un programa. En el apartado 21 se explica la forma de colocar comentarios dentro del lenguaje Javascript, para colocar algunas anotaciones que puedan servir de mucha ayuda en un futuro. En los apartados del 22 al 23, se explican la utilización de algunos objetos que maneja Javascript dentro del explorador. En el apartado 24 el lector podrá colocar el título del explorador con efectos de animación. Este capítulo cierra con el apartado 25 con la consulta de valores introducidos dentro de Javascript.

El capítulo IV da una breve introducción al lenguaje de HTML dinámico, explicando cómo podemos determinar la visualización de un documento, cómo crear y manejar capas, o la forma en que se manipulan de forma dinámica las características de la página Web. Con este capítulo el lector será capaz de crear páginas Web, que no solamente sean correctas, sino que visualmente sean muy atractivas, cargadas de una gran interactividad e independientes del navegador utilizado. Este capítulo consta de 15 apartados, en los que se muestra cómo manejar diferentes eventos que tiene el explorador, con la ayuda de HTML, CSS y Javascript, agrupándoles, de manera justificada para lograr una página muy atractiva.

El capítulo V da una introducción al lenguaje PHP³ el cual es considerado un lenguaje de programación de estilo clásico, es decir un lenguaje de programación con variables, sentencias, condicionales, ciclos y funciones. No es un lenguaje de marcado como lo puede ser HTML, XML⁴ ó WML⁵. Está más cercano a ser Javascript o lenguaje C. Este capítulo consta de 21 apartados los cuales muestran la forma de programar dentro de PHP; así como una breve introducción a lo que es la programación de base de datos dentro de MYSQL, para después fusionarlos en el último apartado, y que trabajen de manera conjunto dentro de un sitio Web, o página la cual necesite, contener dentro de ella una base de datos. En este capítulo se mencionan dos anexos en los cuales se explica la instalación y el uso de los programas.

Se espera que conforme siga avanzando el desarrollo de tecnología, el área de tecnologías Web, siga creciendo cada vez más, con el fin de facilitar algunos de los problemas que se tienen hoy en día en los diferentes dispositivos.

Con el desarrollo de este trabajo se pretende:

- Ⓢ Que el lector pueda adentrarse más a las tecnologías Web, desde los principios básicos, hasta una breve introducción de tecnologías Web avanzadas.
- Ⓢ Dar a conocer algunas de las herramientas básicas, que se deben comprender, para poder manejar con mayor facilidad las nuevas tecnologías Web, que se están utilizando actualmente en los países europeos, las cuales debemos tratar de traer a nuestro país.
- Ⓢ Aplicar los conocimientos adquiridos durante la carrera, en los problemas que se nos presentan a la hora de realizar un sitio o página Web.

³ PHP : Preprocessed Hypertext Pages (Páginas de Hipertexto Procesadas)

⁴ XML: Extensible Markup Language(Lenguaje de Marcas Extendidas)

⁵ WML: Wireless Markup Language



Capítulo I

HTML

Capítulo I HTML

El lenguaje HTML es un lenguaje interpretativo, en el cual un explorador, que comience, a leer una página solo nos mostrará en pantalla, lo que nosotros indiquemos, hay que mencionar, que no es como los lenguajes de compilación, ya que estos nos muestran, el error antes, de ejecutar el programa. HTML, no hace esto, solo ejecuta las instrucciones que conoce, y si tenemos algún error a la hora de colocar algún comando, solo nos daremos cuenta, durante la ejecución del código.

Una ventaja de codificar HTML desde un editor de texto es que podemos cambiar el código tantas veces queramos, sin la ayuda de programas diseñados especialmente para ello.

Ante todo hay que mencionar, que algunos de los problemas que surgen dentro de la Web, es que si una persona quiere consultar alguna de las páginas que existen, puede tener problemas al querer entrar a este sitio, debido a que algunos programas cargan mucho código, o la máquina en donde se encuentra el usuario no tiene las características necesarias para visualizarlo, un ejemplo de esto es Flash, el cual cuando se abre una página hecha desde este programa, puede llegar a tardar en visualizarse completamente unos minutos, lo cual puede desesperar a nuestro visitante, y hacer que cambie a otra página la cual pueda visualizar mucho más rápido.

Esta es la ventaja que tiene HTML, ya que es un lenguaje, el cual pueden interpretar las máquinas sin ningún problema, solo necesitamos, un explorador, y un editor de texto sencillo.

I.1 Estructura del documento

Cuando nosotros tenemos información la cual deseamos que varias personas puedan, leer debemos colocarla, dentro de un editor de texto y guardarla con el nombre y la extensión .txt

Cuando tengamos el archivo debemos localizarlo, y abrirlo directamente desde el explorador, y ver el sitio.

A continuación en la figura I.1 se ilustra la visualización del archivo.

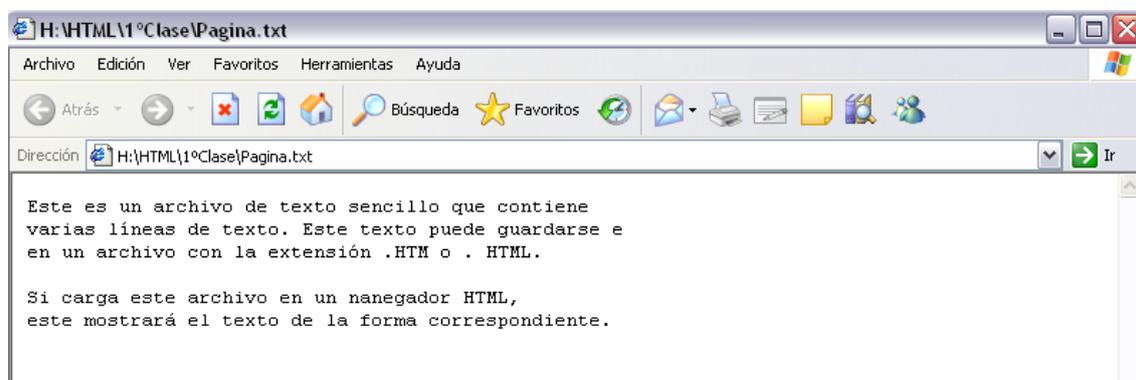


Figura I.1 Visualización del Archivo.

Como podemos observar el texto que colocamos esta en su formato original, pero esta extensión no es que la se visualiza dentro de lo servidores de Web, la extensión correcta con la cual debemos guardar todas nuestra páginas Web es “.html ó .htm”.

Para que nosotros observemos cómo es que se encuentra mi página la podemos ver, ya que si guardamos bien el archivo nos debe aparecer de la siguiente manera:



Si no aparece de esta manera, se tendrá que volver a guardar hasta obtener un archivo con el icono de Explorer.

Una vez que tenemos este archivo, damos doble clic, para ver la página.

A continuación en la figura I.2 se ilustra la visualización de la página.



I.2 Visualización de la Página.

Una vez que colocamos la extensión correcta, algunas de las indicaciones, que le habíamos dado a nuestro texto no se han respetado, esto se debe, a la estructura gramatical que tiene HTML, ya que si se encuentra a la hora de leer el código un enter, no le hará caso, debido a que HTML, tiene su etiqueta propia para hacer esta indicación.

I.2 Estructura de un archivo HTML

En HTML, a todas las instrucciones no se les conoce con el nombre de comando, sino por etiquetas o tags, de aquí en adelante nos referiremos a todas las instrucciones de esta manera. HTML, no es sensible a las mayúsculas y minúsculas, ambas las puede interpretar, a excepción de algunas etiquetas, las cuales se mencionarán, que si tienen sus restricciones a la hora de escribirlas.

Cada una de las etiquetas que tiene HTML, se podrá reconocer por los signos de mayor y menor que un ejemplo de ello es: `
`, como podemos observar, se encuentra dentro de los signos la instrucción de esta manera, nosotros podemos ubicar cual es un comando y que es información dentro de nuestra página Web.

Hay que recordar que un documento HTML, se encuentra conformado por texto.

La estructura básica de un documento de HTML en blanco es:

```
<html>
<head>
<title>      </title>
</head>
<body>

</body>
</html>
```

Esta es la estructura y el orden correcto que deben llevar todas nuestras páginas Web.

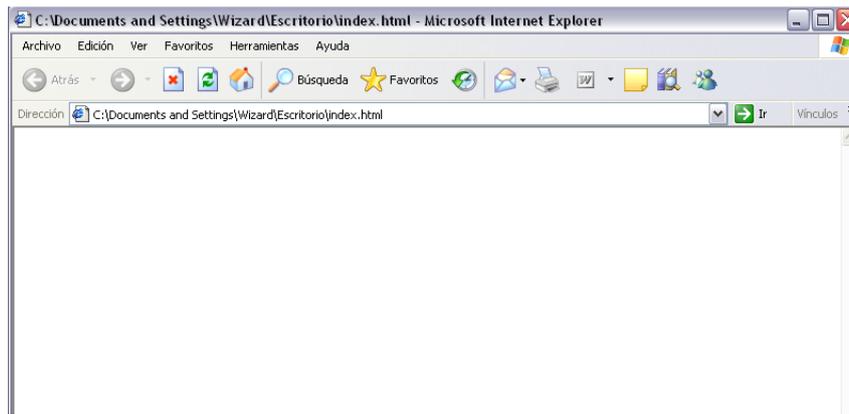


Ahora podemos visualizar varias de las etiquetas que tiene HTML estas son:

1. Estas etiquetas son las que abren y cierran nuestra página Web, todo lo que aparezca dentro de estas etiquetas lo podremos visualizar dentro de nuestro explorador. `<html>` `</html>`, este tipo de etiquetas y algunas otras más se deben cerrar, y se cierran escribiendo nuevamente la etiqueta pero anteponiendo a la palabra html el símbolo "/" para cerrarla.
2. La etiqueta `<head>` `</head>` son las cabeceras de la página Web, hay que recordar que estas se deben cerrar.
3. Etiqueta `<title>` con ella podemos colocar el título a nuestra página en el encabezado del explorador, y desde un principio indicarle al usuario, el tema del cual trata la página, y cerrar la etiqueta una vez colocado el título adecuado.
4. Etiqueta `<body>`, esta es la etiqueta principal de páginas, ya todo lo que este dentro de ella, se podrá, visualizar en la página con los diferentes formatos o decoración que nosotros le demos a la página, de aquí en adelante todas las etiquetas de veamos, deben ir dentro de body, para su buen funcionamiento, de igual manera ésta debe cerrarse.

Si nosotros guardamos este archivo podemos visualizar el explorador de la siguiente manera:

A continuación en la figura I.3 se ilustra la página con estructura HTML.



I.3 Página con estructura HTML

Para colocar el título en el encabezado de la página se escribe:

```
<html>
<head>
<title>Mi primera página</title>
</head>
<body>

</body>
</html>
```

Una vez que colocamos el título, debemos guardar este documento y si tenemos el explorador abierto, podemos dar F5 para actualizar o desde el explorador el símbolo de actualizar . Una vez que actualizamos observamos los cambios efectuados. A continuación en la figura I.4.



I.4 Actualización de la Página.

I.3 Salto de Línea

Como vimos anteriormente HTML, no respeta los saltos de línea, debido a que tiene una etiqueta especial la cual es `
`, esta etiqueta no necesita cerrarse, ya que solo se efectúa una vez, y se pueden poner tantas `
` como sean necesarias, seguidas una de otras, o distribuidas por todo el código de la página.

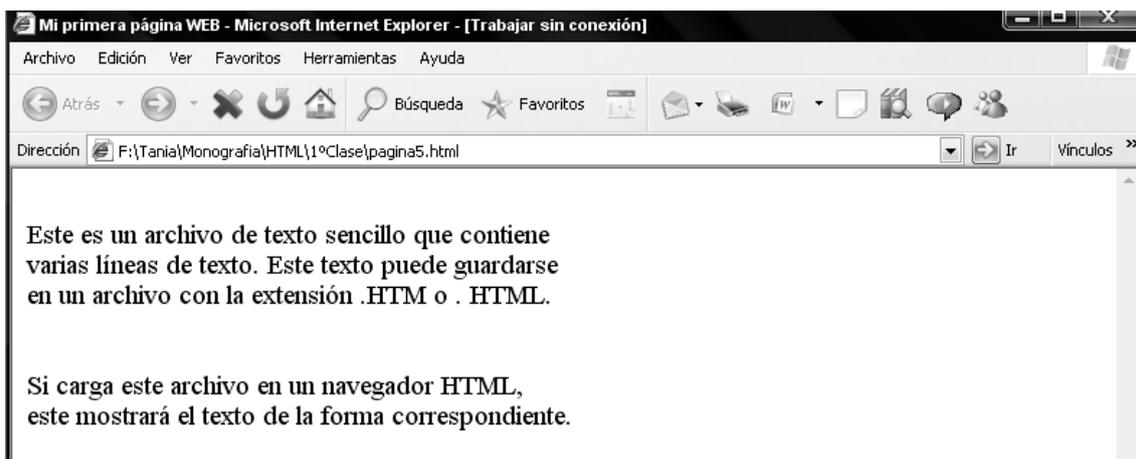
Utilizando el ejemplo anterior agregaremos la etiqueta para darle los espacios correspondientes como se muestra a continuación:

```

<HTML>
<HEAD>
<TITLE> Mi primera p&acute;gina WEB</TITLE>
</HEAD>
<BODY background="I:\HTML\2ºClase\Fondos\hellsing.jpg">
<br>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse e
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</BODY>
</HTML>

```

A continuación en la figura I.5 se ilustra el Salto de Línea.



I.5 Salto de Línea

Aquí podemos ver como HTML comienza a dar el formato que nosotros queremos que tenga nuestra página.

I.4 Párrafos

Para agrupar un texto en forma de párrafo, dejando al navegador ajustar el texto, al tamaño de la venta existe la etiqueta <p> y delimitar hasta donde queremos que se conforme el párrafo con la etiqueta de cierre </p>.

Por ejemplo:

```

<HTML>
<HEAD>
<TITLE> Mi primera p&acute;gina WEB</TITLE>
</HEAD>
<BODY background="I:\HTML\2ºClase\Fondos\hellsing.jpg">

```

```
<p>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse e
<br>
en un archivo con la extensión .HTM o . HTML.
</p>
<br>
<br>
<p>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</p>
</BODY>
</HTML>
```

I.5 Color de Fondo

Para colocar un color de fondo a la página debemos tener en cuenta, que el color deberá la atención del visitador de la página, pero sobre todo que no afecte la vista y que permita ver el contenido de la página.

El comando utilizado para la colocación es `<bgcolor>` el cual va acompañado del color en hexadecimal o escrito en inglés, de esta manera puede quedar como:

```
bgcolor = "FFD700"
```

ó

```
bgcolor = "yellow"
```

No importando cual se use de las dos es importante que se ponga entre comillas dobles, ya que es la forma en como el editor de texto lo reconocerá.

Este comando debe estar colocado dentro del `body` para que haga los cambios indicados como se muestra a continuación:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
</HEAD>
<BODY bgcolor="FFD700">
<br>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse e
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
```

</BODY>
</HTML>

A continuación en la figura I.6 se ilustra el color de fondo



I.6 Color de Fondo

I.6 Imagen de Fondo

Si en vez de colocar un color de fondo en la página podemos una imagen como fondo y así poder personalizar mucho más la página. Ya que si se esta realizando o diseñando la página para una empresa, es muy cotidiano, que estas quieran su logotipo como fondo.

Para colocar la imagen de fondo es con el comando <background>, la estructura es:

background = “dirección de la imagen = nombre de la imagen. Extensión”

Ya aplicado sería:

background= “H:\HTML\1°Clase\Fondos\sprincess.jpg”

En cuanto a las extensiones de imágenes, HTML soporta las extensiones JPG¹, GIF², BMP³, PNG⁴. Es importante colocar la extensión de la imagen de lo contrario no la podrá visualizar, además de la dirección donde localizará la imagen.

También podemos hacer que la imagen se busque de manera automatiza, dejando que el explorador se encargue de buscarla sola, esta abreviatura aplica para todos los comandos que necesiten de una dirección, la estructura es de la siguiente manera:

background= “./sprincess.jpg”

Este comando se coloca frente al body, en vez de colocar bgcolor, la estructura completa se muestra a continuación;

¹ JPEG: Join Photograph Expert Group (Union de Grupo de Expertos Fotografico)

² GIF: Graphics Interchange Format (Formato de intercambio de gráfcos)

³ BMP: Mapa de Bits

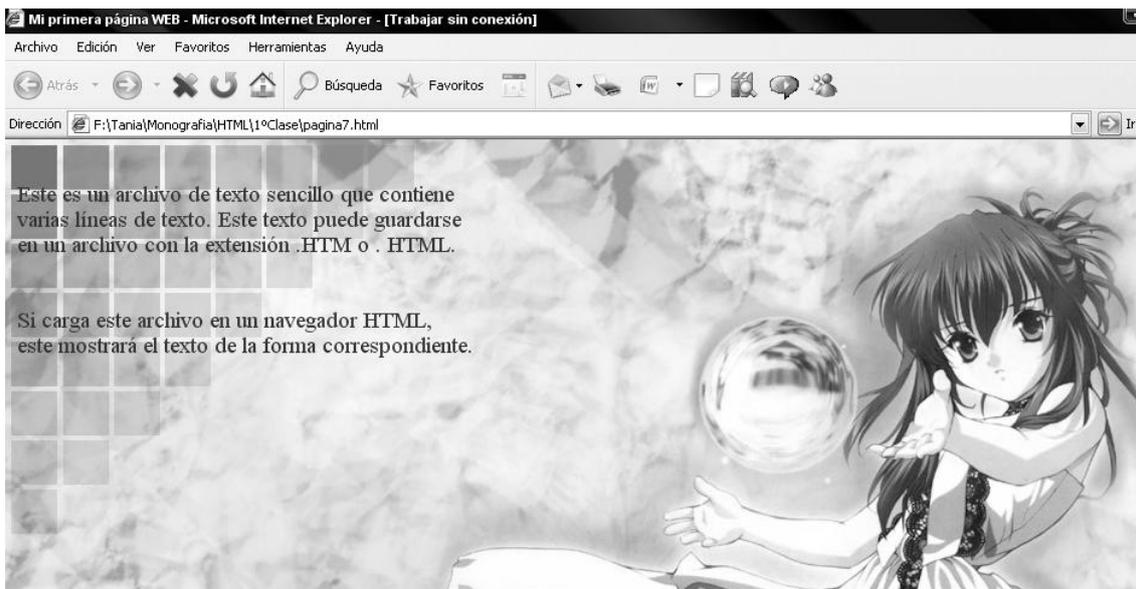
⁴ PNG: Portable Network Graphics

```

<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
</HEAD>
<BODY background= "H:\HTML\1ºClase\Fondos\sprincess.jpg">
<br>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</BODY>
</HTML>

```

A continuación en la figura I.7 se ilustra una imagen de fondo.



I.7 Imagen de Fondo

I.7 Color del Texto

Por default el explorador muestra las letras en color negro, pero si queremos cambiar el color al texto lo podemos hacer por medio de la etiqueta `text="color"`. Si utilizamos esta etiqueta todo el texto que se encuentre dentro de la páginas se tornará del color indicado.

Su sintaxis es:

🚦 **Forma Hexadecimal**

text= "FF00FF"

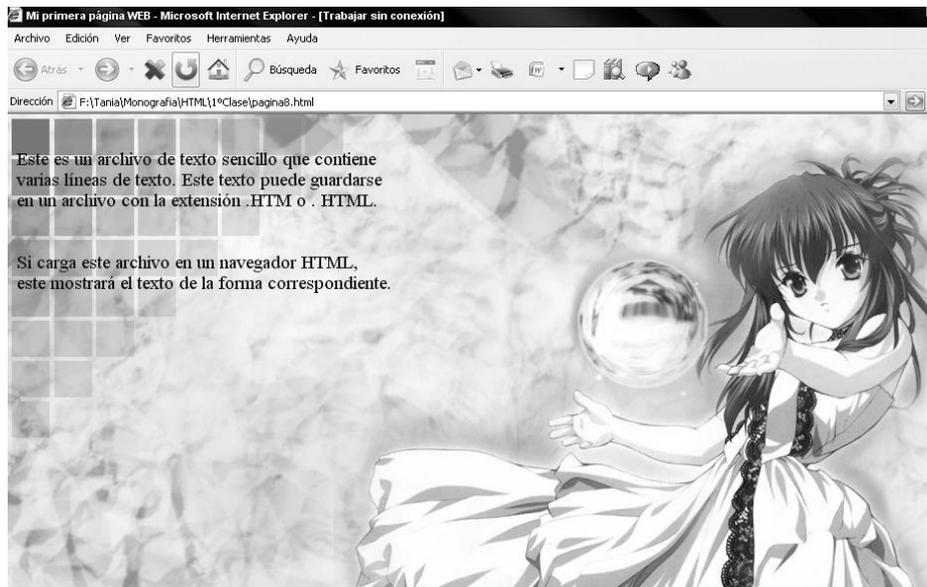
Forma de Nombre

text= "red"

Siempre colocado dentro del body también, cabe mencionar que pueden estar colocados ambos comandos tanto el de color de fondo o la imagen de fondo con el comando de color de texto como se muestra a continuación:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
</HEAD>
<BODY background= "H:\HTML\1°Clase\Fondos\sprincess.jpg" text="red">
<br>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</BODY>
</HTML>
```

A continuación en la figura I.8 se ilustra el color del Texto



I.8 Color del Texto

I.8 Comentarios

Para aplicar comentarios al código html, y así saber qué es lo que estamos haciendo es de la siguiente manera:

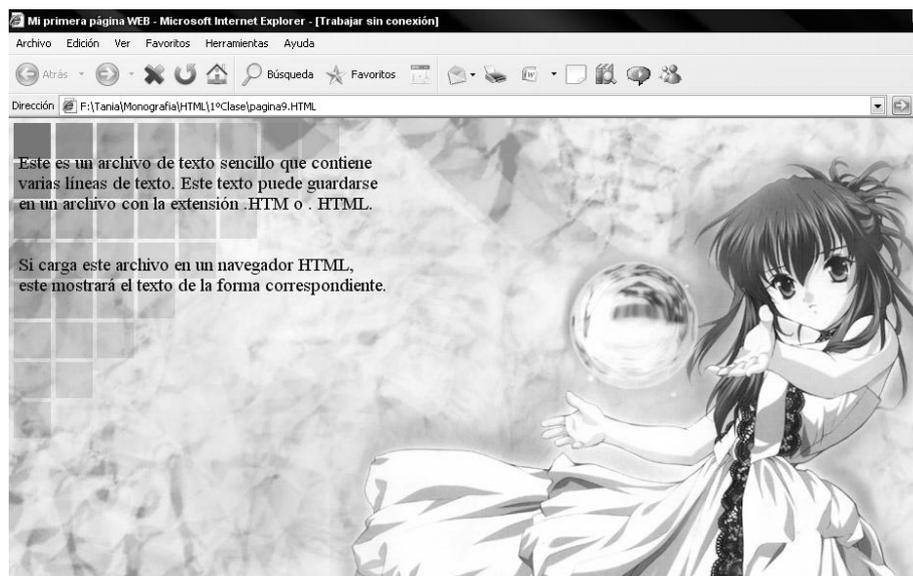
<! -Comentario-- >

Todo lo que se encuentre dentro de la etiqueta no será visible en la página, si se encuentra fuera, se verá en esta.

Por Ejemplo:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
<!--PONER TITULO A LA PAGINA-->
</HEAD>
<BODY background="H:\HTML\1ºClase\Fondos\sprincess.jpg" text="blue">
<!-- Poner imagen de fondo-->
<br><!--salto de linea-->
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</BODY>
</HTML>
```

A continuación en la figura I.9 se ilustran los comentarios.



I.9 Comentarios

I.9 Marquesinas

Una marquesina consiste en una línea de texto que se desplaza por la pantalla de izquierda a derecha, de derecha a izquierda o en ambas direcciones alternativamente.

MARQUEE nos permitirá crear este tipo de marquesina con gran facilidad y rapidez.

El funcionamiento de esta etiqueta es muy sencillo, consta de una instrucción de inicio y otra de fin, entre ellas debemos escribir el texto que queremos que se desplace. Por ejemplo:

```
<MARQUEE>
Mi primera Página WEB
</MARQUEE>
```

Introducido dentro del código se ve de la siguiente manera:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
<!--PONER TITULO A LA PAGINA-->
</HEAD>
<BODY background="H:\HTML\1ºClase\Fondos\sprincess.jpg" text="blue">
<!-- Poner imagen de fondo-->
<br><!--salto de linea-->
<marquee>Mi Primera Página WEB</marquee>
<br>
<br>
<br>
<br>
<br>
Este es un archivo de texto sencillo que contiene
<br>
varias líneas de texto. Este texto puede guardarse
<br>
en un archivo con la extensión .HTM o . HTML.
<br>
<br>
<br>
Si carga este archivo en un navegador HTML,
<br>
este mostrará el texto de la forma correspondiente.
</BODY>
</HTML>
```

Si visualizamos este código con el explorador veremos aparecer el texto “Mi Primera Página WEB” por la derecha de la pantalla y desplazarse hacia la izquierda hasta llegar al borde izquierdo de la pantalla.

Entonces volvería a aparecer de nuevo por la derecha y así indefinidamente.

Una vista en el explorador es muy difícil de mostrar, ya que como se encuentra en movimiento no se puede capturar una sola imagen para poder ejemplificarla. Pero

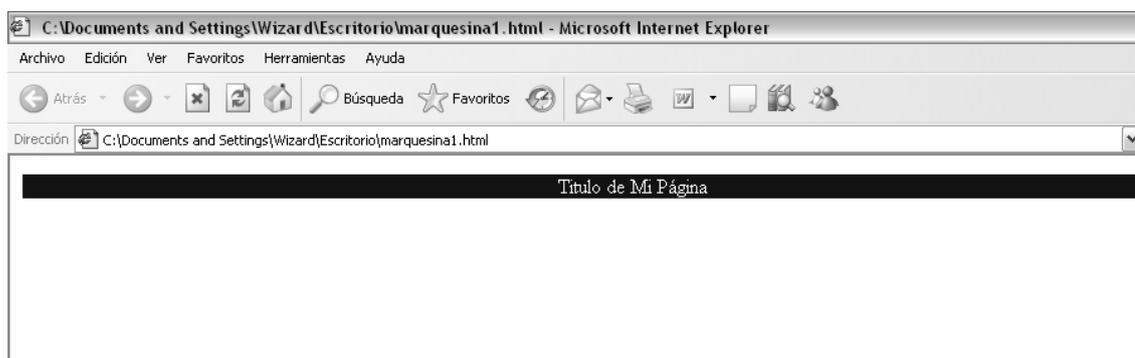
dentro del disco que contiene este trabajo, en la carpeta HTML, Clase 1, Ejercicio 10, se muestra el ejemplo de manera visual.

1.10 Cómo cambiar la apariencia

La etiqueta MARQUEE contiene una serie de atributos que nos permite cambiar su apariencia, algunos de ellos son:

- ✚ **BGCOLOR:** Podemos especificar un color en código RGB (#RRGGBB) o por nombre en inglés (blue, red, etc.) con esto podemos cambiar el color de fondo de la marquesina.

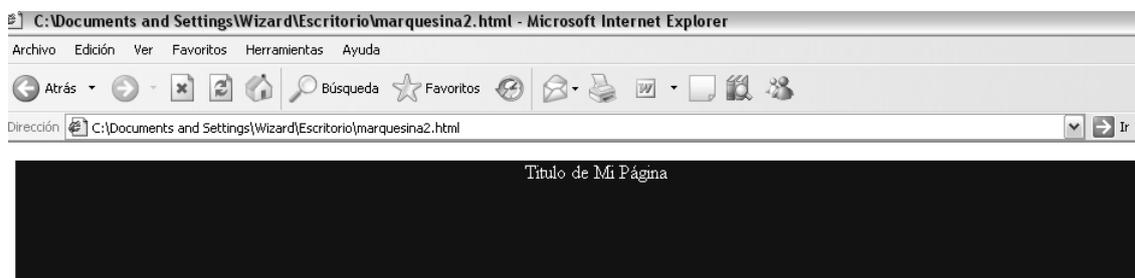
A continuación en la figura I.10 colocación de marquesinas.



I.10 Colocación de Marquesinas

- ✚ **HEIGHT y WIDTH:** Estas etiquetas determinan la altura y el ancho de la marquesina. Ambas etiquetas pueden tomar un valor en píxeles o bien como porcentaje de la ventana del navegador, esto es que se adapte al mismo navegador. Por ejemplo HEIGHT="50%" provocará que la marquesina tenga una altura mitad de la de la ventana del navegador. Estos atributos no afectan al tamaño del texto.

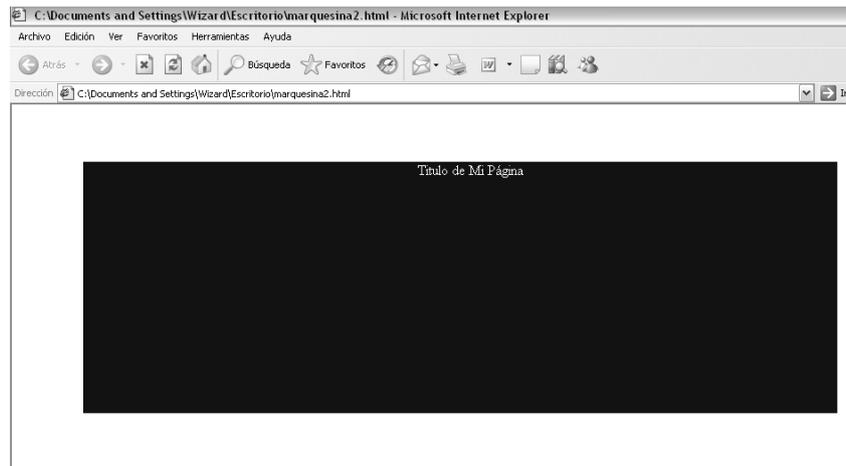
A continuación en la figura I.11 se ilustra la altura y ancho de la marquesina.



I.11 Altura y Ancho de la Marquesina

- ✚ **HSPACE y VSPACE:** Determinan los márgenes entre los bordes de la marquesina y el texto u otros elementos de la página. HSPACE determina el margen a cualquiera de los lados y VSPACE los márgenes superior e inferior.

A continuación en la figura I.12 se ilustra el margen de la marquesina.



I.12 Margen de la Marquesina.

- ✚ **ALIGN:** Este atributo controla el alineamiento vertical del texto que bordea la marquesina. Puede tomar cinco valores diferentes: `ALIGN="top"` (arriba), `ALIGN="middle"` (centro), `ALIGN="bottom"` (abajo), `ALIGN="left"` (izquierda) `ALIGN="right"` (derecha). Este atributo no modifica la posición del texto en el interior de la marquesina, ya que este esta siempre en la parte superior.

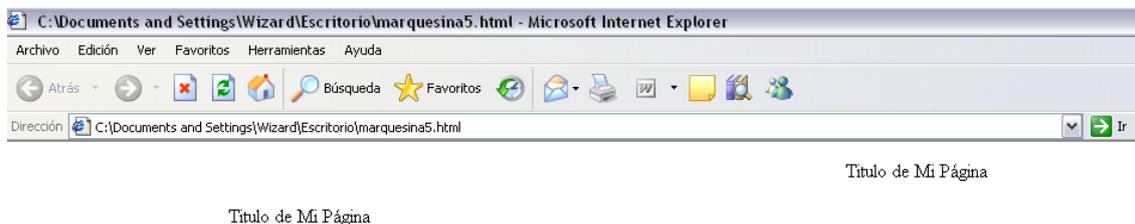
I.11 Cómo cambiar el comportamiento

Cuando creamos una marquesina el texto se desplaza en nuestro navegador de derecha a izquierda desapareciendo totalmente antes de volver a mostrarse por la derecha. Este efecto continúa indefinidamente lo suficientemente despacio para posibilitar su lectura. Este comportamiento, dirección, número de repeticiones y velocidad de movimiento puede ser modificado con 5 atributos de la etiqueta

MARQUEE:

- ✚ **BEHAVIOR:** Este atributo nos permite cambiar el comportamiento del texto y puede tomar tres valores:
 - ✓ **BEHAVIOR="scroll"**, es el valor por defecto y consiste en el efecto de desaparecer por un lado y aparecer por el otro.
 - ✓ **BEHAVIOR="slide"**, el texto tiene un comportamiento similar pero en cuanto llega al borde derecho se detiene.
 - ✓ **BEHAVIOR="alternate"**, en este caso el texto aparecerá igualmente por la derecha, pero una vez llegue al borde izquierdo rebotará y volverá hacia la derecha donde volverá a rebotar repitiendo este comportamiento indefinidamente. Los ejemplos de estas marquesinas se pueden encontrar en la carpeta de marquesinas.
- ✚ **DIRECTION:** Con este atributo podemos controlar la dirección de desplazamiento del texto cuando este es de tipo SCROLL. Puede tomar dos valores:
 - ✓ **LEFT (izquierda) RIGHT (derecha)** siendo el primero el valor por defecto (movimiento de derecha a izquierda). En el ejemplo lo podemos ver en la carpetas de marquesinas, marquesina5.

A continuación en la figura I.13 se ilustra la dirección de Marquesina



I.13 Dirección de Marquesina.

- ✚ **LOOP:** Determina cuantas veces se repetirá el desplazamiento del texto por la pantalla. Por ejemplo si usamos SCROLL="4" el texto se desplazará por la pantalla únicamente 4 veces, mientras que si usamos SCROLL="-1" o bien SCROLL="infinity" seguirá desplazándose indefinidamente. Este ejemplo se muestra en la carpeta antes mencionada marquesina6.
- ✚ **SCROLLAMOUNT:** Aunque haya sensación de movimiento el texto se desplaza a pequeños saltos. Este atributo determina el número de píxeles que debe desplazarse el texto en cada uno de ellos. Junto con el siguiente atributo podemos controlar la velocidad del movimiento. Si ponemos un valor alto el texto se moverá más rápidamente pero con saltos más bruscos (ejemplo marquesina7).
- ✚ **SCOLLDELAY:** Determina el tiempo de espera entre cada uno de los saltos. Este tiempo debe estar dado en milisegundos. Valores mayores también provocan mayor rapidez y brusquedad. Experimentando con SCROLLAMOUNT y SCROLLDELAY encontraremos la combinación con la que satisfacemos las necesidades de velocidad y suavidad de desplazamiento (ejemplo marqiesina8).

I.12 Títulos

En HTML se dispone de seis niveles para la jerarquía de títulos; para mostrar un título se encierra el texto del mismo entre las etiquetas **<Hn>...</Hn>** donde n varía entre 1 (el nivel más alto de la jerarquía) y 6 (el nivel más bajo). El texto delimitado por dichas etiquetas se muestra en negrita y el texto que lo sigue comienza en un párrafo nuevo. Este ejercicio lo podemos encontrar en la carpeta de html, 1º Clase, ejercicio 11.

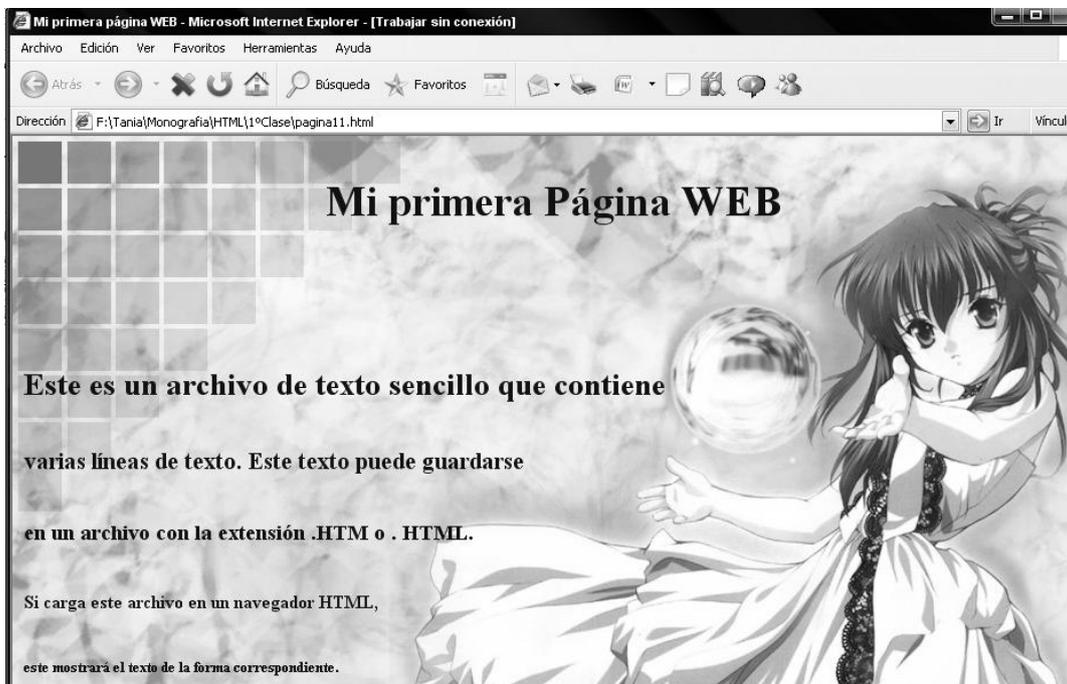
Un ejemplo de esto se muestra a continuación:

```
<h1><marquee>Mi primera Página WEB</marquee></h1>
<br>
<h2>Este es un archivo de texto sencillo que contiene</h2>
<br>
<h3>varias líneas de texto. Este texto puede guardarse </h3>
<br>
<h4>en un archivo con la extensión .HTM o .HTML.</h4>
<br>
```

<h5>Si carga este archivo en un navegador HTML,</h5>

<h6>este mostrará el texto de la forma correspondiente.</h6>

A continuación en la figura I.14 se ilustran Títulos en Página:



I.14 Títulos en Página.

Estas etiquetas sólo se deben utilizar para títulos de apartados, es decir, con la única finalidad de estructurar el texto y nunca para cambiar el estilo de texto dentro de párrafos.

🚩 **NOTA:** Los encabezados empiezan siempre en una línea nueva y el texto que les sigue comenzará en la línea siguiente. Además el navegador podrá mostrar un espacio vertical adicional entre el encabezado y el texto anterior y posterior.

I.13 Sangría etiqueta <blockquote> </blockquote>

Esta etiqueta se encuentra relacionada con el formato de párrafos en HTML. La etiqueta <blockquote>, que consta de una etiqueta de inicio y una de cierre. Que indica que el párrafo que encierra es un texto parafraseado. Al usar la etiqueta el navegador mostrará el texto en un nuevo párrafo con sangría a la izquierda y a la derecha. Este ejemplo lo podemos ver en la carpeta 1ºClase, página12. La etiqueta blockquote la podemos usar tantas veces queramos.

Es conveniente no usar mucho la etiqueta. La razón de esto es que las líneas en medio de la página dificultan la lectura de esta. Si se desea separar secciones un espacio vertical del tamaño adecuado es más efectivo.

A continuación en la figura I.15 se ilustra una sangría en la página



I.15 Sangría en la Página.

I.14 Texto en Negritas

Se usa la etiqueta `..`. La manera de usar esta etiqueta es como cualquier otra compuesta por instrucción de inicio e instrucción de fin, es decir debe introducirse el texto sobre el que se quiere actuar entre ambas instrucciones. Este ejercicio lo encontraremos en el carpeta 2ºClase, pagina1. Su forma de uso es la siguiente:

`TEXTO EN NEGRITA`

Por ejemplo podría usarse de la siguiente manera:

``

En un documento HTML, hay etiquetas que indican los atributos del texto (negrita, centrado, ...). Otras indican al sistema cómo responder a eventos que genera el usuario,

`
`

cómo señalar con el mouse a un icono que representa una película y en respuesta ejecutar

`
`

el programa que reproduce vídeo en formato digital. La etiqueta más importante es el vínculo

`
`

(link), que puede contener el URL de otro documento. Este documento puede residir en el

`
`

mismo lugar en Web que el documento actual o en cualquier otro ordenador de WWW.

`
`

El usuario 'navega' de documento en documento seleccionando estos vínculos con el mouse.

`
`

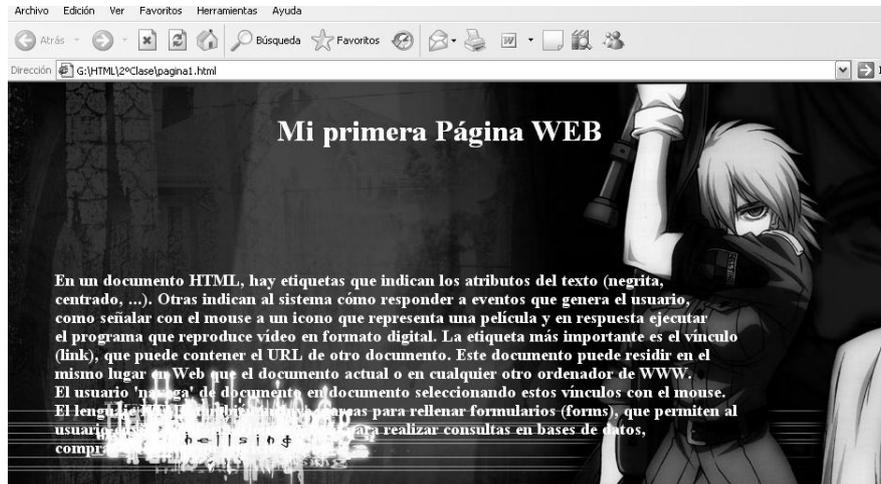
El lenguaje HTML también incluye marcas para rellenar formularios (forms), que permiten al

`
`

usuario enviar la información necesaria para realizar consultas en bases de datos,

comprar o solicitar un servicio.
</h3>

A continuación en la figura I.16 se ilustra el texto en negritas.



I.16 Texto en Negritas.

I.15 Texto en Cursiva.

Poner un texto en cursiva: en este caso se usa la etiqueta <i>..</i> como abreviatura de la palabra inglesa Italics que significa cursiva. Su forma de uso es la siguiente:

<i>
TEXTO EN CURSIVA
</i>

Esta etiqueta suele usarse para dar énfasis a alguna frase o texto pero sin que resalte demasiado. Otro uso común es para dar un formato especial a palabras de otro idioma o a títulos. Este ejercicio lo podemos ver en la carpeta 2ºClase, pagina2.

A continuación en la figura I.17 se ilustra el texto en cursiva.



I.17 Texto en Cursiva.

I.16 Texto Subrayado

Poner un texto subrayado: en este caso se utiliza la etiqueta <U> </U>. El nombre de la etiqueta proviene de la palabra inglesa *Underline* que significa precisamente subrayado. En general un texto se muestra subrayado para resaltar una información importante. Este ejercicio lo podemos ver en la carpeta mencionada en el ejemplo anterior pagina3.

Ejemplo:

El plazo límite para la entrega de solicitudes es el
<U>
15 de Junio
</U>

Su forma de uso es la siguiente:

<u>
En un documento HTML, hay etiquetas que indican los atributos del texto (negrita,

centrado, ...).
</h3>
</u>

A continuación en la figura I.18 se ilustra el texto subrayado.



I.18 Texto Subrayado

I.17 Tamaño del Texto

La etiqueta FONT permite tener un gran control sobre el tamaño de un texto. Este tamaño se especifica con su atributo SIZE y afectará al texto que encierran la instrucción de inicio () y de fin (). Este ejemplo lo podemos observar en la carpeta 2ºClase, pagina8.

La primera de ellas es indicar un tamaño absoluto de 1 a 7. Correspondiendo 7 al tamaño mayor y 1 al menor.

El tamaño por defecto del texto es el equivalente al número 3. U Ejemplo de esto es el siguiente:

Esto es texto normal

```
<FONT SIZE="6">
y esto es grande
</FONT>
```

Un ejemplo de todos los tamaños es el siguiente:

```
<font size=1>
En un documento HTML</font>
<br>
<fontsize=2>
Otras indican al sistema cómo responder a eventos que genera el usuario,
<br>
</font>
</font size=3>
cómo señalar con el mouse a un icono que representa una película y en respuesta
ejecutar
</font>
<br>
<font size=4>
el programa que reproduce vídeo en formato digital. vínculo
</font>
<br>
<font size=5>
Este documento puede residir en el
</font>
<br>
<font size=6>
mismo lugar en Web que el documento actual
</font>
<br>
<font size=7>
Hola
</font>
</FONT>
```

A continuación en la figura I.19 se ilustra el tamaño del texto.



I.19 Tamaño del Texto.

I.18 Color del Texto

La etiqueta `FONT: COLOR` permite cambiar el color de la misma manera como cambiamos el tamaño de la letra.

- ✚ Una de las formas de poder cambiar el color es por medio de código hexadecimal de la siguiente forma:

```
<FONT COLOR="#RRGGBB">
```

Donde:

- ❖ Tenemos que sustituir RR por dos números hexadecimales que indican la cantidad de rojo (desde 00 que es ausencia de color a FF que es máxima cantidad de color), GG por otros dos que indican la cantidad de verde y BB por dos que indican la cantidad de azul.
 - ❖ Combinando estos tres colores en diferentes proporciones se pueden crear hasta 16 millones de colores. Este sistema suele parecer ligeramente complicado en un principio, pero con la práctica o con ayuda de herramientas gráficas de selección de colores se convierte en un método bastante sencillo e increíblemente potente.
-
- ✚ La otra forma es indicando el color por su nombre. El cual consiste en poner como valor de COLOR el nombre del color en inglés. Los colores aceptados por todos los navegadores son:

- ✓ **Black** (negro),
- ✓ **white**(blanco),
- ✓ **green** (verde),
- ✓ **maroon** (marron rojizo),
- ✓ **olive** (verde oliva),
- ✓ **Navy** (azul marino),
- ✓ **purple** (violeta),
- ✓ **red**(rojo),
- ✓ **yellow** (amarillo),
- ✓ **blue** (azul),
- ✓ **teal** (verde azulado),
- ✓ **lime** (verde lima),
- ✓ **aqua** (azul claro),
- ✓ **fuchsia** (fusia),
- ✓ **silver** (gris claro).

Cabe mencionar que existen muchos más colores marcados tanto en código hexadecimal como con nombre en inglés, la tabla con estos colores se encuentra ubicada en el anexo III, tabla de colores de HTML, los cuales no son soportados en todos los navegadores.

La forma correcta de usar la etiqueta es la siguiente

```
<FONT COLOR="WHITE"><h1><marquee>Mi primera PáginaWEB
</marquee>
</h1>
</FONT>
```

```
<br><br>
<br><br>
<br>
<blockquote>
<br>
<h3>
<font color="FF00FF">
En un documento HTML, hay etiquetas que indican los atributos del texto (negrita,
<br>
centrado, ...).
</FONT>
```

Como podemos observar podemos aplicar tantos colores, como deseemos en la página, en este caso se utilizó un color para la marquesina, y otro color diferente para el texto.

A continuación en la figura I.20 se ilustra el color del texto.



I.20 Color del Texto.

I.19 Cambio del Tipo de Fuente

Esta etiqueta permite indicar un tipo de fuente para el texto, debiendo elegir el navegador la más próxima (en cuanto a apariencia se refiere) a la seleccionada por el autor del documento. Cabe mencionar que, si la persona que se encuentra viendo la página, no cuenta con el tipo de fuente seleccionada por default, tomará el tipo arial, que es la más común en todos los exploradores.

Un ejemplo de esto es:

```
<font color="FF00FF">
<font face="arial">
En un documento HTML, hay etiquetas que indican los atributos del texto <br>
</FONT>
</font>
<FONT COLOR="RED">
<font face="Monotype Corsiva">
<br>
El usuario 'navega' de documento en documento seleccionando estos vínculos <br>
</FONT>
</font>
```

A continuación en la figura I.20 se ilustra el tipo de fuente.



I.20 Tipo de Fuente.

- **Nota:** Si queremos colocar diferentes tipos de letra para que cuando abran la página el navegador pueda ver alguno de los predefinidos lo podemos hacer de la siguiente manera:

- **Ejemplo:**

```
<font face="arial", "monotype Cursiva">
```

En un documento HTML, hay etiquetas que indican los atributos del texto


```
</font>
```

De esta manera el explorador buscará entre estas dos fuente, si contiene alguna de ellas visualizará la parte que se encuentre restringida por esta etiqueta, con este formato de letra de lo contrario, la mostrará en el formato por default, que antes se mencionó.

I.20 Alineación del Texto

En HTML se tiene la posibilidad de alinear texto con esto se puede crear documentos *HTML* que no tienen nada que envidiar a documentos elaborados con un procesador de texto; mediante el atributo **ALIGN** de la etiqueta **<P>** el diseñador puede alinear el texto como desee: centrado (**CENTER**), justificado (**JUSTIFY**), alineado a derecha (**RIGHT**) o a izquierda (**LEFT**). Este ejemplo lo podemos ver en la carpeta 2ºClase, pagina7.

Véase el código siguiente:

```
<font color="FF00FF" face="arial">
```

```
<p align="right">
```

En un documento HTML, hay etiquetas que indican los atributos del texto

```
<br>
```

```
</p>
```

```
<p align="left">
```

El usuario 'navega' de documento en documento

```
<br>
<p align="center">
El lenguaje HTML también incluye marcas para
</p>
</font>
```

A continuación en la figura I.21 se ilustra la alineación del texto.



I.21 Alineación del Texto.

- **Nota:** Hay que mencionar que los tres atributos que contiene font, los podemos encapsular en un sola etiqueta de la siguiente manera:

```
<font color = "red" face = "Arial" size=5> Hola </font>
```

I.21 Inserción de Imágenes

Un elemento muy importante en los documentos son las imágenes o gráficos. Hay que recordar que una imagen dice más que mil palabras, HTML puede integrar imágenes en formato GIF, JPEG, PNP, BMP y vectoriales.

Debemos recordar que las extensiones de las imágenes son Gif(.gif), JPEG(.jpg), PNP (.pnp), BMP (.bmp).

Para colocar una imagen utiliza la etiqueta su sintaxis es:

```

```

Ejemplo:

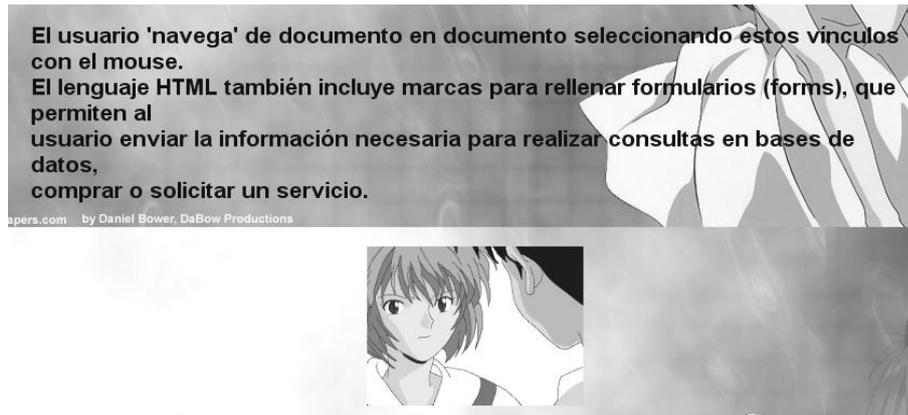
```

```

🚦 Donde:

- ❖ **src** : es la dirección donde se encuentra la imagen, junto con su extensión.

A continuación en la figura I.22 se ilustra Inserción de Imágenes.



I.22 Inserción de Imágenes

Este ejercicio lo podemos encontrar en la carpeta 3ºClase, pagina1.

- ❖ **Atributos:** Son las propiedades que se le dan a la imagen entre las cuales están:
 - **ALIGN:** admite los valores **TOP**, **MIDDLE**, **BOTTOM**, **RIGHT**, **LEFT** para alinear la imagen respecto al texto que la rodea. Este ejemplo se puede ver en la Carpeta 3ºClase, pagina1.

Ejemplo de uso de la etiqueta:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE><!--PONER TÍTULO A LA PÁGINA-->
</HEAD>
<BODY background="G:\HTML\3ºClase\Fondo\angel.jpg"> <!-- Poner imagen de fondo-->
<br>
<bR>

Imagen alineada con el texto por la parte superior.
<br>
<br>

Imagen centrada respecto al texto.
<br>
<br>

Imagen alineada con el texto por la parte inferior.
<br>
<br>
</BODY>
</HTML>
```

A continuación en la figura I.23 se ilustra la alineación de Imágenes con texto.



I.23 Alineación de Imágenes con texto.

Este ejercicio lo podemos encontrar en la carpeta 3ºClase, pagina_1.

- **ALT**: muestra un mensaje cuando se señala con el puntero del Mouse, llamado texto explicativo de la imagen.

Su estructura es la siguiente:

```

```

La estructura Completa es:

```
<HTML>
<HEAD>
<TITLE> Mi primera página WEB</TITLE>
</HEAD>
<BODY background="G:\HTML\3ºClase\Fondo\angel.jpg">
<FONT COLOR="blue"><h1><marquee>Mi primera Página
WEB</marquee></h1></FONT>
<br>
<center></center>
<br>
<center>
</BODY>
</HTML>
```

A continuación en la figura I.24 se ilustra la colocación de texto explícito.



I.24 Colocación de Texto Explicito.

Este ejercicio lo podemos encontrar en la carpeta 3ºClase, pagina4.

- **WIDTH y HEIGHT:** permiten especificar las dimensiones de la imagen (de forma absoluta, mediante un valor en píxeles, como un porcentaje de las dimensiones de la ventana del navegador). Con esto si nosotros queremos hacer que una imagen sea más pequeña o más grande de lo que originalmente es lo podemos hacer. Hay que recordar si nuestra imagen es muy pequeña y la queremos hacer más grande puede tener ciertas que las modificaciones afectan la calidad de la imagen, al contrario de las grandes, su formato, no cambiará en casi nada, los cambios no son muy notorios. Este ejemplo podemos verlo en la carpeta 3ºClase, pagina5.

Su estructura es la siguiente:

```

```

A continuación en la figura I.25 se ilustra el cambio de tamaño de la imagen.



I.25 Cambio de Tamaño de l Imagen.

Este ejercicio lo podemos encontrar en la carpeta 3ºClase, pagina3.

- **BORDER:** permite establecer (o eliminar) un borde alrededor de la imagen, indicándolo por medio de un número mientras más grande sea el número el borde será más grande. Este ejemplo lo podemos visualizar en la carpeta antes mencionada antes pagina2.

Su estructura es:

```

```

A continuación en la figura I.26 se ilustra la colocación del borde.



I.26 Colocación del Border.

Este ejercicio lo podemos encontrar en la carpeta 3ºClase, pagina2.

I.22 Líneas Horizontales

Para separar dos secciones diferentes de una página, ya sea texto o no, puede emplearse la etiqueta: `<hr>`. Esta etiqueta está formada por una única instrucción y provoca que el navegador inserte una línea horizontal que puede separar párrafos o imágenes que se están comparando. Existen cuatro atributos que permiten modificar su apariencia. Estos son:

- ✓ **align:** Permite cambiar la alineación de la línea. Puede tomar tres valores: "*left*", "*center*" y "*right*" para conseguir alineación a la izquierda, centro y derecha respectivamente.

Su estructura es la siguiente:

```
<hr align= "left">
```

- ✓ **noshade:** Al incluir este atributo se elimina de la línea la sombra que algunos navegadores pueden llegar a incluir, de esta manera el navegador sólo mostrará una línea horizontal negra.

Su estructura es la siguiente:

```
<hr noshade>
```

- ✓ **size:** Con este atributo se puede controlar el grosor de la línea. Como valor de este atributo se debe especificar el valor deseado en píxeles (puntos de pantalla).

Su estructura es la siguiente:

```
<hr size=10>
```

- ✓ **width:** Este atributo permite especificar el ancho de la línea. El valor puede ser dado de una manera absoluta, especificando el ancho en píxeles (por ejemplo width=100) o de manera relativa respecto al ancho de la ventana del navegador usando porcentajes (por ejemplo width="90%"). Es más que recomendable usar este último sistema de manera que se obtengan un resultado apropiado independientemente del tamaño que cada usuario quiera dar a la ventana de su navegador.

Su estructura es la siguiente:

```
<hr width=300 align="center">
```

- ✓ **color:** Este atributo permite darle un color diferente al que muestra por default, basta con escribir el nombre en ingles o en código hexadecimal para que muestre el color seleccionado

Su estructura es la siguiente:

```
<hr color = "blue">
```

Estos atributos pueden ser usados conjuntamente para combinar sus efectos. Si desea ver el ejemplo que se muestra lo encuentra en la carpeta 3ºClase, pagina12.

```
<hr align= "left" size= 10 width = 300>
```

A continuación en la figura I.27 se muestra la colocación de líneas horizontales.



I.27 Colocación de Líneas Horizontales

I.23 Listas

Las listas son un elemento muy útil para presentar información de una forma clara y organizada. En *HTML* se pueden crear dos tipos distintos de listas: *descriptivas* y *regulares*; las primeras permiten mostrar una serie de conceptos junto con su descripción formando un glosario, las segundas se utilizan para realizar enumeraciones de elementos.

La forma de crear listas difiere ligeramente de los elementos HTML vistos en este momento, ya que consta de dos pasos.

Primero se debe definir el tipo de lista y después deben insertarse cada uno de los elementos de lista.

Al crear una lista se debe indicar su inicio y su fin. Para empezar la lista usaremos la instrucción de inicio de la etiqueta de esa lista.

🚩 Lista no Numerada

El tipo de listas más usadas en las páginas Web son las listas no ordenadas.

Para hacer una lista no numerada debemos usar la etiqueta *UL* (su instrucción de inicio será `` con su etiqueta de cierre ``). Una vez *abierta* la lista debemos insertar, cada uno de los elementos de lista. Para ello usamos otra etiqueta (o dos en algún caso) que dependerá del tipo de lista. En el caso de la lista no ordenada los elementos se introducen con la etiqueta de una única instrucción *LI*, además esta etiqueta también se debe cerrar ``, por ejemplo:

Lista No Numerada

```
<UL>  
<br>  
<li> Amarillo</li>  
<li> Rojo </li>  
<li> azul</li>  
<li> verde </li>  
<li> morado </li>  
</UL>
```

A continuación en la figura I.28 se muestra la lista no numerada.



I.28 Lista no Numerada.

Lista Ordenada

Este tipo de listas numera los elementos de lista en el orden en el que se han introducido en el código HTML. La numeración se realiza con números empezando por el 1.

Este tipo de listas quedan delimitadas por la etiqueta `...`, y cada elemento que se inserta, al igual que en el caso de listas no ordenadas, los elementos se introducen con la etiqueta única de instrucción `LI`, que se debe cerrar `` con la etiqueta ``. Por ejemplo:

```
<ol>
<li> Etiquetas Básicas</li>
<li> Formato de Texto </li>
<li> Inserción de Imágenes </li>
<li> Creación de Listas</li>
<li> Enlaces </li>
<li> Utilización de Frame </li>
<li> Evaluación </li>
</ol>
```

A continuación en la figura I.29 se muestra la lista numerada.



I.29 Lista Numerada.

Anidamiento de listas

Las etiquetas *HTML* pueden anidarse de tal forma que varias pueden ser aplicadas a un mismo texto. Las listas no son una excepción y un caso típico de anidamiento de etiquetas se da en la creación de listas.

Listas de glosario

Este tipo de listas es usado, como su nombre indica, para crear glosarios, es decir listas de definiciones o de descripciones. El principio y final de este tipo de listas se delimitan con la etiqueta `<DL>...</DL >`.

La inserción de elemento es ligeramente distinta a la del resto de listas ya que no se usa la etiqueta . Esto es así porque en este caso cada uno de los elementos de lista está compuesta de dos partes:

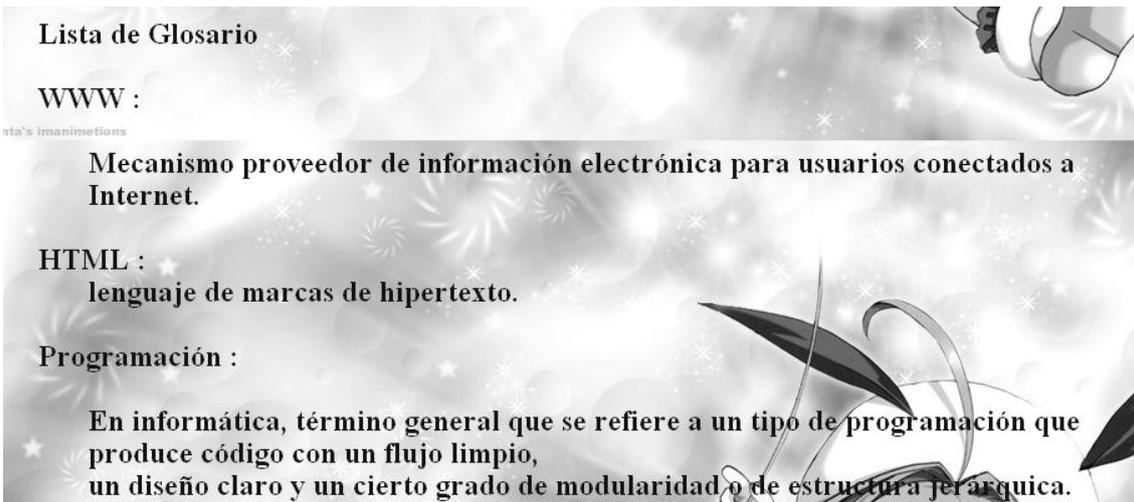
- **La palabra que queremos definir:** para insertar la palabra usaremos la etiqueta <DT>. Esta nueva etiqueta funciona de forma muy similar a y la instrucción de fin (</DT>).

- **La definición o descripción de esa palabra:** una vez indicada la palabra es hora de poner su definición. Para ello usamos otra etiqueta: <DD>. La definición suele ser un párrafo y será mostrado por los navegadores con una sangría con respecto al texto al texto que queremos definir.

Por ejemplo:

```
<dl>
<dt> WWW : </dt>
<br>
<dd>
<br>
Mecanismo proveedor de información electrónica para usuarios conectados a Internet.
</dd>
<br>
<br>
<dt> HTML :
<br>
<dd>lenguaje de marcas de hipertexto.</dd>
<br>
<br>
<dt>Programación : </dt>
<br>
<dd>
<br>
En informática, término general que se refiere a un tipo de programación que produce
código
con un flujo limpio,
<br>
un diseño claro y un cierto grado de modularidad o de estructura jerárquica.
</dl>
```

A continuación en la figura I.30 se muestra la lista de glosario.



I.30 Lista de Glosario.

Nota: Los ejemplos de listas pueden ser vistos en el ejemplo de la carpeta 4ºClase, pagina1.

I.24 Hipervínculos

Ahora que ya conocemos diferentes etiquetas para texto, párrafos e insertar imágenes en una página, tenemos que aprender y ser capaces de crear una página Web bastante completa con los enlaces hipertexto.

Mediante los hipervínculos podemos resaltar frases o palabras de la página para que una vez que se pulse sobre ellos nos manden directamente a la otra página de Internet o del sitio que decidamos ver.

Los enlaces o hipervínculos (en inglés *links*) no son más que palabras o frases especiales de un texto que ofrecen la posibilidad de hacer clic sobre ellas para ir a otras páginas. El concepto del hipervínculo. *Resulta novedoso ya* que ofrecen enlaces en la WWW, lo cual permite la posibilidad de pulsar y navegar hasta páginas que se encuentren en el otro extremo del mundo. El usuario puede moverse de un sitio que se encuentra en México hasta un sitio que se encuentra en Japón con un único clic sobre el texto resaltado.

Hay que mencionar que aunque la mayoría de las veces los links se usan para enlazar una páginas con otras también pueden usarse con imágenes, sonido, vídeo y prácticamente cualquier tipo de archivo que deseemos.

Para poder colocar un hipervínculo es necesario colocar la `<a> ...` la cual permitirá incluir en las páginas enlaces hipervínculo. Su atributo 'href' permite indicar cual es la página a la que nos vamos a dirigir al hacer clic sobre el texto adecuado. El texto del enlace es aquel que se introduce entre la instrucción de inicio y la instrucción de final de la etiqueta y esta frase es resaltada por el explorador subrayándola, e indicándonos que podemos hacer clic sobre ella. La estructura de un link es:

`Texto que será sensible (hipertexto)`

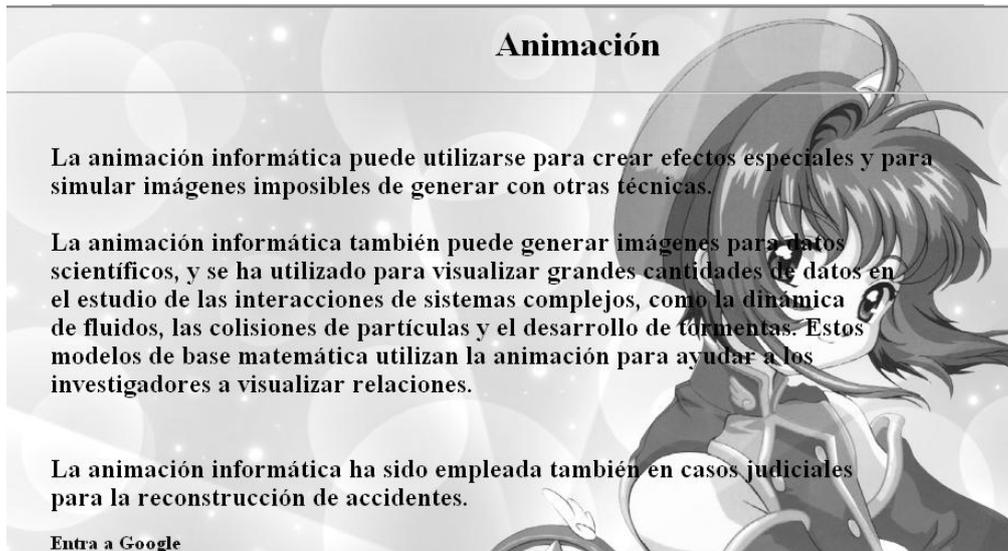
Los diferentes hipervínculos que podemos hacer son:

Hipervínculo a una Página de Internet

`< a href. ="http://www.google.com.mx/">Entra a Google `

Dentro de esta estructura, hay que observar que la dirección a la página que estamos direccionando, debe contener toda la dirección del sitio, así como el protocolo el cual utiliza para comunicarse, de lo contrario, mostrará una página la cual nos indica que no se pudo establecer conexión con la página indicada.

A continuación en la figura I.31 se muestra la colocación de un hipervínculo a un buscador.



I.31 Colocación de Hipervínculo a un buscador.

Aquí podemos observar cómo es que nos muestra el enlace, que por default todos los exploradores, marcan con color azul, a los enlaces. Este ejemplo lo podemos ver en la carpeta de HTML 5ºClase, pagina1.

Hipervínculo a un Archivo

Cuando visitamos algunas páginas en Internet, hemos observado que existen enlaces que nos permiten bajar algún archivo de Internet, ya sean archivos de texto (.doc, .txt), hojas de calculo, imágenes, sonidos, videos, esto lo podemos hacer con la siguiente estructura:

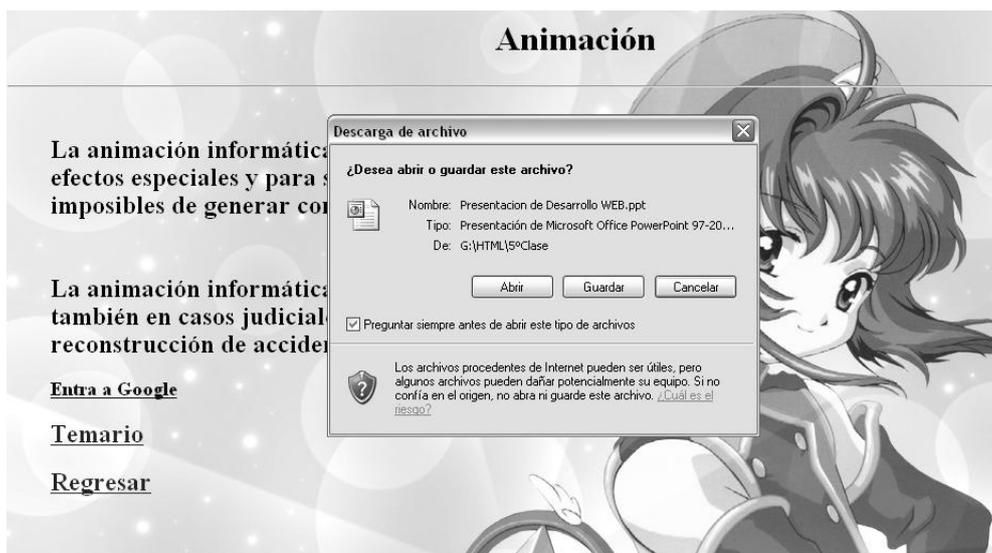
`Temario</h2>`

De igual manera que en el hipervínculo anterior se pone la etiqueta `<a>...`, solo que en esta ocasión en vez de poner la dirección de la página Web a la que deseamos ir, se debe colocar la dirección completa en que se encuentra el archivo el cual queremos que puedan descargar desde nuestra página, agregándole el nombre del archivo, junto con la extensión que tiene el archivo. continuación en la figura I.32 se muestra el hipervínculo direccionado al archivo.



I.32 Hipervinculo direccionado al archivo

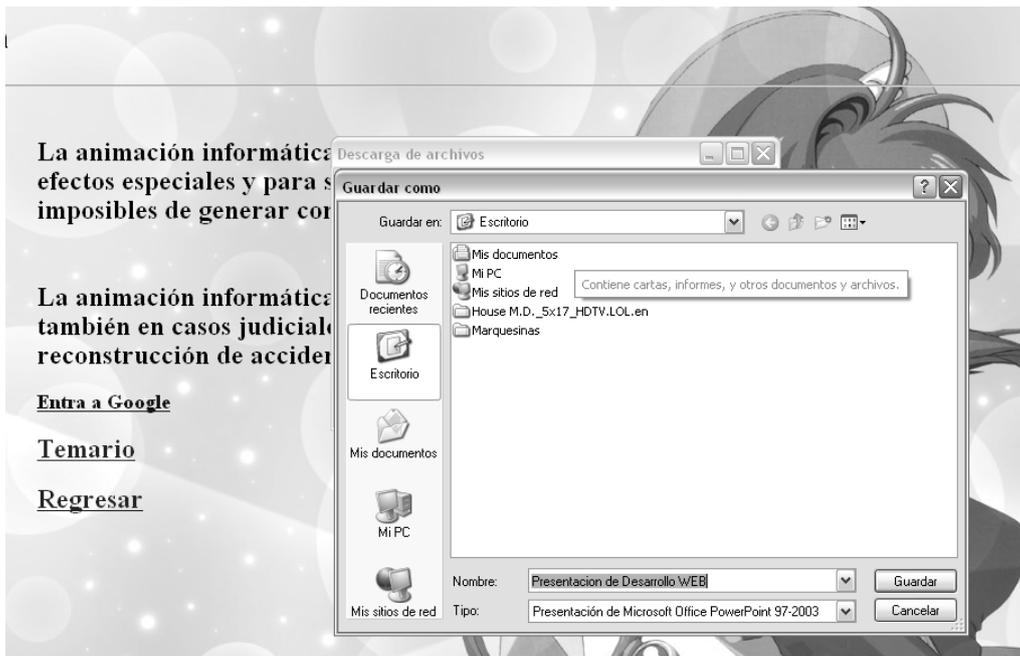
Una vez que demos clic, en el enlace, que en este caso lleva la etiqueta de Temario, nos aparecerá la siguiente ventana. Como se muestra en la figura I.33 descarga de archivo :



I.33 Descarga de Archivo.

En esta ventana se muestran las opciones, de si deseamos abrir el archivo, o guardarlo, para verlo más tarde.

Si seleccionamos la opción de guardar, aparecerá la pantalla que nos indica en qué lugar lo deseamos guardar. Como se muestra en la figura I.34. Guardar archivo:



I.34 Guardar Archivo.

Por el contrario, si seleccionamos la opción de abrir, entonces el documento se abre automáticamente. Como se muestra en la figura I.35 Abrir archivo:



I.35 Abrir Archivo

Hay que recordar que podemos poner cualquier clase de archivo, el cual deseemos que puedan bajar todas las personas que quieran este archivo, aclarando claro que debemos saber la extensión que tiene el archivo, para indicárselo al editor de texto. Un ejemplo de esto lo podemos ver en la carpeta 3ºClase, pagina2.

🚦 Hipervínculo hacia otra página de nuestro sitio.

Si nosotros queremos hacer un hipervínculo, el cual nos vuelva a direccionar a la página que originalmente estábamos, o en caso contrario que nos direcciona a otra página del mismo sitio, lo podemos hacer con la siguiente estructura:

```
<a href="G:\HTML\5ºClase\pagina1.html">Regresar</a>
```

Aquí podemos observar como colocamos la dirección en donde se encuentra la página, el nombre de la página y la extensión de este mismo. Si el enlace se coloca correctamente nos mandará a la página la cual le indicamos. Este ejemplo puede ver con su código completo en la carpeta 5ºClase, pagina2_1.

A continuación en la figura I.36 se muestra el hipervínculo a otra página.



I.36 Hipervínculo a otra página.

Hipervínculo a una dirección de correo.

Si queremos que en el sitio se tenga un enlace con el cual podamos interactuar con la persona que nos visite y nos pueda dejar algunos comentarios sobre la página, lo podemos hacer por medio de la estructura:

```
<a href="mailto:wizard_pag@yahoo.com.mx">Envia Correo con tus Comentarios </a>
```

Donde:

- ❖ **mailto:** Le indica al explorador, que el hipervínculo tendrá una salida a correo electrónico, después de esta palabra seguido de dos puntos, se debe colocar el correo electrónico al cual deseamos sean mandados.

A continuación en la figura I.37 se muestra el hipervínculo al correo electrónico.

Animación

La animación informática puede utilizarse para crear efectos especiales y para simular imágenes imposibles de generar con otras técnicas.

La animación informática ha sido empleada también en casos judiciales para la reconstrucción de accidentes.

[Entra a Google](#)

[Temario](#)

[Envía Correo con tus Comentarios](#)



I.37 Hipervínculo a Correo Electronico.

Si nosotros damos clic, nos mandara a la pantalla de Outlook, la cual por default, nos dará el correo, que nosotros decidimos, para que nos hagan llegar sus comentarios. Este ejercicio, lo podemos ver el la carpeta 5ºClase, pagina3.

🚩 Hipervínculo dentro de la misma página.

Para colocar un enlace que nos posicione en un lugar dentro de la misma página necesitamos utilizar un indicador el cual nos informe a qué lugar deseamos ir, su estructura es la siguiente:

```
<html>
<head>
<title>Hipervínculos en HTML</title>
</head>
<body>
<a name="top"><h2>Ejemplo de hipervínculos locales</h2></a>
<ul>
<li><a href="#Z1">Introducción a Internet</a></li>
<li><a href="#Z2">Qué es HTML</a></li>
<li><a href="#Z3">Herramientas HTML</a></li>
</ul>
<hr>
<a name="Z1"><h3>Introducción a Internet</h3></a>
A finales de los años sesenta el ejército estadounidense buscaba
una red descentralizada para sus ordenadores.
<br>
<a href="#top">Atrás</a>
<hr>
<a name="Z2"><h3>Qué es HTML</h3></a>
Los documentos que contiene la <i>World Wide Web </i></a>(WWW)
se definen a través del lenguaje de descripción de documentos HTML.
<a href="#top">Atrás</a>
<hr>
```

```
<a name="Z3"><h3>Herramientas HTML</h3></a>
Para el diseño de páginas WWW es sumamente útil
que tenga ciertos conocimientos sobre HTML <br>
<a href="#top">Atrás</a>
<hr>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta 5ºClase, pagina7.

Esto se lleva acabo por medio de un identificador, el cual da la señal al enlace para buscar la dirección en donde se debe posicionar, en este caso se utiliza una almohadilla (#), la cual define el nombre que se le dará al enlace. Posterior mente cuando el usuario da doble clic en el enlace, el navegador busca dentro de la página la otra instrucción que nos dará el lugar indicado del sitio, ejemplo:

Primera instrucción:

```
<a href="#Z1"> Introducción a Internet</a>
```

Busca dentro del Navegador:

```
<a name="Z1">Introducción a Internet</a>
```

A continuación en la figura I.38 se muestra el hipervínculo dentro de la misma página.

Ejemplo de hipervínculos locales

- [Introducción a Internet](#)
- [Qué es HTML](#)
- [Herramientas HTML](#)

Introducción a Internet

A finales de los años sesenta el ejército estadounidense buscaba una red descentralizada para sus ordenadores.
[Atrás](#)

Qué es HTML

Los documentos que contiene la *World Wide Web* (WWW) se definen a través del lenguaje de descripción de documentos HTML. [Atrás](#)

Herramientas HTML

Para el diseño de páginas WWW es sumamente útil que tenga ciertos conocimientos sobre HTML
[Atrás](#)

I.38 Hipervínculo dentro de la misma Página.

Este ejercicio, lo podemos ver en la carpeta 5ºClase, pagina7.

Definir el Color de los Hipervínculos.

Cuando creamos una página WEB con enlaces es muy cotidiano que el color que tienen los enlaces por default, no coincidan del todo con los colores de la página, por lo que el color se pierde; dentro de las propiedades que se tienen en el body, tenemos la facilidad de cambiar el color de los enlaces, dentro de esta etiqueta tenemos diferentes opciones:

- ✓ **link** : Con esta opción podemos definir el color que tendrán los enlaces, cuando el usuario habrá el sitio Web, por primera vez, el color lo podemos definir por medio de código hexadecimal o por su nombre en inglés, como lo hemos mencionado en etiquetas anteriores. Su estructura es la siguiente:

<body link="#xyyzz">

- ✓ **vlink** : Esta opción nos permite dar el color a los enlaces ya visitados por el usuario, e indicarle que ya ha visto esta página. Al igual que link, el código del color se define de la misma manera. Su estructura es la siguiente:

<body vlink="#xyyzz"> define el color de los enlaces visitados.

- ✓ **alink** : Esta propiedad define el color del enlace el cual estamos viendo en ese momento indicándole al usuario que actualmente se encuentra visualizando este enlace. Su código de color se define de la misma manera que los anteriores. Su estructura es:

<body alink="#xyyzz">

- ✓ **Todos estos parámetros se pueden agrupar en una única etiqueta <body>:**

<body bgcolor="#xyyzz" text="#xyyzz" link="#xyyzz" vlink="#xyyzz" alink="#xyyzz">

I.25 Frames

Los frames⁵ permiten dividir la ventana del navegador en más de una zona, estas zonas tiene a su vez propiedades como: cargar un archivo independiente del frame. Entre otras las cuales se explicarán a continuación:

Cambios en el Documento HTML

Cuando utilizamos código HTML, siempre para visualizar algo dentro del navegador lo colocamos dentro de la etiqueta <body>, la cual nos muestra todo lo que deseamos ver.

Un documento de frames tiene una estructura muy similar a la de un documento HTML, salvo que carece de cuerpo, y no se utiliza la etiqueta <body>, ya que en esa zona se escribe la etiqueta <frameset>, la cual se encarga de hacer las divisiones correspondientes dentro del navegador.

Ejemplo:

```
<html>
<head>
<title>      </title>
</head>
<frameset>
```

⁵Frame: Marco o cuadro

```
</frameset>
</html>
```

Donde la etiqueta <frameset>, contiene dos atributos los cuales son:

- ✓ **rows**: Este atributo se encarga de la división horizontal de la página
- ✓ **cols** : Este atributo se encargó de la división vertical de la página.

Ambos atributos pueden tomar distintos valores, separados por una coma, para indicar el número de divisiones que deberá de realizar. Dentro de los valores que puede tomar se encuentran:

- ❖ **Entero**: Se toma como un número en píxeles
- ❖ **1%-100%**: Si se elige la opción de porcentajes toma las medidas del navegador, y posiciona al documento dentro del frame.
- ❖ **[Entero]***: Cuando se utilizan asteriscos, estamos hablando de un valor relativo, ya que este, divide la ventana según el tamaño que se requiera.
- ❖ ***** : Toma el valor de la ventana que se tenga, es decir, si nos encontramos con ventanas las cuales tengan dos partes definidas, con un valor exacto, y colocamos un *, este tomará, la ventana restante por ocupar.

Dentro del <frameset></frameset> se coloca la etiqueta <frame>, la cual describe la división que tendrá la página, unos de sus atributos son:

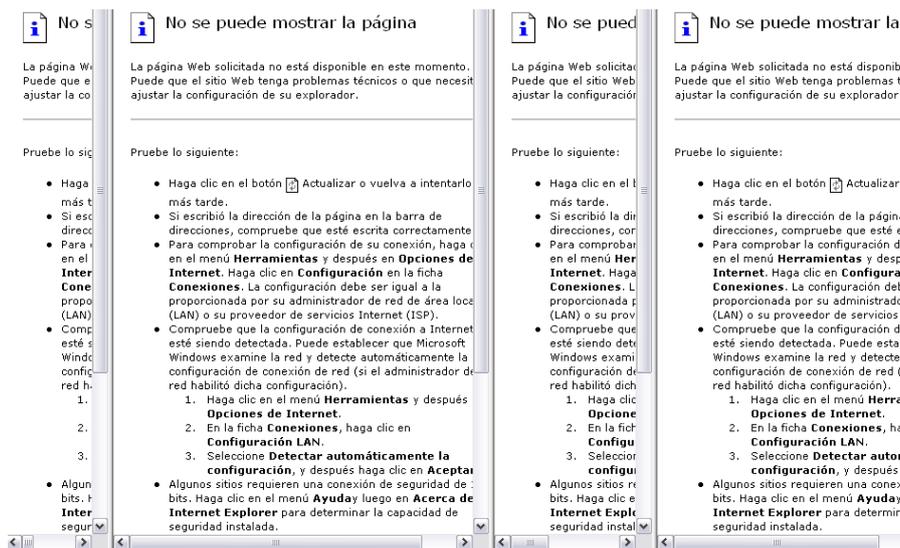
- ✓ **SRC**: Toma la dirección del documento que se desea visualizar.
- ✓ **NAME**: Este atributo se utiliza para dar un nombre al frame y poder, mandarlo a llamar por medio de un enlace.
- ✓ **Marginwidth**: Aquí indicamos el ancho del margen que tendrá el frame en píxeles.
- ✓ **Marginheight**: Con este atributo indicamos la altura que tendrá el margen, en el frame.
- ✓ **Scrolling**: Puede tomar tres valores los cuales son yes, no, auto, las cuales indican si el frame contendrá barras de desplazamiento (yes), no lo las tendrá (no), o si se requieren que las coloque automáticamente (auto).

Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame.

Un ejemplo de la utilización de estos atributos es:

```
<html>
<head>
<title>Desarrollo de Frames</title>
</head>
<frameset cols= "50,50%, *,2*">
<frame src = "1.html">
<frame src = "2.html" scrolling=auto>
<frame src= "3.htm" scrolling=yes>
<frame src = "4.html" scrolling=no>
</frameset>
</html>
```

A continuación en la figura I.39 se muestra división de páginas por medio de Frames.



I.39 División de Página por medio de Frames.

Hay que recordar que mientras no tengamos los documentos con la extensión .html no podremos visualizar en el frame, esta será la vista que nos dará el explorador.

Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame1.

En la actualidad muchos de los navegadores que conocemos tienen la propiedad de soportar frames, por lo cual no debemos tener ningún problema al visualizar las páginas, pero hay que tener en cuenta a las personas, que aun utilizan algún explorador antiguo, esto hace que no puedan visualizar la página de la manera correcta en caso de que contenga frames, existe un comando el cual se encarga de avisar que su explorador no soporta los frames que contiene la página, lo malo de ello, es que este comando solo lo reconocen algunos exploradores. Su sintaxis es:

<NOFRAME>Tu Navegador no Soporta Frames</NOFRAME>

Dentro de esta etiqueta podemos indicarle que su explorador no soporta los frames. Y Dándole un link (enlace), para que puedan visualizar la página de otra manera más sencilla, sin la necesidad de usar frames. Su estructura es la siguiente:

```
<html>
<head>
<title>Uso de NoFrames</title>
</head>
<frameset cols="25%,*">
<frame name="sx" src=" 5.html">
<frame name="central" src=" 2.html">
</frameset>
<noframe>
<body>
<h2>Su navegador no soporta frames</h2>
<br>
Pulse aquí: <a href=" 5.html">Contenido</a>
</body>
```

```
</noframe>
</html>
</html>
```

🚦 Dividir Ventanas Verticalmente.

Para poder diseñar un frame correctamente necesitamos de dos etiquetas básicas las cuales tienen diferentes funciones:

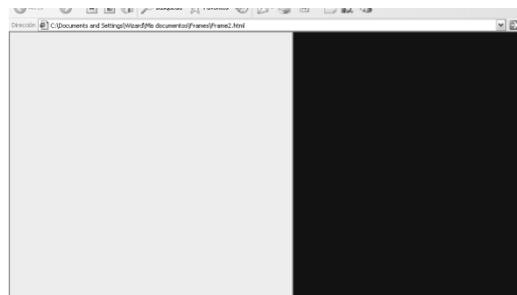
- ✓ **<frameset>**: Se encarga de dividir la ventana del navegador en diferentes frames.
- ✓ **<frame>**: Establece que debe ocurrir dentro de la ventana. (Es decir que debe mostrarse dentro de ella).

Para poder realizar frames de forma vertical necesitamos la ayuda del atributo cols, que ya habíamos mencionado antes. En ejemplo de esto es:

```
<frameset cols= "50%, *">
  <frame src= "1.html">
  <frame src= "2.html">
</frameset>
```

Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame2.

A continuación en la figura I.40 se muestra división de páginas ventana verticalmente.



I.40 División de Ventanas Verticalmente.

El tag `<frameset>.....</frameset>`, se encarga de dividir la ventana del navegador. Con el atributo cols, le indicamos al navegador que dividirá la ventana en columnas. En este ejemplo le estamos indicando que la primera columna tendrá un ancho del 50% de la ventana de navegador y con *, el tamaño de la segunda columna, la cual tomará el resto de la ventana que falta.

La etiqueta `<frame` se encarga de colocar la información que será mostrada en el navegador, utilizando el atributo src, el cual indicará la ruta del documento a mostrar.

Hay que recordar que la forma en que se presentaran los documentos va conforme a la forma en que fueron colocados es decir, el primer `<frame>`, que se encuentre en el código ocupará, la posición de la primera división de la ventana, el segundo la segunda división de la ventana y así sucesivamente, hasta terminar con todas las ventanas que le coloquemos.

Los archivos que deben contener los frames son archivos muy sencillos, los cuales se encuentran escritos en puro código HTML, un ejemplo de ellos es:

```
<html>
<head>
<title> </title>
</head>
<body bgcolor="yellow">
Este es un ejemplo de Código HTML en el frame
</body>
</html>
```

A continuación en la figura I.41 se muestra el contenido de la página.



I.41 Contenido de la Página.

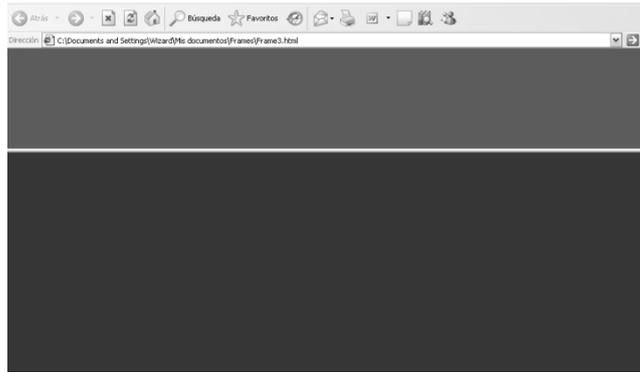
✚ **NOTA:** Por ningún motivo se debe agregar la etiqueta `<body>` después de la etiqueta `</head>`, para encerrar a la etiqueta `<frameset>`. La etiqueta `<body>` solo se podrá colocar junto con la etiqueta `<noframe>`. De lo contrario cuando nos mande un aviso solo veremos una página en blanco.

✚ Dividir Ventanas Horizontalmente

Para dividir la ventana del navegador de forma horizontal se utiliza el atributo `rows`, el cual le indica al navegador que la división será de forma horizontal. Un ejemplo de esto es:

```
<frameset rows= "25%.*">
<frameset src= "3.html">
    <frameset src= "4.html">
</frameset>
```

A continuación en la figura I.42 se muestra la división de ventana horizontalmente.



I.42 División de Ventana Horizontalmente.

El tag `<frameset>.....</frameset>`, se encarga de dividir la ventana del navegador. Con el atributo `rows`, le indicamos al navegador que dividirá la ventana en filas. En este ejemplo le estamos indicando que la primera fila tendrá un ancho del 25% de la ventana de navegador y con `*`, el tamaño de la segunda columna, la cual tomará el resto de la ventana que falta, que será de un 75%. Después con `<frame>`, indicamos los documentos que deben mostrarse.

Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame3.

Crear más de dos Frames.

Los frames permiten incorporar más de dos frames en la ventana. Para hacer esto se tienen que dar nombres a cada una de las ventanas. Esto será necesario para acceder a cada una de las ventanas.

La división del área que representan los frames, se lleva a cabo mediante la etiqueta `<frameset>`, a través de los atributos `rows` y `cols`. Si queremos utilizar más de dos frames debemos indicarlos dentro de estos atributos. Ejemplo:

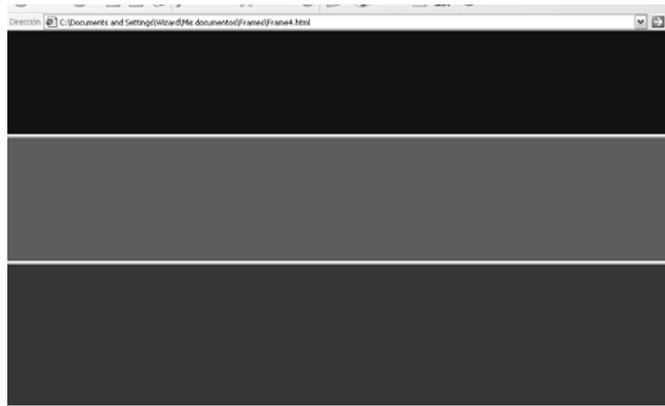
`<frameset rows = "25%,30%,*">`

Con esto dividiremos la ventana en tres filas, la primera división tendrá un 25% de la ventana del navegador, la segunda el 30%, y la tercera el resto de la ventana que es de un 50%. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame4

La estructura completa del código es la siguiente:

```
<html>
<head> <title> </title>
</head>
<frameset rows="25%,30%,*">
<frame src= "2.html">
<frame src="3.html">
<frame src="4.html">
</frameset>
</html>
```

A continuación en la figura I.43 se muestra la creación de dos o más frames.



1.43 Creación de dos o más Frames.

De igual manera también lo podemos realizar de forma vertical, visualizando más de dos columnas, esto se realiza con la instrucción:

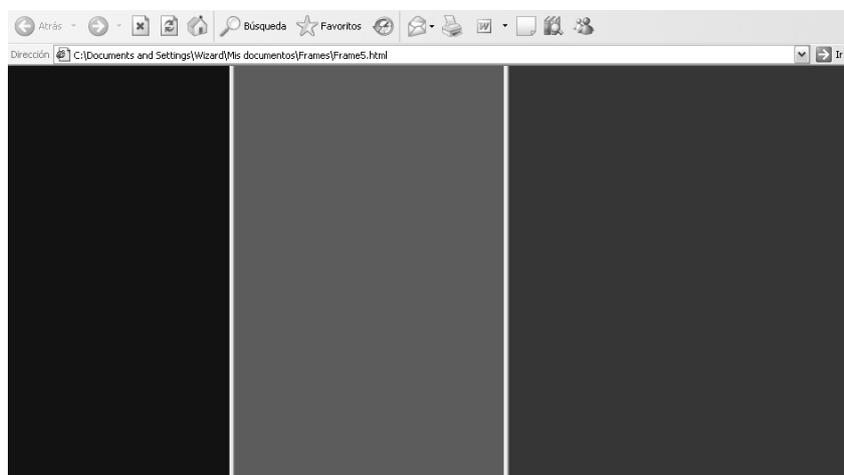
<frameset cols= "205,30%,*">

Con esto dividiremos la ventana en tres columnas, la primera división tendrá un 250 píxeles de la ventana del navegador, la segunda el 30%, y la tercera el resto de la ventana. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame5.

La estructura completa del código es la siguiente:

```
<html>
<head> <title> </title>
</head>
<frameset cols="250,30%,*">
<frame src= "2.html">
<frame src="3.html">
<frame src="4.html">
</frameset>
</html>
```

A continuación en la figura I.43 se muestra la creación de dos o más frames verticales:



I.43 Creación de dos o más frames verticales.



Unión de los Atributos cols y rows.

La etiqueta <frameset> puede tener al mismo tiempo los atributos rows y cols. Ejemplo:

```
<frameset rows= "100%" cols= "30%,*">  
.....  
</frameset>
```



Asignación de Nombre a Frames.

Si queremos colocar más de tres frame debemos nombrar a los distintos frames, a través del atributo "name" dentro de la etiqueta <frame>. De esta manera el navegador sabrá el nombre de cada uno de los frames y mandará a llamar solo al frame que necesite utilizar. Ejemplo:

```
<frame name=alto src= "1.html">
```

Esta instrucción indica al navegador que asigne al frame el nombre alto y muestre el archivo 1.html.

Para poder acceder a un frame desde un hipervínculo, es preciso que tenga asignado un nombre. Para colocar los nombres en HTML de los frames se tienen algunas reglas:

- ✓ Los nombres tienen que empezar con caracteres alfabéticos (a-z ò A-Z), y no deben contener más de 71 caracteres. Pero se aconseja que se utilicen nombres cortos.
- ✓ En algunos navegadores el nombre debe comenzar con un guión de subrayado.



Estructura de los Hipervínculos en el Frame.

La idea de introducir hipervínculos dentro del frame, es para poder visualizar en un frame todos los enlaces que se pueden visitar, y en otra ventana poder ver la página completa, dentro del mismo frame. Para hacer esto se necesita del atributo TARGET, con este se puede indicar al navegador en que frame deseamos que se visualice la página que deseamos ver. Su estructura es la siguiente:

```
<a href="G:\HTML\5ºClase\pagina1.html" TARGET="central" >Clase 4</a>
```

Con este hipervínculo podemos decirle que el enlace se ha direccionado al frame nombrado como central. Hay que mencionar que si no indicamos donde sea visualizada la página, lo hará dentro de la misma página en que se encuentra colocado el enlace, lo cual hará que la presentación de la página Web, se vea mal diseñada. . Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame6.

Un ejemplo de frames con hipervínculos es el siguiente:

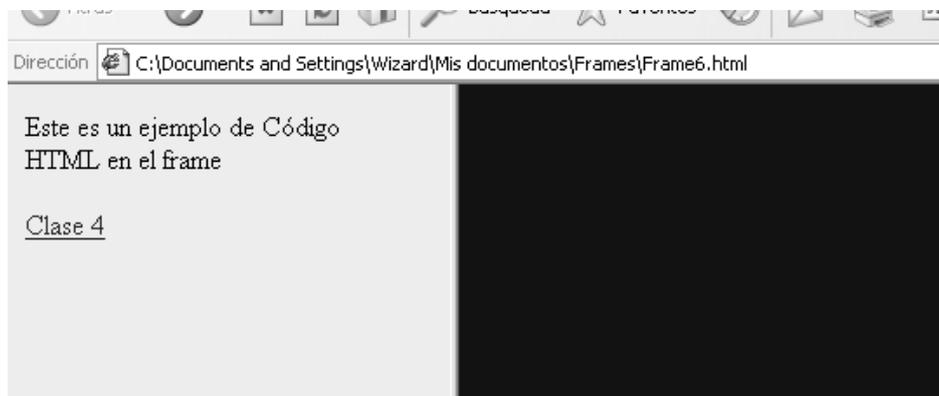
❖ Archivo HTML

```
<html>
<head>
<title> </title>
</head>
<body bgcolor="yellow">
<a href="G:\HTML\5ºClase\pagina1.html" TARGET="central" >Clase 4</a>
</body>
</html>
```

❖ Archivo de Frame

```
<html>
<head> <title> </title>
</head>
<frameset rows="25%,*">
<frame src= "C:\Documents and Settings\Wizard\Mis documentos\Frames\1.html">
<frame src="C:\Documents and Settings\Wizard\Mis documentos\Frames\2.html">
</frameset>
</html>
```

Si nosotros colocamos en el documento que vamos a visualizar este enlace y lo visualizamos en el explorador podemos verlo como se muestra a continuación en la figura I.44:



I.44 Hipervínculo dentro de un Frame.

Existen algunas propiedades del TARGET predefinidas ya, que pueden resultar muy útiles como:

- ✓ **_blank** : El documento se carga en una nueva ventana.
- ✓ **_self** : El documento se carga en el mismo frame, en que se encuentra el enlace.
- ✓ **_parent** : El documento se carga en donde se encuentra el enlace.
- ✓ **_top** : El documento se carga en la ventana original eliminado los frames.
- ✓ **_new** : El documento se carga en una nueva ventana.

Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame7.

Un ejemplo de esto lo podemos ver a continuación:

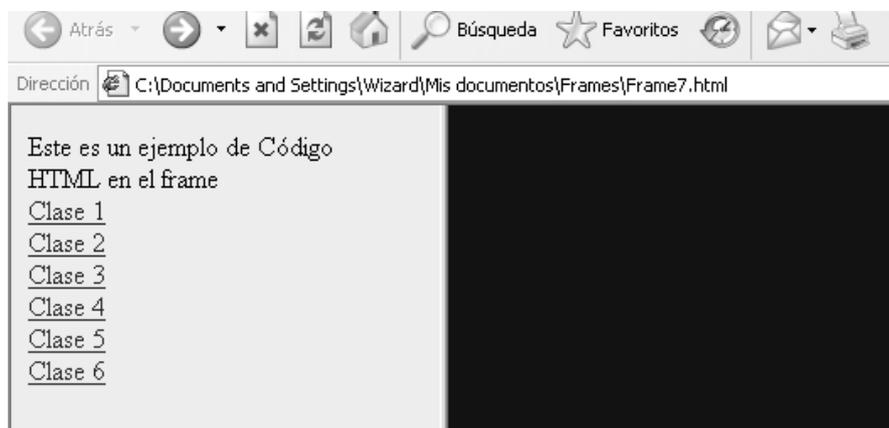
❖ Documento HTML

```
<html>
<head>
<title> </title>
</head>
<body bgcolor="yellow">
Este es un ejemplo de Código HTML en el frame
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="_blank" >Clase 1</a>
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="_self" >Clase 2</a>
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="_parent" >Clase 3</a>
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="_top" >Clase 4</a>
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="_new" >Clase 5</a>
<br>
<a href="H:\HTML\5ºClase\pagina1.html" TARGET="central" >Clase 6</a>
</body>
</html>
```

❖ Documento del Frame

```
<html>
<head> <title> </title>
</head>
<frameset cols="25%,*">
<frame name="sx" src=" 5.html">
<frame name="central" src=" 2.html">
</frameset>
</html>
```

A continuación en la figura I.45 se muestra la utilización de hipervínculos con propiedades predefinidas:



I.45 Hipervínculo con Propiedades Predefinidas.

Trabajar con Frames sin Bordes.

Cuando trabajamos con frames tenemos la posibilidad de trabajar sin bordes y sin barras de desplazamiento. Con un diseño bien planeado del color de fondo podemos lograr páginas visualmente fascinantes.

Este control del frame sobre los bordes y barras de desplazamiento se lleva a cabo dentro de las etiquetas `<frameset>` y `<frame>`, usando el atributo `frameborder`, e indicándole "no", para decirle al navegador que no deseamos que ponga los bordes. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame9. Ejemplo:

```
<html>
<head>
<title>Uso de NoFrames</title>
</head>
<frameset cols="25%,*" FRAMEBORDER="yes">
<frame name="sx" src=" 5.html">
<frame name="central" src2.html">
</frameset>
<noframe>
<body>
<h2>Su navegador no soporta frames</h2>
<br>
Pulse aquí: <a href="5.html">Contenido</a>
</body>
</noframe>
</html>
```

Inhabilitar Cambios del Tamaño del Frame por el Usuario.

El atributo `<noresize>` provoca que el usuario no pueda adaptar el tamaño del frame a su preferencia. En donde si nosotros no colocamos este atributo, cuando el usuario se posiciona dentro de un borde, podrá modificarlo, ya que aparece una flecha para hacerlo `<noresize>`, elimina esta acción imposibilitando el cambio de tamaño en el frame. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame10.

Su estructura es la siguiente:

```
<html>
<head>
<title>Uso de NoFrames</title>
</head>
<frameset cols="25%,*" FRAMEBORDER="yes">
<frame name="sx" src=" 5.html"noresize>
<frame name="central" src=" 2.html" noresize>
</frameset>
<noframe>
<body>
<h2>Su navegador no soporta frames</h2>
<br>
Pulse aquí: <a href=" 5.html">Contenido</a>
</body>
</noframe>
```

</html>

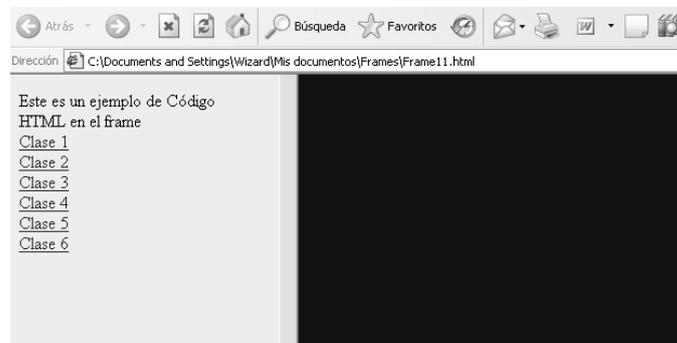
Tamaño del borde

En un frame podemos establecer el tamaño ancho que tendrá el borde del frame con el atributo border, el cual debe estar colocado dentro de la etiqueta <frameset>. Un ejemplo de esto es:

```
<html>
<head>
<title>Uso de NoFrames</title>
</head>
<frameset cols="25%,*" border=15>
<frame name="sx" src=" 5.html"noresize>
<frame name="central" src=" \2.html" noresize>
</frameset>
<noframe>
<body>
<h2>Su navegador no soporta frames</h2>
<br>
Pulse aquí: <a href=" 5.html">Contenido</a>
</body>
</noframe>
</html>
```

Mientras más grande sea el tamaño del borde, más grueso se mostrará en el navegador.

A continuación en la figura I.46 se muestra el tamaño del borde del frame:



I.46 Tamaño del Borde del Frame.

. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame11.

Color del Borde

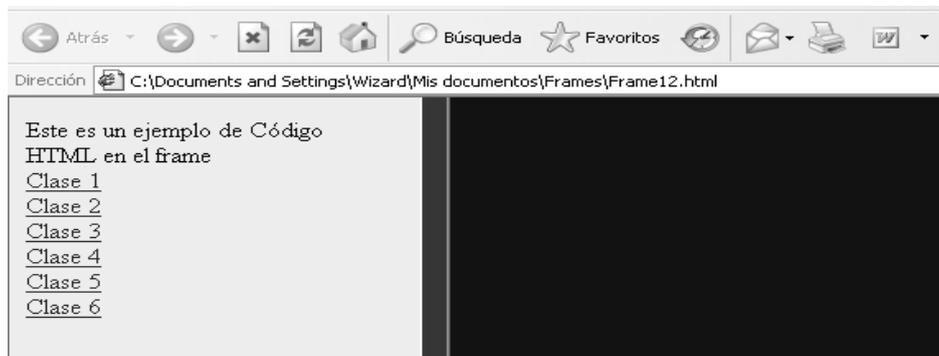
De igual manera que podemos modificar el ancho del borde, nosotros también podemos modificar el color del borde por medio del atributo BORDERCOLOR, el cual se define de igual manera que los colores de fondo del navegador antes mencionados. Este ejercicio, lo podemos ver en la carpeta 6ºClase, frame12. Un ejemplo de este atributo es el siguiente:

```

<html>
<head>
<title>Uso de NoFrames</title>
</head>
<frameset cols="25%,*" border=15 bordercolor="Red">
<frame name="sx" src=" 5.html"noresize>
<frame name="central" src=" 2.html" noresize>
</frameset>
<noframe>
<body>
<h2>Su navegador no soporta frames</h2>
<br>
Pulse aquí: <a href="5.html">Contenido</a>
</body>
</noframe>
</html>

```

A continuación en la figura I.47 se muestra el color del borde del frame:



I.47 Color del Border del Frame.

 Nota: hay que aclarar que así como podemos usar colores de fondo también podemos agregarle imágenes de fondo.

Algunos ejemplos de frames:

```

<html>
<head> <title> </title>
</head>
<frameset cols="120,*">
<frame name="sx" src="H:\HTML\6ºClase\sx.html">
<frameset rows="100,*">
<frame name="alto" src="H:\HTML\6ºClase\top.htm">
<frame name="central" src="H:\HTML\6ºClase\central.html">
</frameset>
</frameset>
</html>

```

A continuación en la figura I.48 se muestra la imagen de fondo del frame:



I.48 Imagen del Fondo del Frame

Este ejercicio, lo podemos ver en la carpeta 6ºClase, index2.

```

<html>
<head> <title> </title>
</head>
<frameset rows="100,*">
<frame name="alto" src="top.htm">
<frameset cols="150,*">

<frame name="sx" src=" sx.html">
<frame name="central" src=" central.html">
</frameset>
</html>

```

A continuación en la figura I.49 se muestra la imagen de fondo del frame, con etiquetas de HTML:



I.49 Imagen de Fondo del Frame con Etiquetas de HTML.

Este ejercicio, lo podemos ver en la carpeta 6ºClase, index3.

I.26 Multimedia

Sonido al abrir la página.

Un explorador puede reproducir un archivo de audio como fondo musical cuando se abre un documento Web. De este modo podemos dotar de efectos especiales a la página. Para realizar esto utilizamos la etiqueta <bgsound>, la cual es soportada por todos los navegadores. Su estructura es la siguiente:

```
<bgsound src="cancion.wav">
```

Los archivos de sonido pueden encontrarse en los formatos .WAV , .AU ó .midi(.mid).

Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina1. Un ejemplo de cómo integrar el sonido de fondo a nuestra página es el siguiente:

```
<html>
<head>
<title>Página con fondo musical</title>
<bgsound src="F:\Tania\HTML Prácticas\8ºClase\cancion.wav">
</head>
<body background="sakura.jpg">
<center><h1>Página con Fondo Musical</h1></center>
</body>
</html>
```

Se recomienda que la etiqueta <bgsound> se coloque dentro de la etiqueta <head>...</head>, debido a que casi por lo regular no se han muchas modificaciones al sonido, pero de igual manera puede colocarlo dentro del body, sin ningún problema.

El archivo de sonido se reproduce una vez al abrir el documento. Para hacer posible la reproducción múltiple, el tag <bgsound> tiene dos atributos adicionales los cuales son:

```
<bgsound src="cancion.wav" loop= "infinitivo" delay= "10">
```

Donde:

- ✓ **loop:** Con este atributo especificamos cuántas veces se repite el archivo de sonido, mientras el documento se encuentra abierto. Con loop=1, el archivo se reproduce solo una vez. Podemos aumentar este número para hacer una reproducción múltiple. Aunque existe un valor predeterminado loop="infinite", el cual hace que el archivo de sonido se repita constantemente.
- ✓ **delay:** Este atributo retrasa unos segundos la reproducción del archivo de sonido después de cargar la página. Con delay = 3 el navegador esperará 3 segundos para reproducir el sonido.

Un ejemplo de esto es:

```
<html>
<head>
<title>Pagina con fondo musical</title>
```

```

<bgound src="cancion.wav" loop="infinite" delay="20">
</head>
<body background="sakura.jpg">
<center><h1>Página con Fondo Musical</h1></center>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina2.

Hay que recordar que para poder escuchar la canción necesitamos de la ayuda de los drivers necesarios en cada computadora para reproducir sonido.

A continuación en la figura I.50 se muestra la página con fondo musical:



I.50 Página con Fondo Musical.

Integrar un Archivo de Sonido Externo.

Actualmente existen muchas páginas Web desde las que podemos descargar y reproducir música. En un documento HTML podemos definir un hipervínculo a un archivo de sonido externo.

En caso de que integremos un hipervínculo a un archivo de sonido externo, en el documento HTML, al elegir este hipervínculo el navegador puede descargar o abrir este archivo.

La estructura para hacerlo es la siguiente:

```

<html>
<head>
<title>Pagina con fondo musical</title>
</head>
<body background="sakura.jpg">
<center><h1>Enlace a un archivo musical</h1></center>
<br>
<br>
<h2><a href=" Hero.mp3">Canci&ocute;n Heroe</a></h2>

```

</body>
</html>

Una vez que nosotros seleccionamos el enlace que nos muestra, el archivo se comenzará a reproducir en el reproductor que tengamos en la computadora.

A continuación en la figura I.51 se muestra un enlace a un archivo musical:



I.51 Enlace a un Archivo Musical.

Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina3.

Para reproducir un archivo de alta calidad, recientemente se usa el formato MPEG-Audio ó MP3. Este formato se utiliza con frecuencia para reproducir archivos musicales en Internet, que también son soportados por la etiqueta <bgsound>.

Integrar Archivos de Video.

Para integrar un archivo de video en un documento HTML, utilizamos el tag , su estructura es la siguiente:

Donde:

- ✓ **dynscr** : Define el nombre y la ruta donde se encuentra el video así como su extensión.
- ✓ **start**: Determina cuándo debe empezar el navegador a reproducir el video. Se permiten algunos valores:
 - ❖ **fileopen**: Establece que la reproducción comience inmediatamente después de abrir el documento HTML.
 - ❖ **mouseover**: Este atributo indica que el archivo de video debe reproducirse en cuanto el puntero del ratón, apunte al identificador del lugar.

Un ejemplo de esto es:

```

<html>
<head>
<title>Página con fondo musical</title>
</head>
<body background="F:\Tania\HTML Prácticas\8ºClase\sakura.jpg">
<center><h1>Enlace a un archivo musical</h1></center>
<br>
<br>

</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina4.

A continuación en la figura 1.52 se muestra la página con fondo musical y sus propiedades predefinidas :



1.52 Página con Fondo Musical y sus Propiedades Predefinidas.

Opciones para mostrar el archivo de video.

Un archivo de video solo se reproduce una vez. No obstante podemos indicarle con el atributo loop, un contador de repetición. Ejemplo:

```
<img dynsrc= "video.avi" start= "fileopen" loop= "3">
```

El archivo de video se carga al abrir el documento y se reproduce un total de 3 veces. Si utilizamos el valor loop= "infinitivo", el navegador repetirá el video de manera continua. Si deseamos cambiar las dimensiones del archivo de video se utilizan los atributos width y height. Ejemplo:

```
<img dynsrc= "video.avi" start= "fileopen" loop= "3" alt= "video" border = "0"
width= "180" height= "180">
```

Algunos atributos adicionales son:

- ✓ **border** : Controla si el archivo de video debe mostrarse con o sin borde, al ser integrado en un hipervínculo.
- ✓ **atl** : Muestra un texto como un identificador de lugar.

Debemos tener en cuenta que cuando visualizamos un video desde Internet, lleva mucho tiempo poder descargarlos para poderlos visualizar correctamente. Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina5.

Integrar un Archivo de Video Externo.

Podemos integrar un archivo de video en la página por medio de un hipervínculo. Cuando abramos la página, nos mostrará en enlace para ver el video. Un ejemplo de esto es:

```
<html>
<head>
<title>Página con fondo musical</title>
</head>
<body background="./sakura.jpg">
<center><h1>Enlace a un archivo musical</h1></center>
<br>
<br>
<h2><a href="./video.avi">Video</a></h2>
</body>
</html>
```

A continuación en la figura 1.53 se muestra la integración de video externo :



1.53 Integración de Video Externo.

Para guardar archivos de video se disponen de diferentes formatos, los cuales se utilizan de forma distinta dependiendo de la plataforma y del fin, para que lo usen. Este ejercicio, lo podemos ver en la carpeta 8ºClase, pagina5.

I.27 Creación de Tablas.

Las tablas no solo las podemos emplear para poder tabular datos; son también, para lograr un diseño llamativo. Como ya hemos mencionado antes HTML no es un lenguaje de diseño de páginas, sino de documentos, sin embargo esto no debe ser un obstáculo para nosotros y poder lograr un diseño llamativo con la colaboración del uso de tablas.

Etiquetas para Construir Tablas.

Para crear una tabla debemos usar siempre las etiquetas `<table>.....</table>`, en las cuales estará contenida nuestra tabla. Dentro de la etiqueta `<table>` existen diferentes atributos que permiten darle formato a la tabla. Algunos de ellos son:

- ✓ **`<tr>.....</tr>`**: Con este atributo podemos definir las filas de la tabla, mientras más `<tr>` se coloquen, mayor será el número de filas. Indica donde inicia una línea de la tabla y donde terminará con su etiqueta de cierre.
- ✓ **`<td>.....</td>`**: Con este atributo podemos configurar las celdas de la tabla, estas celdas pueden contener cualquier elemento HTML: texto, listas, imágenes y ¿por que no? otra tabla.

Distancias entre Celdas y Contornos.

Una tabla tiene los siguientes atributos:

- ❖ **border**
- ❖ **cellpadding**
- ❖ **cellspacing**

- ✓ **border**: Con este atributo decidimos si la tabla llevará un borde o no. Ejemplo:
`<table border= "1">`

Si decidimos que la tabla no debe llevar borde se debe colocar `border=0`, para que no aparezcan las líneas.

- ✓ **cellspacing**: Este atributo permite aumentar o reducir la distancia entre cada celda. Ejemplo:

`<table border= "1" cellspacing= "0">`

A medida que se defina el atributo con valores más grandes, el navegador separará las celdas cada vez más.

- ✓ **cellpadding**: Define el espacio en píxeles entre el borde y el texto. Ejemplo:

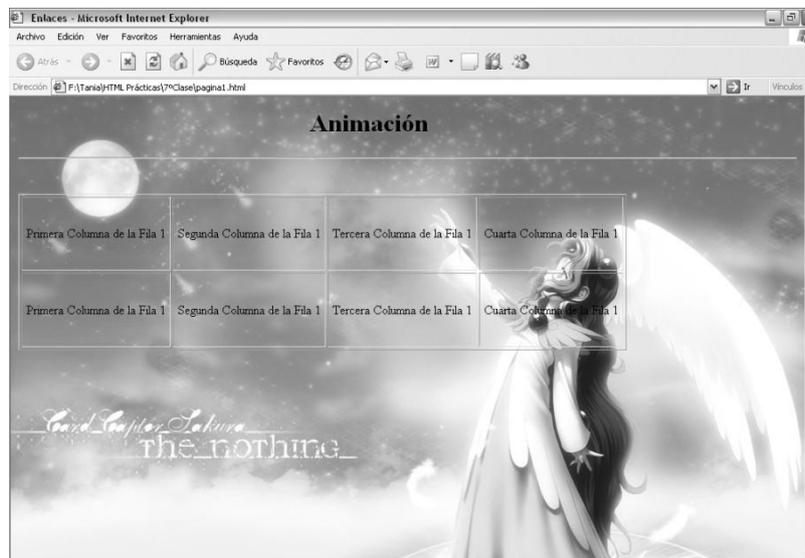
`<table border = "1" cellpadding= "4">`

Con los atributos **width** y **height** podemos forzar a la tabla a ocupar un cierto porcentaje de la anchura o altura de la ventana del navegador o un número de píxeles determinado.

Un ejemplo de cómo utilizar estas etiquetas y atributos en una tabla es el siguiente:

```
<html>
<head>
<title>Enlaces</title>
</head>
<body background="sakura.jpg">
<h1><marquee> Animaci&oacute;n</marquee></h1>
<hr>
<br>
<br>
<table widht=100%" height="200" border="1" cellspacing="3" cellpadding="5">
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
</table>
</body>
</html>
```

A continuación en la figura I.54 se muestra la creación de tabla :



I.54 Creación de Tabla

Este ejemplo lo podemos ver en 7ºClase, pagina1.

 **NOTA:** El valor de todos estos atributos suele especificarse en píxeles.

Atributos de la Etiqueta <table>

La etiqueta <table> contiene atributos los cuales permiten dar una mejor presentación a la tabla estos son:

- ✓ **valign:** Este atributo proporciona la alineación vertical de todo el contenido de la celda. Donde este atributo admite los siguientes valores:
 - ❖ **top** (parte superior del navegador)
 - ❖ **bottom** (parte inferior del navegador)
 - ❖ **middle** (centrado vertical)

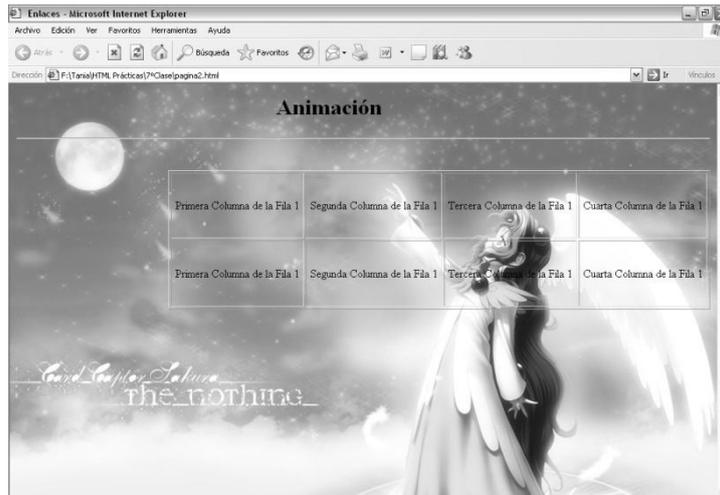
- ✓ **align:** Este atributo nos permite alinear de forma horizontal.
 - ❖ **right** (derecha)
 - ❖ **left** (izquierda)
 - ❖ **center** (centrado horizontal)

Un ejemplo de estos atributos es el siguiente:

```
<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7ºClase\muñeca.jpg">
<h1><marquee> Animaci&oacute;n</marquee></h1>
<hr>
<br>
<br>
<table width="100%" height="200" border="1" cellspacing="3" cellpadding="5"
align="right">
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
</table>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina2.

A continuación en la figura I.55 se muestra la alineación de tabla :



I.55 Alineación de Tabla

Visualización de Tablas

Las tabla en HTML, tienen la posibilidad de tener diferente formato, esto quiere decir, que no todas las filas o columnas deben tener el mismo tamaño. Este lo podemos hacer con los atributos **colspan** y **rowspan** dentro de la etiqueta <td>.

- ✓ **colspan**: Une dos celdas de dos columnas adyacentes en una sola celda.
- ✓ **rowspan**: Une celdas de filas adyacentes, en una única celda.

Un ejemplo de estos atributos es:

```
<html>
<head>
<title>Enlaces</title>
</head>
<body background="muñeca.jpg">
<h1><marquee> Animaci&oacute;n</marquee></h1>
<hr>
<br>
<br>
<table border="1" cellspacing="3" cellpadding="5">
  <tr>
    <td colspan=2>Primera Columna de la Fila 1</td>
    <td rowspan=3>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
</table>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta 7°Clase, pagina4.

A continuación en la figura I.56 se muestra la visualización de tabla :



I.56 Visualización de Tabla.

Uso de las Etiquetas **<th>** y **<caption>**

Estas etiquetas permiten darle mucha mayor presentación a las tablas.

- ✓ **<th>** : Se encarga de convertir el texto de cabecera de cada columna, el cual se centra automáticamente y se pone en letras negritas.
- ✓ **<caption>**: Este atributo permite poner el título a la tabla en la parte superior. Puede alinear el título de dos maneras con **align=top** y **align=bottom**.

Ejemplo del uso de estos dos atributos:

```
<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7°Clase\muñeca.jpg">
<h1><marquee> Calendario</marquee></h1>
<hr>
<br>
<br>
<table border="1" cellspacing="3" cellpadding="5">
  <caption align=top>Agosto</caption>
  <tr>
    <td >Lunes</td> <td >Martes</td> <td>Miercoles</td> <td>Jueves</td>
    <td >Viernes</td> <td>Sabado</td> <td>Domingo</td>
  </tr>
  <tr>
    <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> <td>6</td>
    <td>7</td>
  </tr>
  <tr>
    <td>8</td> <td>9</td> <td>10</td> <td>11</td> <td>12</td> <td>13</td>
```

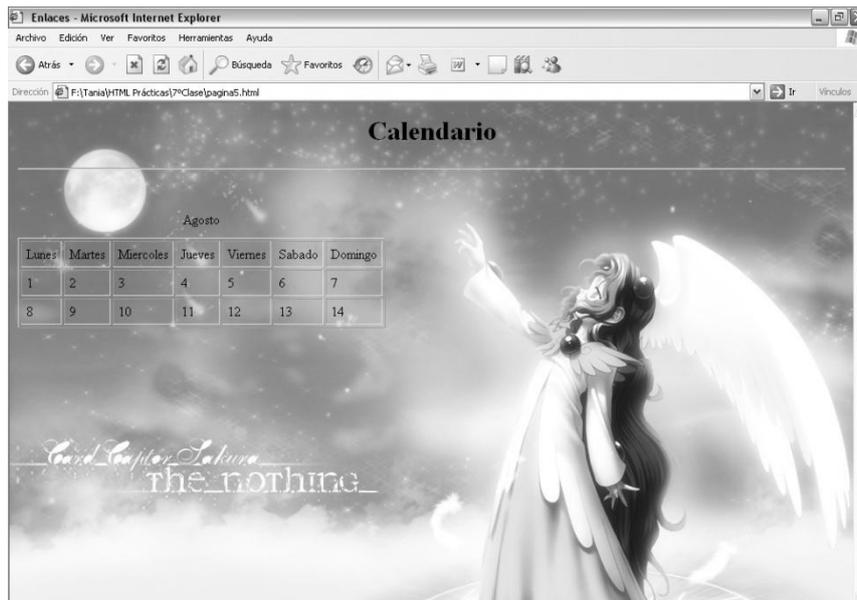
```

        <td>14</td>
      </tr>
    </table>
  </body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina5.

A continuación en la figura I.57 se muestra colocación de Título a la Tabla :



I.57 Colocación del Título a la Tabla.

Uso de Color en Tablas.

De la misma manera que podemos indicar el color de fondo de una página Web, podemos brincar el color de fondo para una tabla, una fila de la tabla o una sola celda, empleando el atributo bgcolor dentro de las etiquetas <table>, <tr> y <td>.

Para aplicar el color de fondo en la tabla necesitamos colocar el atributo bgcolor en la etiqueta <table>. Ejemplo:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="muñeca.jpg">
<h1><marquee> Animaci&oacute;n</marquee></h1>
<hr>
<br>
<br>
<table border="1" cellspacing="3" cellpadding="5" bgcolor="yellow">
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
  </tr>
</table>

```

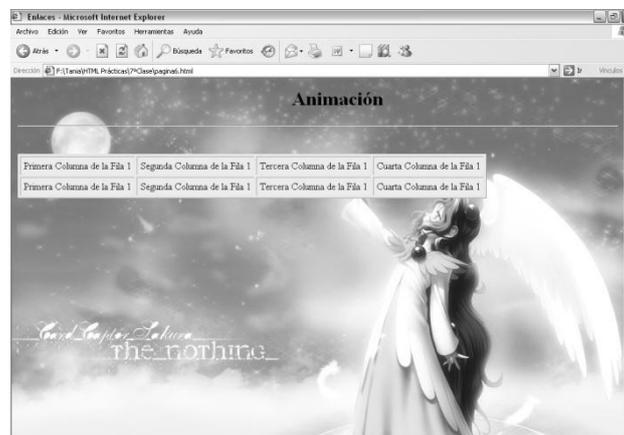
```

<td>Tercera Columna de la Fila 1</td>
<td>Cuarta Columna de la Fila 1</td>
</tr>
<tr>
<td>Primera Columna de la Fila 1</td>
<td>Segunda Columna de la Fila 1</td>
<td>Tercera Columna de la Fila 1</td>
<td>Cuarta Columna de la Fila 1</td>
</tr>
</table>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7°Clase, pagina6.

A continuación en la figura I.58 se muestra el color en Tabla :



I.58 Color en Tabla

Si nosotros deseamos colocar el color de fondo a una sola fila, se puede hacer indicándolo con la etiqueta <tr>, además del atributo bgcolor. Un ejemplo de esto es:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7°Clase\sugar.jpg">
<h1><marquee> Animaci&ocute;n</marquee></h1>
<hr>
<br>
<br>
<table border="1" cellspacing="3" cellpadding="5">
<tr bgcolor="yellow">
<td>Primera Columna de la Fila 1</td>
<td>Segunda Columna de la Fila 1</td>
<td>Tercera Columna de la Fila 1</td>
<td>Cuarta Columna de la Fila 1</td>
</tr>
<tr>
<td>Primera Columna de la Fila 1</td>
<td>Segunda Columna de la Fila 1</td>

```

```

        <td>Tercera Columna de la Fila 1</td>
        <td>Cuarta Columna de la Fila 1</td>
    </tr>
</table>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina7.

A continuación en la figura I.59 se muestra el color en una sola Fila de la Tabla :



I.59 Color en una sola Fila de la Tabla.

Pero si solo deseamos que solo sea una celda; se hace por medio de la etiqueta <td> seguido del atributo bgcolor. Un ejemplo de esto es:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7ºClase\sugar.jpg">
<h1><marquee> Animación</marquee></h1>
<hr>
<br>
<br>
<table border="1" cellspacing="3" cellpadding="5">
    <tr>
        <td bgcolor="yellow">Primera Columna de la Fila 1</td>
        <td>Segunda Columna de la Fila 1</td>
        <td>Tercera Columna de la Fila 1</td>
        <td>Cuarta Columna de la Fila 1</td>
    </tr>
    <tr>
        <td>Primera Columna de la Fila 1</td>
        <td>Segunda Columna de la Fila 1</td>
        <td>Tercera Columna de la Fila 1</td>
        <td>Cuarta Columna de la Fila 1</td>
    </tr>
</table>

```

</body>
</html>

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina8.

A continuación en la figura I.60 se muestra el color en una sola celda de la Tabla :



I.60 Color en una sola celda de la tabla.

Bordes y Líneas en la Tabla.

El atributo border como habíamos dicho anteriormente indica el grosor del contorno de la tabla. Además de este atributo existe el atributo frame, el cual se encarga de trazar cuatro líneas exteriores alrededor de la tabla. Este se debe colocar dentro de la etiqueta <table>.

El frame puede contener los siguientes atributos:

- ✓ **void** : Elimina el borde que rodea la tabla (equivalente a border=0)
- ✓ **box**: Traza un borde alrededor de la tabla, en sus cuatro lados.
- ✓ **above**: Traza una línea en la parte superior de la tabla
- ✓ **below** : Traza una línea en la parte inferior de la tabla.
- ✓ **hsides**: Traza una línea en la parte superior e inferior de la tabla
- ✓ **lhs** : Traza la línea en el lado izquierdo de la tabla.
- ✓ **rhs** : Traza la línea en el lado derecho de la tabla.
- ✓ **rules**: Este atributo establece cómo debe mostrarse las líneas divisoras que hay entre cada celda.

El atributo rules tiene a su vez otros atributos:

- ❖ **none** : Elimina las líneas dentro de la tabla.
- ❖ **rows** : Traza las líneas entre filas.
- ❖ **cols** : Traza las líneas entre las columnas.
- ❖ **all** : Traza las líneas alrededor de todas las celdas de la tabla.

Un ejemplo de la utilización de estos atributos es:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7ºClase\sugar.jpg"><!--Fondo de la
página -- >
<h1><marquee> Animaci&oacute;n</marquee></h1>
<hr><br><br>
<table border="1" cellspacing="3" cellpadding="5" frame= "void">
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
  <tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
</table>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina9.

A continuación en la figura I.61 se muestra el color en una sola celda de la Tabla :



I.61 Color en una sola celda de la tabla.

Ejemplo del uso del atributo rules:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7ºClase\sugar.jpg">
<h1><marquee> Animaci&oacute;n</marquee></h1>

```

```

<hr><br><br>
<table border="1" cellspacing="3" cellpadding="5" rules="none">
  <tbody>
    <tr>
      <td>Primera Columna de la Fila 1</td>
      <td>Segunda Columna de la Fila 1</td>
      <td>Tercera Columna de la Fila 1</td>
      <td>Cuarta Columna de la Fila 1</td>
    </tr><tr>
      <td>Primera Columna de la Fila 1</td>
      <td>Segunda Columna de la Fila 1</td>
      <td>Tercera Columna de la Fila 1</td>
      <td>Cuarta Columna de la Fila 1</td>
    </tr></tbody>
</table>
</body>
</html>

```

 **NOTA:** La etiqueta <tbody> es necesaria si se utiliza el atributo rules.

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina10.

A continuación en la figura I.62 se muestra el color uso del atributo rules :



I.62 Uso del atributo rules

Tablas con Imágenes de Fondo

Para asignar una imagen de fondo a una tabla debemos incorporar el atributo background en lugar del atributo bgcolor en la etiqueta <table>. Este atributo recibe la dirección de la imagen y su extensión correspondiente.

Ejemplo:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="sugar.jpg">
<h1><marquee> Animaci&ocute;n</marquee></h1><hr><br><br>

```

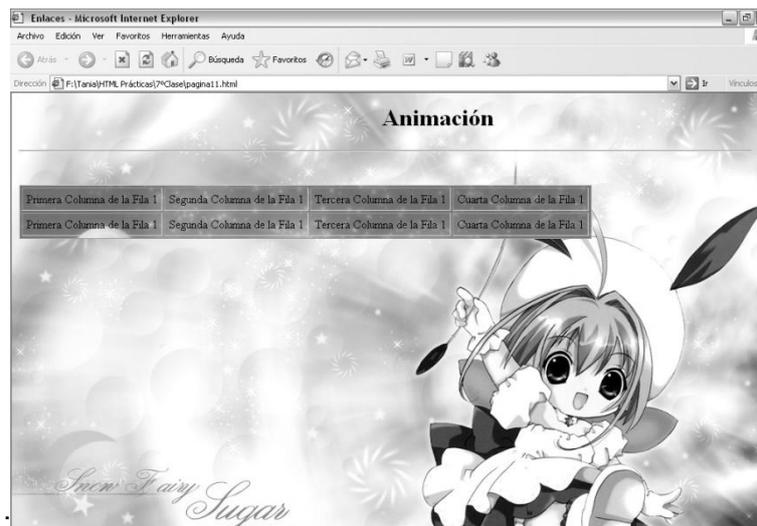
```

<table border="1" cellspacing="3" cellpadding="5" background="muñeca.jpg">
  <tbody>
    <tr>
      <td>Primera Columna de la Fila 1</td>
      <td>Segunda Columna de la Fila 1</td>
      <td>Tercera Columna de la Fila 1</td>
      <td>Cuarta Columna de la Fila 1</td>
    </tr><tr>
      <td>Primera Columna de la Fila 1</td>
      <td>Segunda Columna de la Fila 1</td>
      <td>Tercera Columna de la Fila 1</td>
      <td>Cuarta Columna de la Fila 1</td>
    </tr>
  </tbody>
</table>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7°Clase, pagina11.

A continuación en la figura I.63 se muestra la tabla con imagen de fondo.



I.63 Tabla con imagen de fondo

Si es preciso asignar una imagen de fondo para una sola celda, podemos insertarlo en la etiqueta <td>. Ejemplo:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="sugar.jpg">
<h1><marquee> Animación</marquee></h1>
<hr><br><br>
<table border="1" cellspacing="3" cellpadding="5">
  <tr >
    <td background="muñeca.jpg">Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>

```

```

<td>Tercera Columna de la Fila 1</td>
<td>Cuarta Columna de la Fila 1</td>
</tr><tr>
<td>Primera Columna de la Fila 1</td>
<td background="sakura.jpg">Segunda Columna de la Fila 1</td>
<td>Tercera Columna de la Fila 1</td>
<td>Cuarta Columna de la Fila 1</td></tr>
</table>
</body>
</html>

```

A continuación en la figura I.64 se muestra la tabla con imagen de fondo para una celda



I.64 Imagen de fondo para una celda.

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina12.

Modificar los Colores de las Líneas de la Tabla.

Para modificar el color de las líneas de una tabla debemos insertar el atributo bordercolor en el tag <table>, siguiendo las mismas reglas de color de fondo y letra. Su estructura es la siguiente:

```
<table bordercolor= "blue">
```

Con este atributo solo podemos cambiar el color del borde exterior, así como los bordes las celdas y filas.

Ejemplo:

```

<html>
<head>
<title>Enlaces</title>
</head>
<body background="F:\Tania\HTML Prácticas\7ºClase\sugar.jpg">
<h1><marquee> Animación</marquee></h1>
<hr><br><br>

```

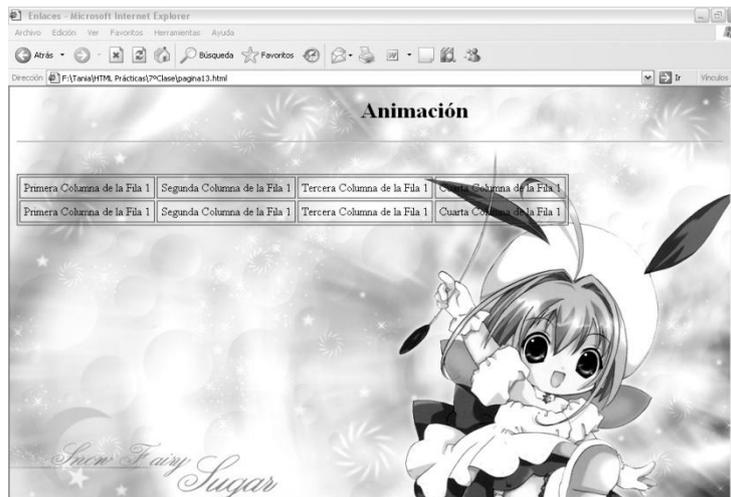
```

<table border="1" cellspacing="3" cellpadding="5" bordercolor="red">
  <tr >
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr><tr>
    <td>Primera Columna de la Fila 1</td>
    <td>Segunda Columna de la Fila 1</td>
    <td>Tercera Columna de la Fila 1</td>
    <td>Cuarta Columna de la Fila 1</td>
  </tr>
</table>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta 7ºClase, pagina13.

A continuación en la figura I.65 se muestra la tabla con imagen de fondo para una celda.



I.65 Tabla con imagen de fondo para una celda.

Cambiar el color de las líneas para la representación 3D.

Con el atributo bordercolordark es posible variar el color de la parte oscura del borde. El atributo bordercolorlight, se encarga de modificar la parte clara del borde. Ejemplo:

```

<table border= "10" bordercolorlight= "blue">
*****
</table>

```

I.28 Formularios

Los formularios están formados por determinados elementos como botones, cuadros de texto, etc., colocados en un documento HTML. Si un usuario abre la página, se mostrarán todos los elementos que se encuentran en el formulario. A través de un botón podemos enviar los datos que introducimos en el formulario a un servidor.

Los datos introducidos en el formulario y enviados al servidor pueden tratarse de maneras diferentes.

- ✓ Los datos introducidos en el formulario pueden reenviarse a una dirección de correo electrónico.
- ✓ Los datos introducidos pueden ser editados en el servidor mediante un script(javascript).
- ✓ Podemos colocar los datos en una base de datos o guardarlos en archivo de texto en el mismo servidor.

El tag <form>

El tag <form> se usa para iniciar un formulario en una página HTML, además contiene su etiqueta de cierre </form>, lo cual indica que todo lo que se encuentre dentro de estas dos etiquetas será un formulario.

Dentro de esta etiqueta se encuentran todos los controles que se pueden tener de los datos.

Un documento HTML puede contener varias secciones de <form>. Sin embargo no es posible anidar tags <form>.

Dentro del tag <form> se encuentra el atributo **action**, el cual establece lo que debe ocurrir con los datos introducidos en el formulario. Dentro de este atributo existen dos variantes las cuales son:

- ✓ **http** : este atributo indica la dirección a donde se mandarán los datos. Ejemplo:

<form action= http://www.yahoo.com.mx>

- ✓ **mailto**: Se indica una dirección de correo electrónico a la que debe enviarse los datos introducidos.

<form action= "mailto:wizard_pag@yahoo.com.mx">

El segundo atributo que debe indicarse dentro del tag <form>, es **method**. Este método define el método con el que se transfieren y procesan los datos de formulario. Existen dos opciones:

- ✓ **Get**: Con este método el servidor transfiere los datos del formulario al script, mostrando los datos introducidos a toda persona.
- ✓ **Post**: El servidor transfiere los datos del formulario al script, de una manera oculta para los usuarios.

El atributo **value** presenta un texto fijo en los cuadros de texto u objetos de un formulario.

<input type= "submit" name= "Enviar" value= "Enviar">

Cuadros de Texto.

Los cuadros de texto comúnmente utilizan controles. Estos controles se insertan a través de la etiqueta <input> en el formulario dentro de la etiqueta <form>. Para la etiqueta <input> se pueden generar varios controles los cuales explicaremos uno a

uno. Por esta razón dentro de esta etiqueta debemos especificar que vamos a utilizar un cuadro de texto. Para generar un cuadro de texto la instrucción es la siguiente:

<input type= "text" name= "apellido">

Donde:

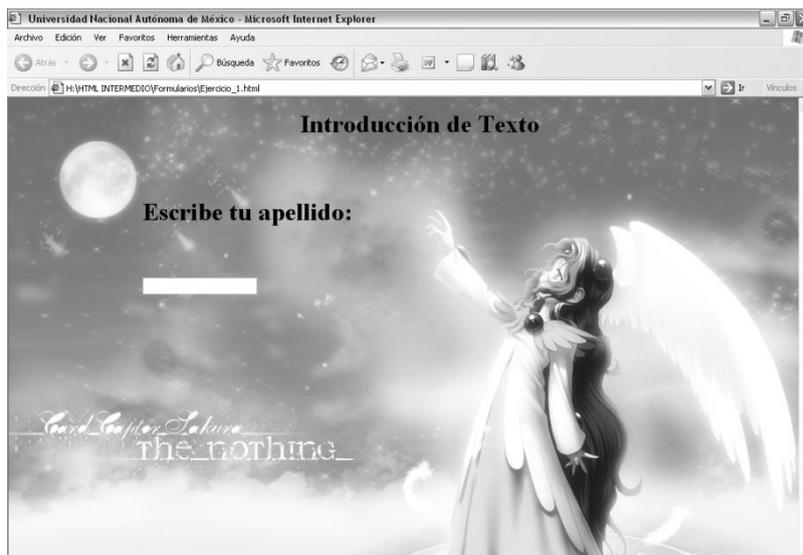
- ✓ **text** : En el atributo type, indica al navegador que debe mostrar un cuadro.
- ✓ **name** : Este atributo se utiliza para dar un nombre, con el cual administraremos los controles.

El ejemplo completo sería:

```
<html>
<head>
<title>
Universidad Nacional Autónoma de México
</title>
</head>
<body background="princesa.jpg">
<h1><b><marquee> Introducción de Texto </marquee></h1><b>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<FORM ACTION= "1" METHOD="POST" ENCTYPE="TEXT/PLAIN">
<br>
<br>
<br>
<h1>Escribe tu apellido:<h1>
<BR><INPUT TYPE="text" NAME="Apellido">
</FORM>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina1.

A continuación en la figura I.66 se muestra un cuadro de texto.



I.66 Cuadro de Texto.

Con estas instrucciones se puede dar un cuadro de texto de un tamaño específico pero a través del atributo **size** es posible configurar la longitud del cuadro. Ejemplo:

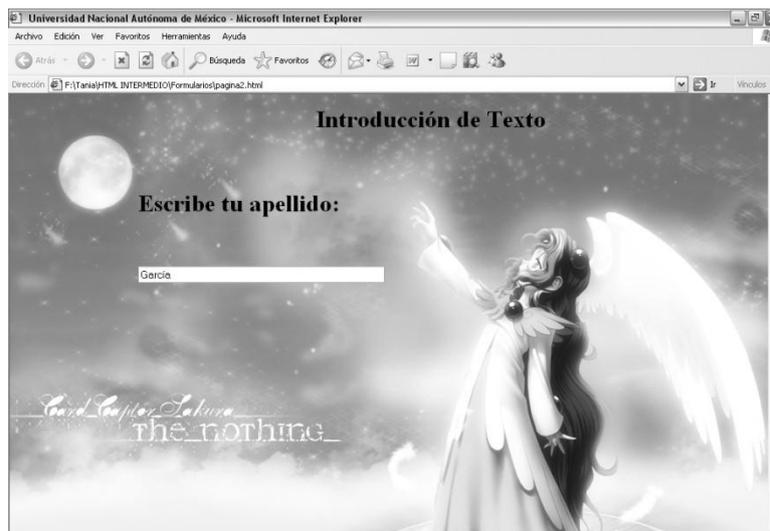
<input type= "text" name= "apellido" size="20" value= "García">

La estructura completa es la siguiente:

```
<html>
<head>
<title>
Universidad Nacional Aut&ocaronoma de M&eacute;xico
</title>
</head>
<body background="princesa.jpg">
<h1><b><marquee> Introducción de Texto </marquee></h1><b>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<FORM ACTION="mailto:wizard_pag@yahoo.com.mx" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<br><br><br>
<h1>Escribe tu apellido:<h1>
<BR><INPUT TYPE="text" NAME="Apellido" size="50" value="Garc&iacute;a">
</FORM>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina2.

A continuación en la figura I.67 se muestra un cuadro de texto con tamaño específico:



I.67 Cuadro de texto con tamaño específico.

A través del atributo opcional `maxlength`, podemos indicar cuantos caracteres podemos insertar en un cuadro de texto. Si no se utiliza y el texto introducido es más grande que el cuadro de texto, el navegador irá desplazando el texto por el cuadro. La estructura de este atributo es:

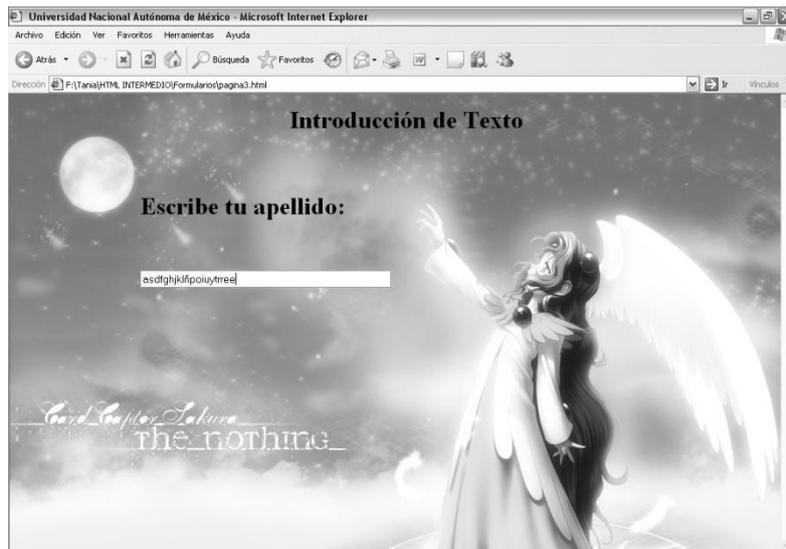
```
<input type= "text" name= "apellido" size="20" value= "García"
      maxlength= 20>
```

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="princesa.jpg">
<h1><b><marquee> Introducción de Texto </marquee></h1><b>
<blockquote><blockquote><blockquote><blockquote>
<FORM ACTION="mailto:wizard_pag@yahoo.com.mx" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<br><br><br>
<h1>Escribe tu apellido:<h1>
<BR><INPUT TYPE="text" NAME="Apellido" size="50" value="García"
maxlength= 20>
</FORM>
</blockquote>
</blockquote>
</blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina3.

A continuación en la figura I.68 se muestra un cuadro de texto con número específico de caracteres:



I.68 Cuadro de texto con número específico de caracteres

Petición de Contraseña en el Formulario.

Una de las ventajas de utilización de cuadros de texto, es la poder pedir contraseñas. Con el ejemplo anterior podemos ver que todo lo que escribimos dentro del cuadro de texto lo podemos visualizar sin ningún problema. Esto lo podemos modificar de tal manera que cuando el usuario introduzca el texto dentro del cuadro, en vez de visualizar su texto, solo veamos un asterisco (*), por cada carácter que escriba.

Para realizar esto utilizamos también la etiqueta `<input>`. Con una pequeña modificación dentro del atributo **type**, en vez de escribir **text**, ahora escribiremos la palabra **password**, la cual dará la señal al navegador que por cada carácter introducido, pondrá un asterisco. Ejemplo:

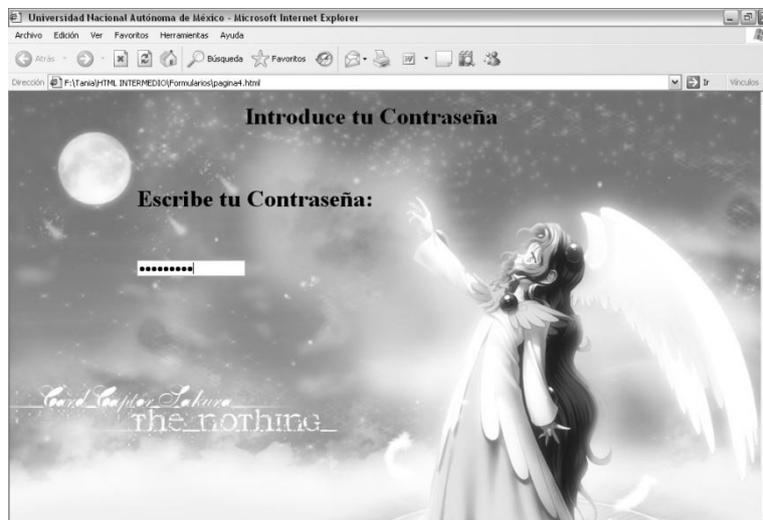
```
<input type= "password" name= "apellido">
```

La estructura completa es la siguiente:

```
<html>
<head>
<title>
Universidad Nacional Autónoma de México
</title>
</head>
<body background="princesa.jpg">
<h1><b><marquee> Introduce tu Contraseña</marquee></b></h1><b>
<blockquote><blockquote><blockquote><blockquote>
<FORM ACTION="mailto:wizard_pag@yahoo.com.mx" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<br><br><br>
<h1>Escribe tu Contraseña:</h1>
<BR><INPUT TYPE="password" NAME="Contraseña">
</FORM>
</blockquote></blockquote></blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina4.

A continuación en la figura I.69 se muestra un cuadro con petición de contraseña:



I.69 Cuadro con petición de contraseña.

Cuadros de Texto Multilínea.

Un ejemplo de esto es:

```
<textarea name = "Comentarios" cols = "40" rows= "5">
```

Donde:

- ✓ **name:** Establece el nombre de se utilizará para el control del formulario
- ✓ **cols :** Indica el ancho que tendrá el cuadro de texto, además del número de caracteres máximos.
- ✓ **rows :** Establece el número de líneas que podrán introducirse.

La estructura completa es la siguiente:

```
<html>
<head>
<title>
Universidad Nacional Autónoma de México
</title>
</head>
<body background="princesa.jpg">
<font color=black><h1><b><marquee>Comentarios en la
Página</marquee></h1><b></font>
<blockquote><blockquote><blockquote><blockquote>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
Introduce tus comentarios:
<br><BR><TEXTAREA NAME="Comentarios" ROWS="6" COLS="40">
</TEXTAREA>
```

```

</FORM>
</blockquote></blockquote></blockquote></blockqoute></body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina5.

A continuación en la figura I.70 se muestra un cuadro de texto multilinea:



I.70 Cuadro de Texto Multilinea.

Casillas de Verificación.

Para poder seleccionar algunas de las opciones que nosotros tengamos definidas dentro del formulario, existan las casillas de verificación. En las cuales el usuario solo tiene que seleccionar con el ratón la opción deseada.

Cuando mandemos la información del formulario al servidor, la casilla que se encuentre marcada, el navegador transfiere al servidor el nombre que tenga en el atributo value. En caso de no encontrarse seleccionada manda una cadena vacía. Para insertar las casillas de verificación en el formulario la estructura es la siguiente:

```
<input type= "checkbox" name= "Cuadro" value= "1" checked>Azul
```

Donde:

- ✓ **type:** Define la palabra checkbox, la cual indica que serán casillas de verificación.
- ✓ **name:** Asigna el nombre a la casilla
- ✓ **checked:** Si queremos que una de las casillas de verificación se encuentre activada a la hora de abrir la página se utiliza esta propiedad.
- ✓ **value:** Establece el valor que debe devolver el servidor a una silla de verificación marcada.

La estructura completa es la siguiente:

```

<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="princesa.jpg">
<center><h1><b>Formulario de Casillas de Verificación</b></h1><b></b></center>

```

```

<blockquote><blockquote><blockquote><blockquote>
<br><br><br>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<h3> ¿Cuál de los siguientes colores es tu favorito?</h3>
<br><br><INPUT TYPE="checkbox" NAME="Azul"> Azul
<br>
<INPUT TYPE="checkbox" NAME="Verde"> Verde
<br>
<INPUT TYPE="checkbox" NAME="Rojo"> Rojo
<br>
<INPUT TYPE="checkbox" NAME="Amarillo"> Amarillo
</FORM>
</blockquote></blockquote></blockquote></blockquote>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina6.

A continuación en la figura I.71 se muestra las casillas de verificación:



I.71 Casillas de Verificación

🚦 Botones de Opción.

Si debemos escoger una entre varias opciones, lo correcto es utilizar botones de opción. Estos botones podemos acomodarlos en grupos, para solamente escoger una opción de un grupo determinado. Cuando enviemos el formulario una de las opciones se encuentra seleccionada el valor que tenga el atributo value será enviado al servidor. Para las opciones que no se encuentran marcadas se transfiere una cadena vacía. Para colocar el botón de opción la estructura es la siguiente:

```
<input type="radio" name="SistemaOperativo" value = "Windows" checked>
```

Donde:

- ✓ **type:** Define el radio, es decir que es un botón de opción.

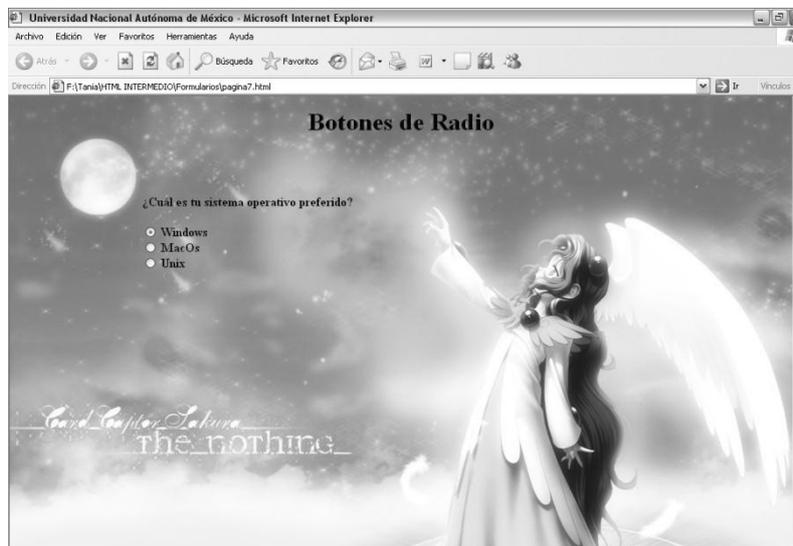
- ✚ **NOTA:** Hay que mencionar que si estamos realizando un grupo para poder seleccionar una de las opciones el atributo name, en todas las opciones debe ser el mismo, el único que cambia es el atributo value.

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\princesa.jpg">
<center><h1><b>Botones de Radio</b></h1><b></b></center>
<blockquote><blockquote><blockquote><blockquote>
<br><br><br>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
¿Cuál es tu sistema operativo preferido?
<BR><BR>
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="Windows" CHECKED>
Windows <br>
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="Mac"> MacOS <br>
<INPUT TYPE="radio" NAME="SistemaOperativo" VALUE="Unix"> Unix
</FORM> </blockquote></blockquote></blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina7.

A continuación en la figura I.72 se muestra los botones de radio:



I.72 Botones de Radio.

✚ Botones en el Formulario.

En la actualidad muchas de las páginas utilizadas en Internet utilizan los botones de enviar y borrar. HTML tiene esta propiedad dentro de la etiqueta <input>.

La función enviar provoca la transferencia de los datos del formulario al servidor. Su estructura es la siguiente:

```
<input type= "submit" name= "Enviar" value= "Enviar">
```

Donde:

- ✓ **type:** Define el valor **submit**, para configurar la función del botón al cual será enviar los datos del formulario al servidor.

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="princesa.jpg">
<center><font color=white><h1><b>Botones de Envio</b></font></center>
<blockquote><blockquote><blockquote><blockquote><br><br><br>
<FORM ACTION="mailto:wizard_pag@yahoo.com.mx" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
Escribe tu Apellido:
<BR><INPUT TYPE="text" NAME="Apellido">
<P><INPUT TYPE="submit" VALUE="Enviar datos">
</FORM>
</blockquote></blockquote></blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina8.

A continuación en la figura I.73 se muestra un botón:



I.73 Botón

Junto con el botón enviar, en los formularios también podemos integrar el botón borrar, el cual colocará el formulario, tal y como estaba la primera vez que lo abrimos. Para generar este botón la estructura es la siguiente:

```
<INPUT TYPE="reset" VALUE="Borrar datos">
```

Donde:

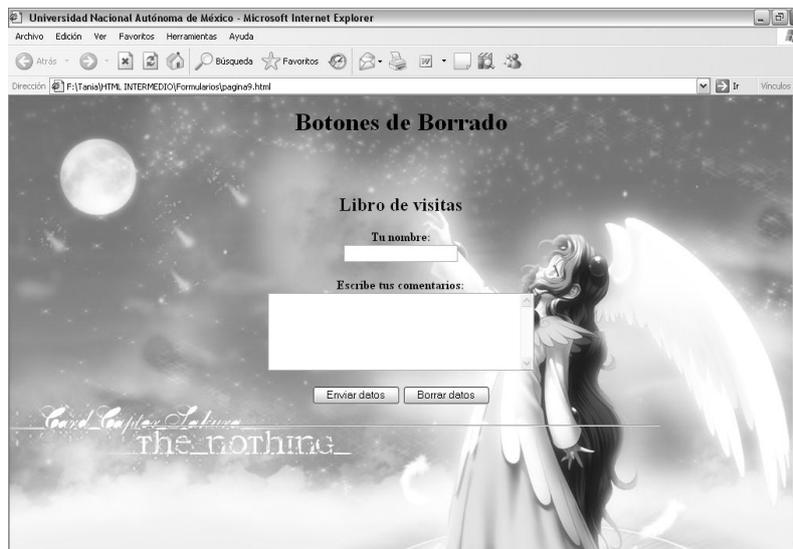
- ✓ **type:** Define la función reset, la cual se encarga de restablecer el formulario a su forma original.
- ✓ **value:** Establece el nombre del botón.

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\princesa.jpg">
<center><h1><b>Botones de Borrado</b></h1><b></b></center>
<blockquote><blockquote><blockquote><blockquote><br><br><br>
<P><CENTER> <H2>Libro de visitas</H2>
<P><FORM ACTION="mailto:farocena@lander.es" METHOD="POST"
ENCTYPE="TEXT/PLAIN"> Tu nombre:
<BR><INPUT TYPE="text" NAME="Nombre">
<P>Escribe tus comentarios:
<BR><TEXTAREA NAME="Comentarios" ROWS="6" COLS="40"> </TEXTAREA>
<P><INPUT TYPE="submit" VALUE="Enviar datos">
<INPUT TYPE="reset" VALUE="Borrar datos">
</FORM> <P> <HR> </blockquote></blockquote></blockquote></blockquote>
</body></html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina9.

A continuación en la figura I.74 se muestra un botón de borrado:



I.74 Botón de Borrado.

Cuadros de Lista Despegables.

La selección de una opción mediante casillas de verificación y botones de opción tiene un gran inconveniente debido a que ocupan mucho espacio dentro de la página. Para esto HTML tiene la opción de poder utilizar cuadros de lista despegables, en los cuales el usuario tiene la posibilidad de elegir una o varias opciones.

Para Mostrar un cuadro de lista despegable necesitamos de la etiqueta <select>. Ejemplo:

```
<SELECT NAME="ColorPreferido" MULTIPLE SIZE="3">
```

Si utilizamos el atributo multiple size= "3", indicará al navegador que debe mostrar tres elementos de los que se encuentren en la lista.

Para mostrar las opciones de las cuales podemos escoger, se usa la etiqueta <option>, de forma individual con cada una de las opciones. Ejemplo:

```
<OPTION>Rojo </OPTION>
```

La estructura completa es la siguiente:

```
<html>
<head>
<title>
Universidad Nacional Autónoma de México
</title>
</head>
<body background="princesa.jpg">
<center><h1><b>Introducción por Medio de Menús</b></h1><b></b></center>
<blockquote><blockquote><blockquote><blockquote>
<br><br><br>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
¿Cuál es tu color preferido? <BR>
<SELECT NAME="ColorPreferido" MULTIPLE SIZE="3">
<OPTION>Rojo </OPTION>
<OPTION>Verde </OPTION>
<OPTION>Azul </OPTION>
<OPTION>Amarillo </OPTION>
</SELECT >
</FORM>
</blockquote></blockquote></blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina10.

A continuación en la figura I.75 se muestra un cuadro de lista despegable:



I.75 Cuadro de lista despegable.

Si nosotros solo queremos que nos muestre una ventana y que al seleccionar la ventana nos muestre todas las opciones que tenemos para seleccionar lo podemos hacer mediante la siguiente estructura:

<SELECT NAME="ColorPreferido">

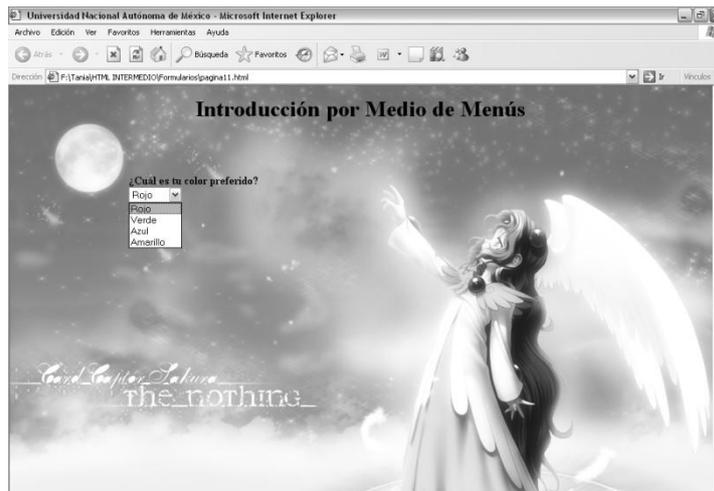
Si no damos ningún otro atributo solo nos mostrará una ventana de selección.

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\princesa.jpg">
<center><h1><b>Introducción por Medio de Menús</b></h1><b></b></center>
<blockquote><blockquote><blockquote><blockquote>
<br><br><br>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
¿Cuál es tu color preferido?
<BR><SELECT NAME="ColorPreferido">
<OPTION>Rojo
<OPTION>Verde
<OPTION>Azul
<OPTION>Amarillo
</SELECT >
</FORM>
</blockquote></blockquote></blockquote></blockquote></body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina11

A continuación en la figura I.76 se muestra la ventana de selección:



I.76 Ventana de selección.

🚫 Controles Ocultos.

En algunos casos cuando realizamos un formulario necesitamos enviar datos que el mismo usuario no debe de ver. Para estos casos debemos ocupar los controles ocultos que tienen los formularios. Para esto cuando el navegador esté listo para enviar los datos, no mostrará los valores. Esto se puede realizar mediante la estructura:

```
<input type="hidden" name="valor" value="nombre">
```

Donde:

- ✓ **type:** Contiene el elemento **hidden**, este elemento oculta el valor a la vista del usuario.

La estructura completa es la siguiente:

```
<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\princesa.jpg">
<center><h1><b>Formulario de Casillas de Verificación</b></h1><b></b></center>
<blockquote><blockquote><blockquote><blockquote><br><br><br>
<FORM ACTION="mailto:dirección_de_email" METHOD="POST"
ENCTYPE="TEXT/PLAIN">
<h3> ¿Cual de los siguientes colores es tu favorito?</h3><br><br>
<INPUT TYPE="hidden" NAME="Azul" checked> Azul<br>
<INPUT TYPE="hidden" NAME="Verde"> Verde<br>
<INPUT TYPE="hidden" NAME="Rojo"> Rojo<br>
<INPUT TYPE="hidden" NAME="Amarillo"> Amarillo
</FORM>
</blockquote></blockquote></blockquote></blockquote>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina12

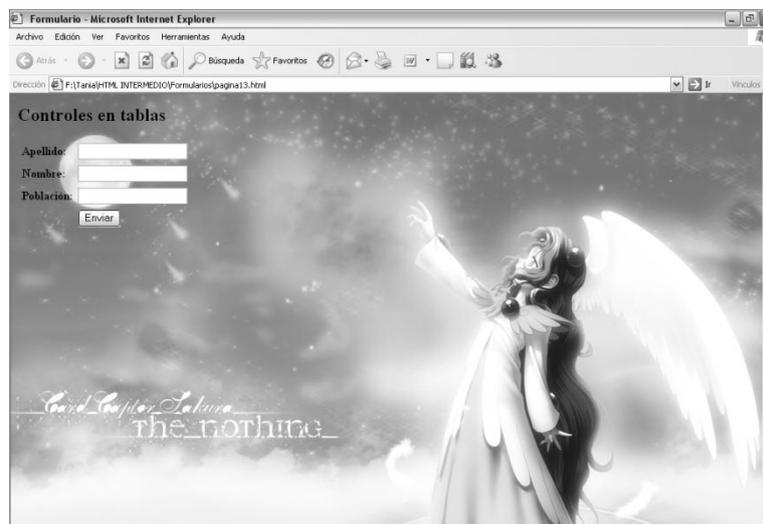
🚦 Tablas dentro del Formulario.

Dentro de los formularios podemos usar cualquiera de las etiquetas que contiene HTML, pero si queremos darle más presentación a los formularios podemos incrustarlos dentro de una tabla, para darle un mejor formato. Un ejemplo sencillo de un formulario dentro de una tabla es el siguiente:

```
<html>
<head>
<title>Formulario</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\princesa.jpg">
<h2>Controles en tablas</h2>
<form method="POST" action="/cgi-bin/test">
<table BORDER="0" CELSPACING="4">
<tr>
<td><b>Apellido:</b>
<td><input type="text" name="ct1">
<tr>
<td><b>Nombre:</b>
<td><input type="text" name="ct2">
<tr>
<td><b>Población:</b>
<td><input type="text" name="ct3">&nbsp;
<tr><td>&nbsp;
<td><input type="submit" name="bsend" value="Enviar">
</table>
</form>
</body>
</html>
```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina13.

A continuación en la figura I.77 se muestra una tabla dentro del formulario:



I.77 Tabla dentro del formulario.

Con estas herramientas de formularios que hemos visto, podemos crear formularios como el siguiente:

```

<html>
<head>
<title>Universidad Nacional Autónoma de México</title>
</head>
<body background="F:\Tania\HTML INTERMEDIO\lain.jpg" text="white">
<center>
<blockquote><blockquote><blockquote><blockquote>
<br><br><br>
<p><center>
<td><h2>Datos Personales</h2> </center></td>
<td><p><form action="programa1.php"
method="POST"enctype="text/plain"></td>
<table border=1>
<th colspan=3>
<tr>
<td>Nombre:</td>
<td>Apellido Paterno:</td>
<td>Apellido Materno:</td>
</tr>
<tr >
<td><input type="text" name="Nombre"></td>
<td><input type="text" name="Apellido_Paterno"></td>
<td><input type="text" name="Apellido_Materno"></td>
</tr>
<tr>
<td>Fecha de Nacimiento:</td>
<td colspan=2><select name="Nacimiento">
<option>1
<option>2
<option>3
<option>4
<option>5
</select >
<select name="MES">
<option>Enero
<option>Febrero
<option>Marzo
<option>Abril
<option>Mayo
</select >
<select name="MES">
<option>1986
<option>1987
<option>1988
<option>1989
<option>1989
</select ></td>
</tr><tr><td><p>Domicilio: </td>
<td colspan=2><br><textarea name="Comentarios" rows="6" cols="40">
</textarea> </td></tr><tr><p> <hr> <td>Sexo</td><br><br>
<td colspan=2><input type="radio" name="sexo" value="Femenino" checked>
Femenino

```

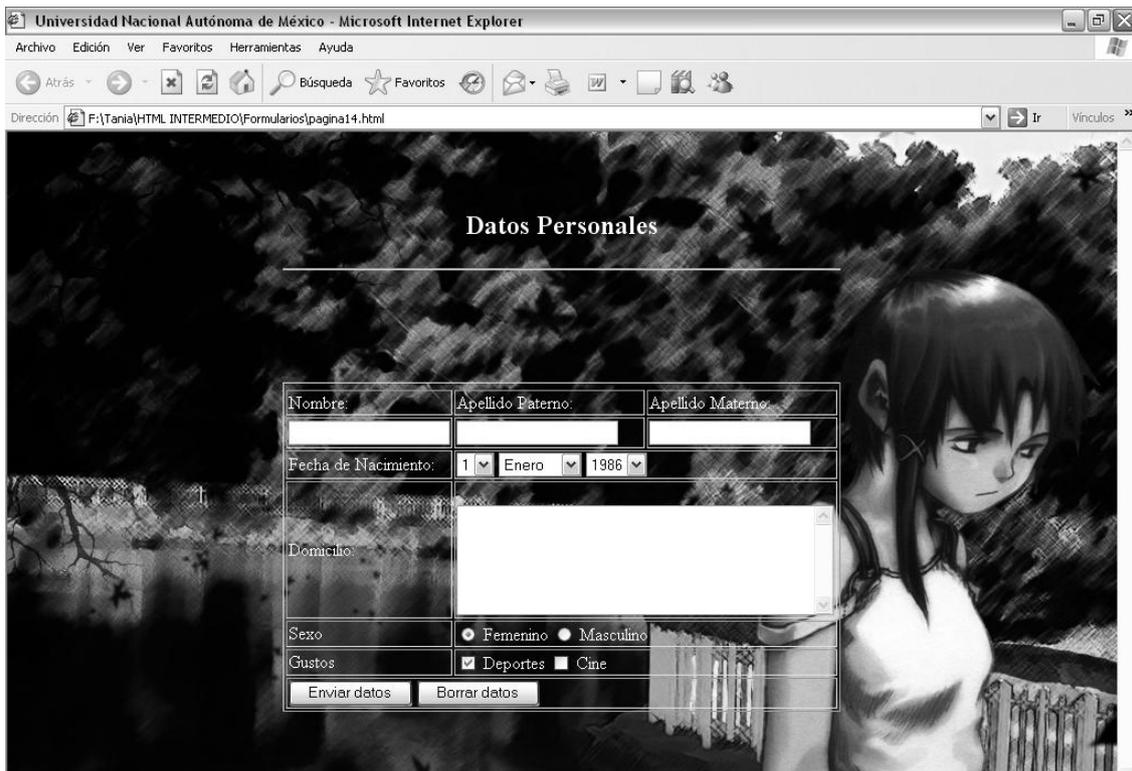
```

<INPUT TYPE="radio" NAME="sexo" VALUE="Masculino"> Masculino </td>
<BR></tr><tr><BR><br><td>Gustos</td>
<td colspan=2><INPUT TYPE="checkbox" NAME="sexo" VALUE="Deportes"
CHECKED> Deportes
<INPUT TYPE="checkbox" NAME="sexo" VALUE="Cine"> Cine</td>
</tr><tr><P>
<td colspan=3><INPUT TYPE="submit" VALUE="Enviar datos">
<INPUT TYPE="reset" VALUE="Borrar datos"> </td>
</tr></FORM>
</p>
</TABLE>
</blockquote></blockquote></blockquote></blockquote>
</center>
</body>
</html>

```

Este ejercicio, lo podemos ver en la carpeta HTML Intermedio, Formularios, pagina14.

A continuación en la figura I.78 se muestra una tabla dentro del formulario:



I.78 Tabla dentro del formulario.



Capítulo II

CSS

Capítulo II CSS

CSS es un lenguaje de hojas de estilo creado para controlar el aspecto y presentación de los documentos que nos encontramos en Internet, algunos definidos como HTML y XHTML. Utilizar hojas de estilo es la mejor forma de separar el contenido de un documento y el formato de presentación que tendrá a la vista del usuario.

Utilizar hojas de estilo tiene grandes ventajas, ya que con ello podemos mejorar la presentación, pero sobre todo si nosotros necesitamos colocar este mismo formato en más páginas con solo hacer una hoja de estilo, podemos hacer que todas tengan el mismo formato, sin la necesidad de estar escribiendo los atributos dentro del HTML, ya que estos serán almacenados en un archivo por aparte, aunque cabe mencionar que los estilos también los podemos colocar dentro del documento HTML.

Una vez que tenemos creados todo el contenido que tendrá el documento, en la hoja de CSS, podemos definir el color, tamaño y tipo de letra, la posición de cada elemento dentro de la página etc.

¿Que es lo que hace exactamente CSS?

CSS contiene atributos los cuales se colocan dentro de alguna de las etiquetas que contiene HTML, en la cual, cuando mandamos llamar a la hoja de estilo, el documento HTML, verificará que ninguna de las etiquetas de formato que tiene, se encuentre dentro de la hoja de estilo, de lo contrario si encuentra una etiqueta en el CSS, este tomara las propiedades que le hayamos dado a la etiqueta y las colocará en el explorador para poderlas mostrar al usuario.

II.1 Formas de Estructurar un CSS

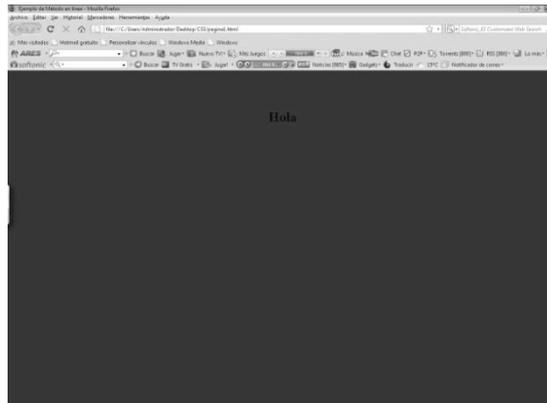
Método en Línea.

Una de las maneras de colocar el estilo en la página es haciendolo dentro del mismo HTML, insertándolo en la etiqueta, además de indicar que manejaremos un estilo con el atributo style. Un ejemplo de esto es:

```
<html>
<head>
<title>Ejemplo de Método en Línea</title>
</head>
<body style="background-color:red;">
<br>
<br>
<br>
<center><h1>Hola</h1></center>
</body>
</html>
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina 1.

A continuación en la figura II.1 se muestra el método en línea:



II.1 Método en línea.

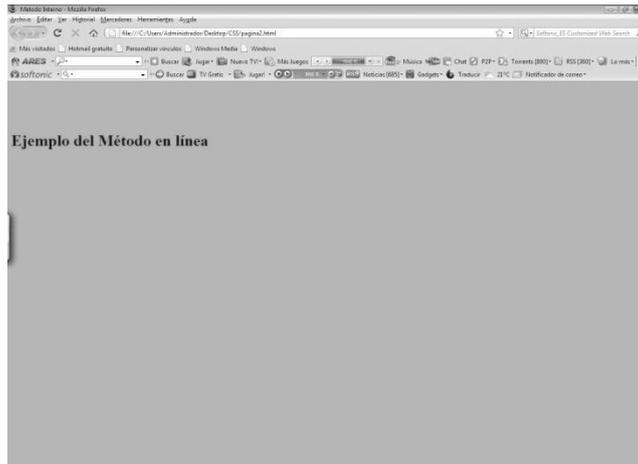
Método Interno.

Otro de los métodos que podemos utilizar es antes de entrar a la etiqueta `<body>`, este método también se encuentra dentro del documento HTML. Su estructura es la siguiente:

```
<html>
<head>
<title>Metodo Interno</title>
<style type = "text/css">
  body
  {
    background-color:#00FF00;
  }
</style>
</head>
<body>
<br>
<br>
<br>
<H1><p>Ejemplo del Método en línea</p><H1>
</body>
</html>
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina2.

A continuación en la figura II.2 se muestra el método interno:



II.2 Método Interno

🚦 Método Externo

Otra manera en la que podemos incluir las hojas de estilo es mediante un archivo externo al cual le podemos agregar todas las propiedades que necesitemos, y solo mandarlas llamar, a nuestro documento, este documento debe llevar la extensión .css, en vez de utilizar .html, una de las ventajas de utilizar este tipo de métodos, es que podemos utilizar este archivo tantas veces sean necesario, con un documento, u otro documento diferente a él. Un ejemplo de esto es:

✓ Documento pagina3.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <h1>Mi primera hoja de estilo</h1>
</body>
</html>
```

✓ Style.css

```
body {
  background-color:"black";
  background-image:URL("H:\HTML INTERMEDIO\wam_04.GIF");
  background-repeat:NO-REPEAT;
  background-attachment:FIXED;
```

```
background-position:LEFT;  
}
```

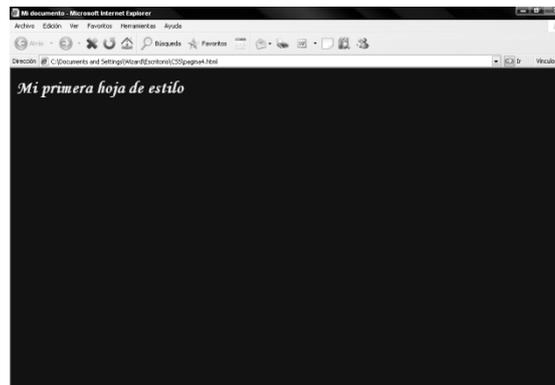
Donde con la etiqueta link, estamos haciendo la referencia de donde encontrará el archivo, del cual extraeremos todas las propiedades necesarias para la página.

De esta manera cada que necesitemos cambiar el color de fondo o las letras de la página lo podemos hacer de la manera más sencilla, sin la necesidad de buscar en todo el código de la página, todos los formatos para decoración.

De ahora en adelante solo hablaremos de los atributos que permiten hacer los cambios al documento HTML, pero hay que mencionar que todos los atributos que se explicarán a continuación se pueden utilizar en todas la etiquetas de HTML, vistas en el capítulo anterior, salvo algunas que tienen sus excepciones, pero en estos casos se explicará cuándo no la podemos usar. Todos estos atributos siguientes se escribirán en el archivo con la extensión .css

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina3.

A continuación en la figura II.3 se muestra el método externo:



II.3 Método Externo.

Reglas de las Hojas de Estilo.

1. La estructura de CSS para una propiedad es:

```
propiedad:valor;
```

2. Una propiedad debe colocarse en un bloque encerrado entre llaves.

3. Un elemento puede ser incluido aún y cuando ya se encuentre otro elemento colocado.

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

Esta son las reglas básicas que debemos tener en cuenta a la hora de realizar una hoja de estilo.

II.2 Color de Fondo

Como ya hemos visto en el tema anterior nosotros podemos cambiar el color de fondo de una página, o varias si se tiene un sitio establecido esto se lleva a cabo con el atributo:

background-color : #FF0000;

Este atributo aplica el color de fondo en la página, podemos colocar el valor en forma hexadecimal o su nombre en ingles solo que este llevará comillas dobles y el nombre entre ellas. Ejemplo:

background-color : "yellow";

Esta etiqueta no solo puede estar conectada dentro del body, sino que también la podemos colocar en cualquier otra, como por ejemplo un párrafo, cuando CSS, localice la etiqueta de párrafo, mandará como color de fondo el que se tendrá dictaminado, por lo que la página puede tener muchos colores de fondo.

II.3 Color de los Títulos y Textos.

Con el atributo **color**, podemos definir el color que tendrán textos, dentro de la página, este atributo funciona con todas las etiquetas que conocemos, sin ningún problema, de igual manera que el color de fondo podemos colocarlo de la forma hexadecimal o en su forma de color en ingles. Ejemplo:

✓ Documento pagina4.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
  <h1>Mi primera hoja de estilo</h1>
</body>
</html>
```

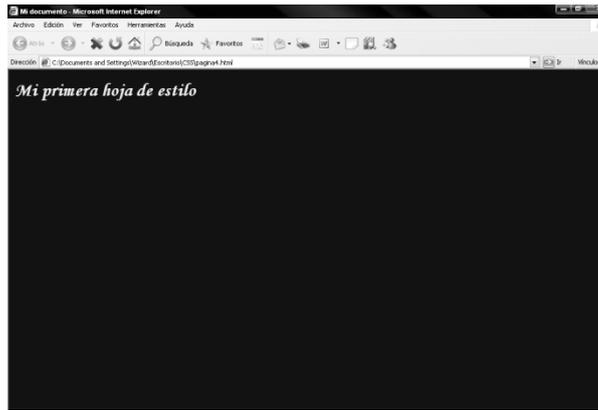
✓ Documento style.css

```
h1 {
  color: "YELLOW";
}
```

Donde cuando nosotros estemos listos para ver el sitio, el explorador buscará en la hoja de estilo y encontrará que a la etiqueta h1, tendrá que ponerle como color de letra el amarillo, a todo el texto que se encuentre encerrado dentro de la etiqueta <h1>.

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina4.

A continuación en la figura II.4 se muestra el color de fondo:



II.4 Color de fondo.

II.4 Imágen de Fondo.

Al igual que en HTML, podemos colocar una imágen de fondo desde una hoja de estilo, esto ayuda mucho, cuando hablamos de páginas para empresas las cuales deben tener el mismo fondo, si en algún momento la empresa necesita cambiar el fondo de la página, en HTML, tendrá que cambiar página por página, mientras que si lo hacemos con CSS, solo cambiamos una sola línea, y automáticamente se heredan a todas las demás ya que CSS, funciona en forma de árbol.

El atributo para cambiar el color de fondo es:

Background-image: URL("ruta de la imágen");

Ejemplo:

✓ Documento pagina5.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style1.css" />
</head>
<body>
<br><br><br>
<blockquote> <blockquote>
  <h1>Mi primera hoja de estilo</h1>  <br>
    <h1>Colocación de Imágen de Fondo en CSS</h1>
</blockquote></blockquote>
</body>
</html>
```

✓ Documento style1.css

```
body {
  background-image:URL("2.jpg");
}

h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}
}
```

Hay que mencionar que cuando colocados más de un atributo de propiedad debemos colocar un punto y coma, para que una vez que el explorador se encuentre leyendo las propiedades de la página identifique una por una.

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina5.

A continuación en la figura II.5 se muestra la imagen de fondo:



II.5 Imagen de fondo

II.5 Repetir Imágen de Fondo.

Dentro de Internet existen diferentes páginas las cuales, la imagen que colocan de fondo es muy pequeña para verse en el explorador, por lo que el explorador la repite, las veces que sean necesarias, hasta poder, llenar toda ventana.

En CSS Existen propiedades en las cuales le indiquemos, cuantas veces deseamos que se repita la imagen, la estructura de esta propiedad es:

background-repeat: no-repeat;

En donde este atributo contiene varias opciones:

- ✓ **background-repeat: repeat-x:** La imagen solo se repite en el eje vertical de la ventana del explorador. Ejemplo:

❖ Documento pagina7.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style3.css"/>
</head>
<body>
<br><br><br>
<blockquote><blockquote>
<br><br><br><br>
  <h1>Mi primera hoja de estilo</h1> <br>
<h1>Repetición de Imagen de Fondo en CSS</h1>
</blockquote></blockquote>
</body>
</html>
```

❖ Documento style3.css

```
body {  
  background-color:"lightpink";  
  background-image:URL("zorro1.jpg");  
  background-repeat: repeat-x;  
}  
h1 {  
  color: "black";  
  font-family:Monotype Corsiva,verdana;  
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina7.

A continuación en la figura II.6 se muestra la repetición de la imagen de fondo verticalmente:



II.6 Repetir imagen de fondo verticalmente.

- ✓ **background-repeat: repeat-y:** La imagen solo se repite en el eje horizontal de la ventana del explorador. Ejemplo:

❖ Documento pagina8

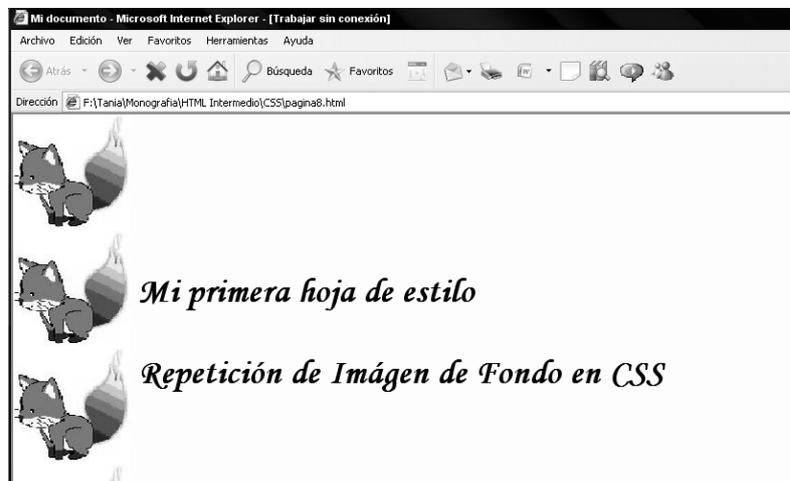
```
<html>  
<head>  
<title>Mi documento</title>  
<link rel="stylesheet" type="text/css" href="style4.css"/>  
</head>  
<body><br><br><br>  
<blockquote><blockquote><blockquote>  
<br><br><br><br>  
<h1>Mi primera hoja de estilo</h1>  
<br><h1>Repetición de Imágen de Fondo en CSS</h1>  
</blockquote></blockquote></blockquote>  
</body>  
</html>
```

❖ Documento style4.css

```
body {
  background-color:"lightyellow";
  background-image:URL("C:\Documents and
Settings\Wizard\Escritorio\CSS\Imágenes\zorro1.jpg");
  background-repeat: repeat-y;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina8.

A continuación en la figura II.7 se muestra la repetición de la imagen de fondo horizontalmente:



II.7 Repetir Imagen de Fondo Horizontalmente.

- ✓ **background-repeat: repeat** : La imagen se repite en el eje horizontal y vertical. Esta función hace lo mismo, cuando no especificamos cuantas veces queremos que lo repita.

❖ Documento pagina9.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style5.css"/>
</head>
<body>
<br><br><br>
  <h1>Mi primera hoja de estilo</h1>
  <br>
  <h1>Repetición de Imagen de Fondo en CSS</h1>
</body>
</html>
```

❖ Documento style5.css

```
body {
  background-color:"lightyellow";
  background-image:URL("zorro1.jpg");
  background-repeat: repeat;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina9.

A continuación en la figura II.8 se muestra la repetición de la imagen de fondo en toda la página:



II.8 Repetir imagen de fondo en toda la página

- ✓ **background-repeat: no-repeat:** La imagen no se repite, solo se coloca una sola vez. Ejemplo:

❖ Documento pagina6.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style2.css" />
</head>
<body>
<br><br><br><blockquote><blockquote>
  <h1>Mi primera hoja de estilo</h1>
  <br>
  <h1>Repetición de Imagen de Fondo en CSS</h1>
</blockquote></blockquote>
</body>
</html>
```

❖ Documento style2.css

```
body {
  background-color:"yellow";
  background-image:URL("zorro1.jpg");
  background-repeat:NO-REPEAT;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}
```

Con esta opción nos permite solo repetir una vez la imagen y poder colocar además un color de fondo, teniendo ambos a la vez.

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina6.

A continuación en la figura II.9 se muestra la no repetición de la imagen de fondo:



II.9 No repetir la imagen de fondo.

II.6 Fondo de Imagen Fija ó Desplazada.

En algunas ocasiones cuando tenemos mucha información que debemos colocar en la página la imagen no es lo suficientemente grande para contener todo el texto, por lo que cuando aparecen las barras de desplazamiento, el explorador vuelve a repetir la imagen se fondo, en cuanto se termina la imagen primera, lo cual hace que la página pierda presentación.

Para resolver este tipo de problemas utilizamos el atributo attachment, el cual contiene dos opciones para utilizarlo, estas son:

✓ **background-attachment:scroll**

Con esta propiedad hacemos que la imagen se repita las veces que sea necesaria, aunque se ve de forma muy notoria, el corte de imagen, en algunas ocasiones existen imágenes, las cuales cuando se repiten no se notan y se piensa que es solo una imagen, pero la realidad es que son varias. Ejemplo:

❖ Documento pagina10.html

```
<html>
<head>
```

```

<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style6.css"/>
</head>
<body>
<br><br><br>
<blockquote>
<blockquote><blockquote>
<br><br><br><br>
<h1>Mi primera hoja de estilo</h1>
<br>
<h1>Imágen de Fondo en CSS Desplazada</h1>
<br><br>
<p>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>

<p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p>

<p>Lo que acabamos de describir es la forma tradicional de hacer dibujos animados. Hoy día, con los ordenadores, el procedimiento ha cambiado, se ha simplificado; pero la magia de ver cómo un dibujo cobra vida sigue asombrándonos.</p>

<h1>LOS PRINCIPIOS DE LA ANIMACIÓN</h1>

<p>Mucho antes de que se proyectara la primera película, en el año 1895, ya se sabía cómo dar apariencia de movimiento a un personaje inanimado. Por ejemplo, existía un juguete llamado zoótropo que consistía en una sucesión de imágenes dibujadas en el interior de un cilindro que, al hacerlo girar, producía la impresión de movimiento.</p>

<p>Los primeros dibujos animados de la historia del cine los hizo Edwin S. Porter, en 1905. Porter fotografió por separado una serie de letras; al proyectar estas imágenes a gran velocidad, las letras parecían correr por la pantalla hasta formar una frase.</p>

<p>En 1906, James Stuart Blackton, de la productora Vitagraph, experimentó con esta técnica para dotar de movimiento a una serie de sencillos dibujos y, más tarde, de muñecas de juguete. Al año siguiente, probó con objetos modelados en arcilla que iba transformando poco a poco y fotografiando. Así, Blackton se convirtió en el inventor de la animación de muñecos.</p>

<h1>LAS PRIMERAS PELÍCULAS DE DIBUJOS ANIMADOS</h1>

<p>Entre 1908 y 1910, un cineasta europeo, el francés Émile Cohl, concentró sus esfuerzos en el desarrollo de estas técnicas de animación. Fue él quien creó la primera serie de dibujos animados; consistía en una sucesión de figuras muy esquemáticas que daban saltos y se caían, asombrosamente, sin hacerse daño.</p>
</blockquote></blockquote></blockquote>
</body>
</html>

```

❖ Documento style6.css

```
body {
  background-color:"lightyellow";
  background-image:URL("1.jpg");
  background-repeat:repeat;
  background-attachment:scroll;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina10.

A continuación en la figura II.10 se muestra la imagen de fondo repetida:



II.10 Imagen de fondo repetida.

❖ Background-attachment:fixed

Con esta propiedad la imagen queda fija, y no se repite, ya que lo que tendrá movimiento será la información, a que se moverá por toda la ventana del explorador. Ejemplo:

❖ Documento pagina11.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style7.css"/>
</head>
<body>
<br><br><br><blockquote><blockquote><blockquote>
<br><br><br><br>
  <h1>Mi primera hoja de estilo</h1>
  <br><h1>Imagen de Fondo en CSS Desplazada</h1>
<br><br>
  <p>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento
del personaje que quiere representar por separado, en una hoja diferente, de celuloide
```

o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>

<p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p>

<p>Lo que acabamos de describir es la forma tradicional de hacer dibujos animados. Hoy día, con los ordenadores, el procedimiento ha cambiado, se ha simplificado; pero la magia de ver cómo un dibujo cobra vida sigue asombrándonos.</p>

<h1>LOS PRINCIPIOS DE LA ANIMACIÓN</h1>

<p>Mucho antes de que se proyectara la primera película, en el año 1895, ya se sabía cómo dar apariencia de movimiento a un personaje inanimado. Por ejemplo, existía un juguete llamado zoótropo que consistía en una sucesión de imágenes dibujadas en el interior de un cilindro que, al hacerlo girar, producía la impresión de movimiento.</p>

<p>Los primeros dibujos animados de la historia del cine los hizo Edwin S. Porter, en 1905. Porter fotografió por separado una serie de letras; al proyectar estas imágenes a gran velocidad, las letras parecían correr por la pantalla hasta formar una frase.</p>

<p>En 1906, James Stuart Blackton, de la productora Vitagraph, experimentó con esta técnica para dotar de movimiento a una serie de sencillos dibujos y, más tarde, de muñecas de juguete. Al año siguiente, probó con objetos modelados en arcilla que iba transformando poco a poco y fotografiando. Así, Blackton se convirtió en el inventor de la animación de muñecos.</p>

<h1>LAS PRIMERAS PELÍCULAS DE DIBUJOS ANIMADOS</h1>

<p>Entre 1908 y 1910, un cineasta europeo, el francés Émile Cohl, concentró sus esfuerzos en el desarrollo de estas técnicas de animación. Fue él quien creó la primera serie de dibujos animados; consistía en una sucesión de figuras muy esquemáticas que daban saltos y se caían, asombrosamente, sin hacerse daño.</p>

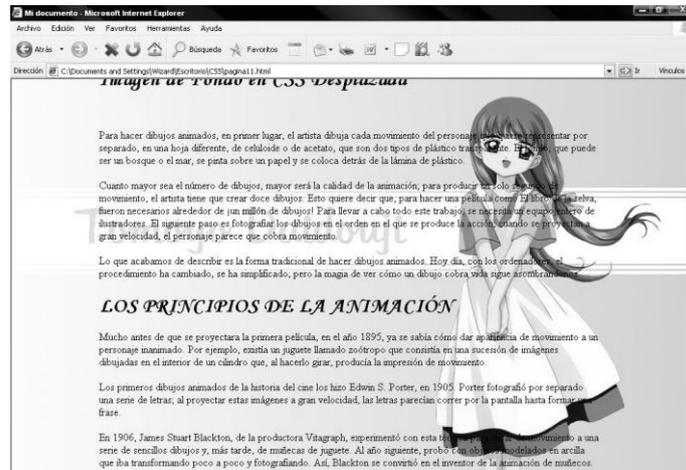
</blockquote></blockquote></blockquote>
</body>
</html>

❖ Documento style7.css

```
body {  
    background-color:"lightyellow";  
    background-image:URL("1.jpg");  
    background-repeat:repeat;  
    background-attachment:fixed;  
}  
h1 {  
    color: "black";  
    font-family:Monotype Corsiva,verdana;  
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina11.

A continuación en la figura II.11 se muestra la imagen de fondo fija:



I.11 Imagen de fondo fija.

Esto dará mejor presentación a nuestra página Web.

II.7 Ubicación de la Imagen de Fondo.

La imagen de fondo nosotros la podemos colocar en la posición que nosotros queremos, para esto CSS, nos da la posibilidad de hacerlo con varias opciones, su estructura es:

background-position:right;

Aquí tenemos que indicar que la imagen no se repetirá o solo se repetirá en alguno de los dos ejes, ya mencionados anteriormente.

Las opciones que tiene esta propiedad son:

✓ **background-position:bottom**

Con esta propiedad podemos colocar la imagen en la parte inferior del explorador. Ejemplo:

❖ **Documento pagina12.html**

```
<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style8.css"/>
</head>
<body>
<br><br><br>
<h1>Mi primera hoja de estilo</h1> <br>
    <h1>Repetición de Imagen de Fondo en CSS</h1>
</body>
</html>
```

❖ **Documento style8.css**

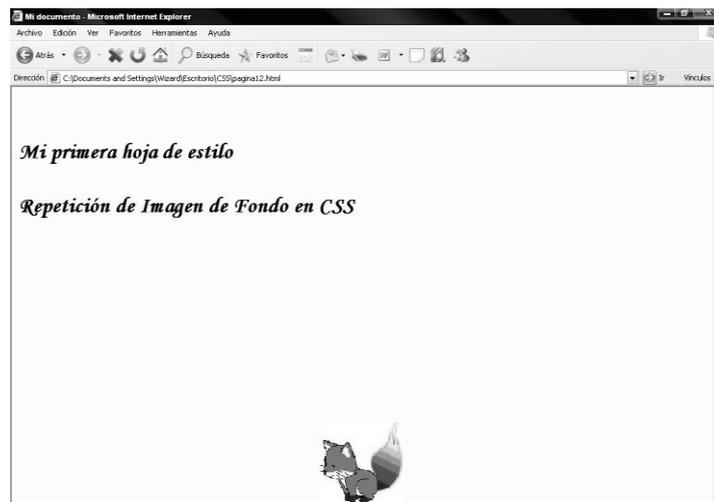
```

body {
  background-color:"lightyellow";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:bottom;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina12.

A continuación en la figura II.12 se muestra la ubicación de imagen de fondo:



II.12 Ubicación de imagen de fondo.

✓ **background-position:center**

Esta propiedad ubica a la imagen en la parte central del explorador.

❖ **Documento pagina13.html**

```

<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style9.css"/>
</head>
<body>
<br><br><br>
  <h1>Mi primera hoja de estilo</h1>
  <br>
  <h1>Repetición de Imagen de Fondo en CSS</h1>
</body>
</html>

```

❖ **Documento style9.css**

```

body {
  background-color:"lightyellow";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:center;
}

```

```

}
h1 {
    color: "black";
    font-family:Monotype Corsiva,verdana;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina13.

A continuación en la figura II.13 se muestra la ubicación de imagen de fondo centrada:



II.13 Ubicación de imagen de fondo centrada.

✓ **background-position:left**

Con esta propiedad posicionamos a la imagen del lado izquierdo del explorador.
Ejemplo:

❖ **Documento pagina14.css**

```

<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style10.css"/>
</head>
<body>
<br><br><br> <h1>Mi primera hoja de estilo</h1> <br>
<h1>Posición de Imagen de Fondo en CSS</h1>
</body>
</html>

```

❖ **Documento style10.css**

```

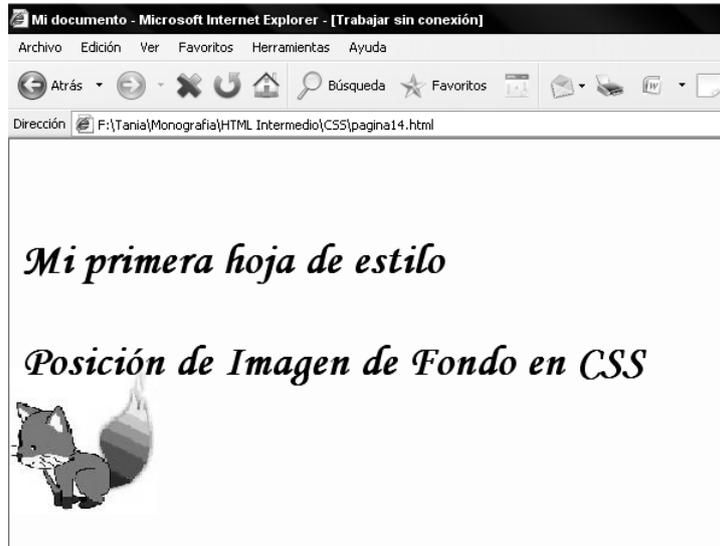
body {
    background-color:"lightyellow";
    background-image:URL("zorro1.jpg");
    background-repeat:no-repeat;
    background-position:left;
}
h1 {
    color: "black";
    font-family:Monotype Corsiva,verdana;
}

```

}

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina14.

A continuación en la figura II.14 se muestra la ubicación de imagen de fondo a la izquierda:



II.14 Ubicación de imagen de fondo a la izquierda.

✓ **background-position:right**

Con esta propiedad posicionamos a la imagen del lado derecho del explorador.
Ejemplo:

❖ **Documento pagina15.html**

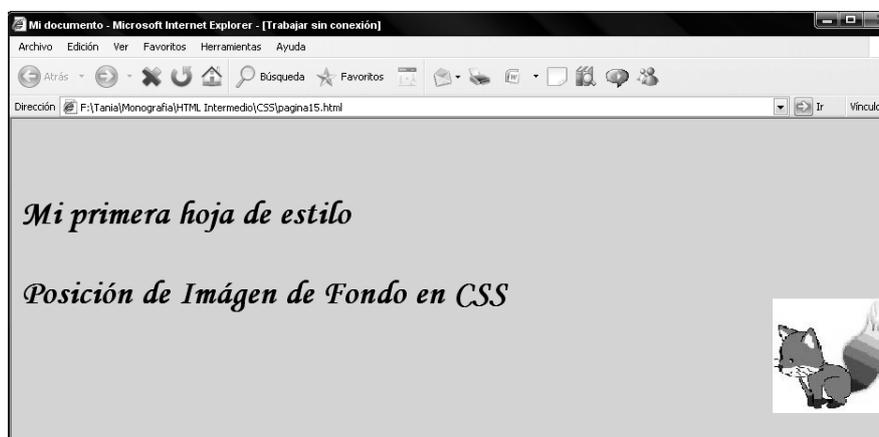
```
<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style11.css"/>
</head>
<body>
<br><br><br>
<h1>Mi primera hoja de estilo</h1> <br>
<h1>Posición de Imagen de Fondo en CSS</h1>
</body>
</html>
```

❖ **Documento style11.css**

```
body {
background-color:"lightgreen";
background-image:URL("zorro1.jpg");
background-repeat:no-repeat;
background-position:right;
}
h1 {
color: "black";
font-family:Monotype Corsiva,verdana;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina15.

A continuación en la figura II.15 se muestra la ubicación de imagen de fondo a la derecha:



II.15 Ubicación de imagen de fondo a la derecha.

II.8 Familia de Fuentes.

Algunas uno de los atributos que tenemos en HTML, es poder cambiar el tipo de fuente de letra de la página. CSS también permite hacer el cambio de letra por medio de la propiedad:

font-family: arial;

En donde solo tenemos que poner el atributo font-family, y el nombre correcto del tipo de letra que deseamos colocar en la página.

✚ NOTA: Si el nombre de la fuente llega a contener varios espacios en blanco, se debe colocar el nombre de la fuente entre comillas dobles. Ejemplo:

font-family: "Times New Roman", Arial;

Una de las maneras de asegurarnos que el usuario de la página contenga algún tipo de fuente de los que nosotros especificamos en la misma, es colocando dos a tres tipos de fuentes, así el explorador, cuando habrá la página en la computadora del usuario, y no contenga la letra principal, que es la primera que elegimos, pasará a la segunda, y así sucesivamente hasta encontrar una de las letras que se tengan especificadas, de lo contrario, pondrá la letra por default de los exploradores la cual es arial.

Este ejercicio, ya lo hemos colocado antes en los ejercicios anteriores, en los cuales podemos observar cómo se debe colocar esta propiedad.

II.9 Tamaño de la Fuente

Para cambiar el tamaño del texto que contendrá la página lo podemos hacer utilizando la propiedad:

font-size:30px;

En donde podemos utilizar las siguientes unidades, para especificar el tamaño en píxeles (px), porcentaje (%).

✚ **Nota:** Cabe destacar que esta propiedad no funciona dentro de las etiquetas de títulos (<h1>), debido a que esta etiqueta contiene ya tamaños específicos.

Ejemplo:

❖ Documento pagina16.html

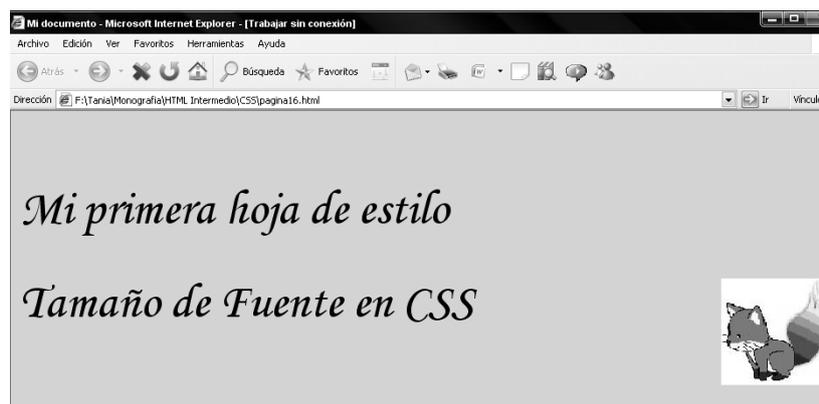
```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style12.css"/>
</head>
<body>
<br><br><br>
  <p>Mi primera hoja de estilo</p> <br>
  <p>Tamaño de Fuente en CSS</p>
</body>
</html>
```

❖ Documento style12.css

```
body {
  background-color:"lightgreen";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:right;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 300%;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina16.

A continuación en la figura II.16 se muestra el tamaño de la fuente:



II.16 Tamaño de la fuente.

II.10 Colocación de Sangrías

Así como nosotros colocamos sangrías dentro del documento HTML, en CSS también lo podemos hacer mediante la propiedad:

text-indent: 30px;

De igual manera que en el tamaño de fuente, podemos ocupar las unidades de tamaño, aunque cabe destacar, que para el caso de las sangrías se aconseja, utilizar la unidad de porcentaje, así cuando el usuario abra la página el tamaño de la sangría, se ajustará automáticamente al tamaño del monitor en que se este visualizando.

Ejemplo:

❖ **Documento pagina17.html**

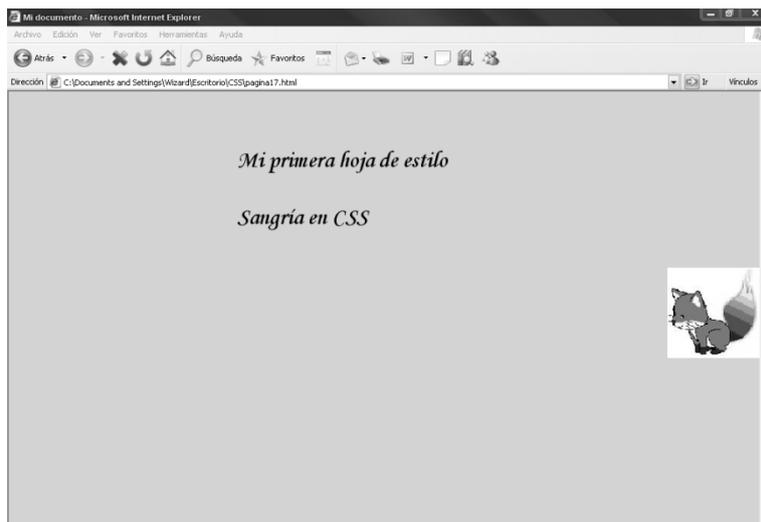
```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style13.css"/>
</head>
<body>
<br><br><br>
  <p>Mi primera hoja de estilo</p> <br><p>Sangría en CSS</p>
</body>
</html>
```

❖ **Documento style13.css**

```
body {
  background-color:"lightgreen";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:right;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 200%;
  text-indent: 30%;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina17.

A continuación en la figura II.17 se muestra la colocación de sangrías:



II.17 Colocación de Sangrías.

II.11 Alineación del Texto.

Dentro de CSS, podemos decidir cual será, la forma de alinear el texto, que colocaremos en nuestra página o por qué no una imagen, la estructura de esta propiedad es la siguiente:

text-align:right;

Las opciones que tiene esta propiedad son:

- ✓ **text-align:right** Alinea el texto del lado derecho del explorador. Ejemplo:

❖ **Documento pagina18.html**

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style14.css"/>
</head>
<body>
<br><br><br>
<p>Mi primera hoja de estilo</p> <br>
<p>Sangría en CSS</p>
</body>
</html>
```

❖ **Documento style18.css**

```
body {
  background-color:"lightgreen";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:right;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 200%;
  text-indent: 30%;
```

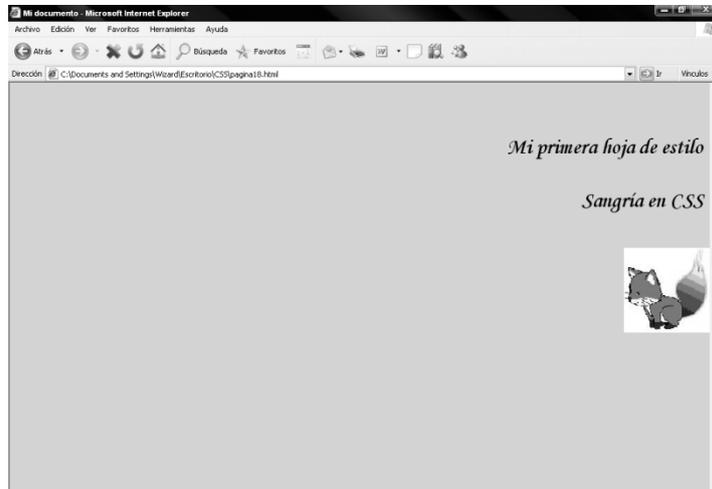
```

    text-align:right;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina18.

A continuación en la figura II.18 se muestra la alineación de texto a la derecha:



II.18 Alineación del Texto a la Derecha.

- ✓ **text-align:left** Alinea del texto a la izquierda de la ventana del navegador. Ejemplo:

❖ **Documento pagina19.html**

```

<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style15.css"/>
</head>
<body>
<br><br><br>
  <p>Mi primera hoja de estilo</p> <br>
  <p>Alineación de Texto en CSS</p>
</body>
</html>

```

❖ **Documento style19.css**

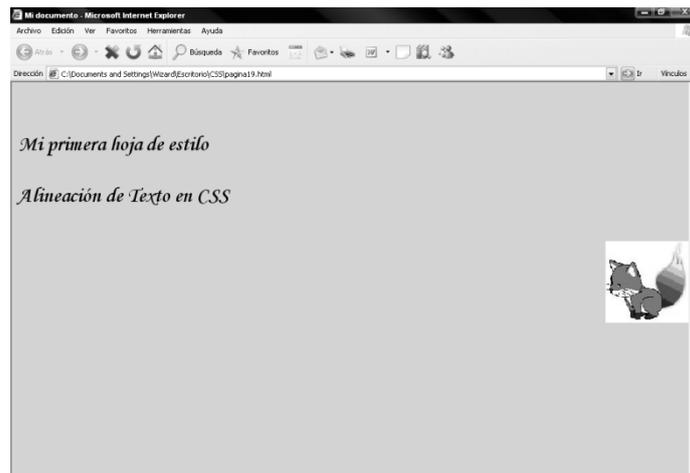
```

body {
  background-color:"lightgreen";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:right;
}
h1 {
  color: "black";
  font-family:Monotype Corsiva,verdana;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina19.

A continuación en la figura II.19 se muestra la alineación de texto a la izquierda:



II.19 Alineación del Texto a la Izquierda.

- ✓ **text-align: justify** Alinea el texto justificadamente dentro del explorador.
Ejemplo:

❖ **Documento pagina20.html**

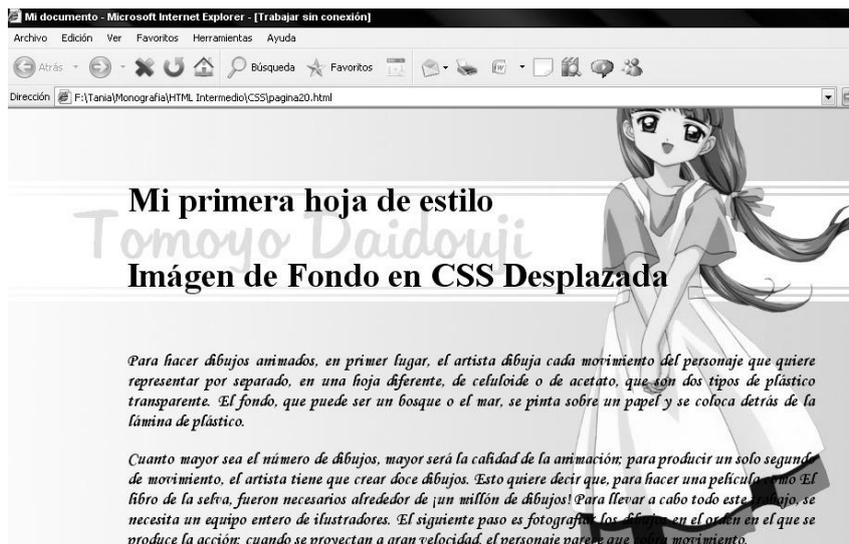
```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style16.css"/>
</head>
<body>
<br><br><br>
  <p>Mi primera hoja de estilo</p> <br>
  <p>Alineación de Texto en CSS</p>
</body>
</html>
```

❖ **Documento style16.css**

```
body {
  background-color:"lightgreen";
  background-image:URL("zorro1.jpg");
  background-repeat:no-repeat;
  background-position:right;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 200%;
  text-align:justify;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina20.

A continuación en la figura II.20 se muestra la alineación de texto justificado:



II.20 Alineación del Texto Justificado.

II.12 Decoración del Texto.

Con esta propiedad podemos darle mejor presentación a nuestras páginas ya que podemos resaltar un párrafo de los demás, por medio de un línea o un parpadeo. Su estructura es la siguiente:

text-decoration:underline;

Donde text-decoration tiene las siguientes opciones:

- ✓ **text-decoration:underline** : Con esta propiedad podemos hacer que el párrafo que seleccionemos sea subrayado por la parte inferior del texto. Ejemplo:

❖ Documento pagina21.html

```
<html>
  <head>
    <title>Mi documento</title>
    <link rel="stylesheet" type="text/css" href="style17.css"/>
  </head>
  <body>
    <br><br><br>
    <blockquote><blockquote><blockquote>
    <br><br>
    <h1>Mi primera hoja de estilo</h1> <br>
    <h1>Imágen de Fondo en CSS Desplazada</h1>
    <br><br>
    <p>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>

    <p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos
```

en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p></blockquote></blockquote></blockquote>

```
</body>
</html>
```

❖ Documento style17.css

```
body {
    background-image:URL("1.jpg");
    background-repeat:no-repeat;
    background-position:right;
    background-attachment:fixed;
}

p {
    color: "black";
    font-family:Monotype Corsiva,verdana;
    font-size: 100%;
    text-align:justify;
    text-decoration:underline;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina21.

A continuación en la figura II.21 se muestra la decoración del texto:



II.21 Decoración del Texto.

- ✓ **text-decoration:underline** : Con esta propiedad podemos hacer que el párrafo que seleccionemos sea subrayado por la superior del texto. Ejemplo:

❖ Documento pagina22.html

```
<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style18.css"/>
</head>
```

```

<body>
<br><br><br>
<blockquote><blockquote><blockquote>
<br><br>
<h1>Mi primera hoja de estilo</h1> <br>
<h1>Imágen de Fondo en CSS Desplazada</h1>
<br><br>
<p>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>

<p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p>

</blockquote></blockquote></blockquote>
</body>
</html>

```

❖ Documento style18.css

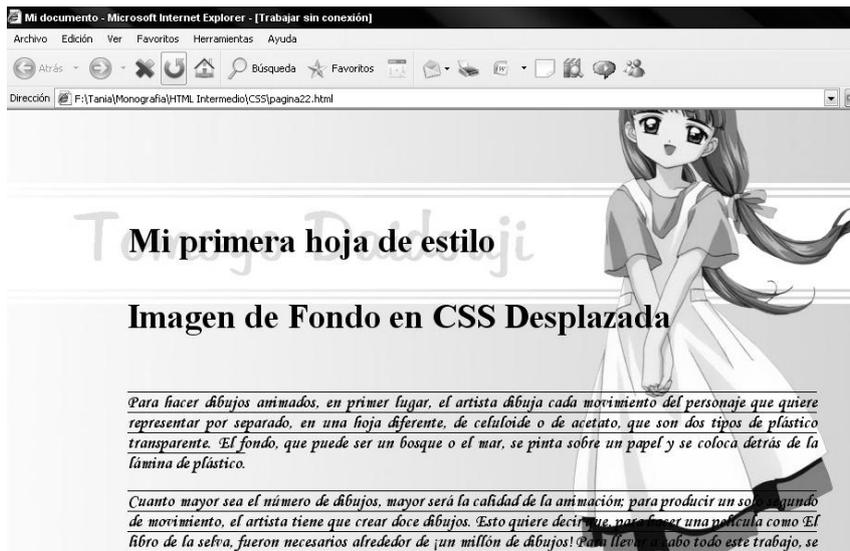
```

body {
  background-image:URL("1.jpg");
  background-repeat:no-repeat;
  background-position:right;
  background-attachment:fixed;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 100%;
  text-align:justify;
  text-decoration:underline;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina22.

A continuación en la figura II.22 se muestra la decoración del texto:



II.22 Decoración del Texto.

- ✓ **text-decoration:blink** Con esta propiedad podemos hacer que el párrafo que seleccionemos parpadee el texto mientras el usuario se encuentre en la página. Cabe mencionar que esta propiedad solo funciona en otros exploradores diferentes al de Microsoft. Ejemplo:

❖ Documento pagina23.html

```

<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style18.css"/>
</head>
<body>
<br><br><br>
<blockquote><blockquote><blockquote>
<br><br>
<h1>Mi primera hoja de estilo</h1>
<br><h1>Imagen de Fondo en CSS Desplazada</h1>
<br><br>
<p>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>

<p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p>

</blockquote></blockquote></blockquote>
</body>
</html>

```

❖ Documento style19.css

```
body {
  background-image:URL("1.jpg");
  background-repeat:no-repeat;
  background-position:right;
  background-attachment:fixed;
}
p {
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 100%;
  text-align:justify;
  text-decoration:blink;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina23.

A continuación en la figura II.23 se muestra el texto parpadeando:



II.23 Texto Parpadeando.

II.13 Enlaces.

Con las propiedades que nos da CSS, podemos cambiar el color de los enlaces tal y como lo hacemos en HTML, por medio de diferentes propiedades. La estructura general de esta propiedad es:

```
a{ color:blue; }
```

Esta es la forma de colocar la etiqueta dentro de archivo style, debe llevar la etiqueta <a>, la cual representan los hipervínculos, en HTML. Para identificar cada una de las propiedades que se tienen aquí, se utilizan clases, las cuales se mencionarán a mayor detalle, más adelante.

Las propiedades de los enlaces son:

✓ Enlace visitado

Con esta propiedad se puede indicar cuándo una página ya ha sido visitada. Con esta propiedad podemos indicar de qué color queremos mostrar los enlaces visitados por medio del atributo **color**. Así aunque salga de la página y vuelva entrar seguirá mostrando que es un enlace visitado, tal y como lo vemos en las páginas Web que se encuentran en Internet actualmente. Su estructura es la siguiente:

```
a: visite {  
    color:green;  
}
```

Donde con los dos puntos y la palabra visite, indicamos que ocuparemos una clase, ya definida en CSS, con las propiedades que tiene la clase visite. Ejemplo:

❖ Documento pagina24.html

```
<html>  
<head>  
<title>Mi documento</title>  
<link rel="stylesheet" type="text/css" href="style20.css"/>  
</head>  
<body>  
<br><br><br>  
<blockquote><blockquote><blockquote>  
<br><br>  
<a href="/pagina15.html"><h1>Entrar al sitio</h1></a><br>  
<a href="/pagina20.html"><h1>Ejercicio Anterior</h1></a>  
<br><br>  
</blockquote></blockquote></blockquote>  
</body>  
</html>
```

❖ Documento style20.css

```
body {  
    background-image:URL("1.jpg");  
    background-repeat:no-repeat;  
    background-position:right;  
    background-attachment:fixed;  
}  
a: visite {  
    color: "black";  
    font-family:Monotype Corsiva,verdana;  
    font-size: 100%;  
    text-align:justify;  
    color:green;  
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina24.

A continuación en la figura II.24 se muestran los enlaces:



II.24 Enlaces.

✓ Enlace Activo

Con esta propiedad podemos indicar al usuario que página se encuentra visitando en el sitio, por medio un enlace de diferente color se indica que página se encuentra viendo actualmente esta propiedad solo se aplica cuando se utilizan frames, los cuales muestran de un lado la página donde se encuentra el enlace y en otro frame dentro del mismo explorador la página mostrándose. Además de poder indicar el color con el cual deseamos mostrar los enlaces activos por medio del atributo **color**. Su estructura es la siguiente:

```
a: active {
    color:green; }
```

✓ Documento pagina25.html

```
<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style21.css"/>
</head>
<body>
<br><br><br>
<blockquote><blockquote><blockquote>
<br><br>
<a href="/pagina15.html"><h1>Entrar al sitio</h1></a><br>
<a href="/pagina20.html"><h1>Ejercicio Anterior</h1></a>
<br><br>
</blockquote></blockquote></blockquote>
</body>
</html>
```

✓ Documento style21.css

```
body {
    background-image:URL("1.jpg");
    background-repeat:no-repeat;
    background-position:right;
```

```

background-attachment:fixed;
}
a {
    color: "black";
    font-family:Monotype Corsiva,verdana;
    font-size: 100%;
    text-align:justify;
    color:blue;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina25.

A continuación en la figura II.25 se muestran el enlace activo:



II.25 Enlace Activo.

✓ Enlace Decorado

Por medio de esta propiedad podemos decorar el enlace de nuestra página, utilizando cualquiera de las propiedades anteriores al mismo tiempo. Además de poder indicar el color con el cual deseamos mostrar los enlaces activos por medio del atributo color, podemos indicar que existirá una decoración con las propiedades de la clase hover. Su estructura es la siguiente:

```

a: hover {
    color:green;
}

```

❖ Documento pagina26.html

```

<html>
<head>
<title>Mi documento</title>
<link rel="stylesheet" type="text/css" href="style21.css"/>
</head>
<body>

```

```

<br><br><br>
<blockquote><blockquote><blockquote>
<br><br>
<a href="/pagina15.html"><h1>Entrar al sitio</h1></a><br>
<a href="/pagina20.html"><h1>Ejercicio Anterior</h1></a>
<br><br>
</blockquote></blockquote></blockquote></body>
</html>

```

❖ **Documento style22.css**

```

body {
background-image:URL("1.jpg");
background-repeat:no-repeat;
background-position:right;
background-attachment:fixed;
}
a :hover{
color: "black";
font-family:Monotype Corsiva,verdana;
font-size: 100%;
text-align:justify;
color:blue;
}
a :visited{
color: "black";
font-family:Monotype Corsiva,verdana;
font-size: 100%;
text-align:justify;
color:red;
}

```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina26.

A continuación en la figura II.26 se muestran el enlace decorado:



II.26 Enlace Decorado.

✓ **Quitar Subrayado del link**

Con una de las propiedades que tiene CSS, podemos quitar el subrayado que nos presenta el link, aunque no es muy recomendado, debido a que los usuarios en algunas ocasiones no se darán cuenta, que es un enlace para acceder a otra página, hasta que pase el Mouse por encima del enlace, nos daremos cuenta del enlace. Esto se realiza por medio del atributo **text-decoration**, el cual anteriormente ya mencionamos todas sus propiedades que tiene además del atributo none, el cual indica que no se mostrará el subrayado del hipervínculo. Su estructura es la siguiente:

```
a: link {  
    color:green;  
    text-decoration:none; }
```

✓ **Enlace sin visitar**

En esta propiedad podemos definir el color del enlace que se tendrá a la hora que el usuario abra nuestra página por primera vez. Por medio de la clase link. Su estructura es la siguiente:

```
a: link {color:green; }
```

II.14 Visualización en Mayúsculas.

En CSS, podemos hacer que todo el texto que se encuentra en la página, el usuario pueda verlo en letras mayúsculas, aunque nosotros tengamos el texto originalmente en minúsculas, con esta simple propiedad se puede hacer el cambio. La estructura de esta propiedad es:

font-variant:small-caps;

✚ **Nota:** Con esta propiedad aparece todo en mayúsculas, pero las que deben ser minúsculas las visualiza en un menor tamaño.

II.15 Uso de Clases en CSS.

La utilización de clases dentro de CSS, es de mucha utilidad, ya que si queremos que en la página cada párrafo tenga características diferentes, y no las mismas, lo podemos hacer mediante las clases.

Las clases las podemos utilizar en cualquiera de las etiquetas existentes de HTML.

Cuando utilizamos clases dentro de CSS, la estructura general de la clase es:

etiqueta.nombre_clase{.....}

No hay que olvidar que después de la etiqueta se debe indicar por medio de un punto que será una clase y el nombre que le daremos a la clase, para poder mandarla a llamar y que el explorador sepa a que clase nos estamos refiriendo.

Llamándola después con código HTML con la siguiente instrucción:

<etiqueta CLASS=nombre_clase>

Por medio del atributo **CLASS**¹ podemos mandar a llamar la clase desde la hoja de estilo.

¹ Class: Clases en Programación

Las clases permiten especificar reglas distintas para una misma etiqueta, que se comparte de manera distinta en función de la clase o subtipo al que pertenece la etiqueta. Ejemplo:

❖ Documento pagina28.html

```
<html>
<head>
  <title>Mi documento</title>
  <link rel="stylesheet" type="text/css" href="style24.css"/>
</head>
<body>
<br><br><br>
<blockquote><blockquote><blockquote>
<br><br>
  <h1>Mi primera hoja de estilo</h1> <br>
<h1>Imagen de Fondo en CSS Desplazada</h1>
<br><br>
<p class=2>Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.</p>
```

```
<p>Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de ¡un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.</p>
```

```
<p class=2>Lo que acabamos de describir es la forma tradicional de hacer dibujos animados. Hoy día, con los ordenadores, el procedimiento ha cambiado, se ha simplificado; pero la magia de ver cómo un dibujo cobra vida sigue asombrándonos.</p>
```

<h1>LOS PRINCIPIOS DE LA ANIMACIÓN</h1>

```
<p class=3>Mucho antes de que se proyectara la primera película, en el año 1895, ya se sabía cómo dar apariencia de movimiento a un personaje inanimado. Por ejemplo, existía un juguete llamado zoótropo que consistía en una sucesión de imágenes dibujadas en el interior de un cilindro que, al hacerlo girar, producía la impresión de movimiento.</p>
```

```
<p>Los primeros dibujos animados de la historia del cine los hizo Edwin S. Porter, en 1905. Porter fotografió por separado una serie de letras; al proyectar estas imágenes a gran velocidad, las letras parecían correr por la pantalla hasta formar una frase.</p>
```

```
<p class=2>En 1906, James Stuart Blackton, de la productora Vitagraph, experimentó con esta técnica para dotar de movimiento a una serie de sencillos dibujos y, más tarde, de muñecas de juguete. Al año siguiente, probó con objetos modelados en arcilla que iba transformando poco a poco y fotografiando. Así, Blackton se convirtió en el inventor de la animación de muñecos.</p>
```

<h1>LAS PRIMERAS PELÍCULAS DE DIBUJOS ANIMADOS</h1>

<p class=3>Entre 1908 y 1910, un cineasta europeo, el francés Émile Cohl, concentró sus esfuerzos en el desarrollo de estas técnicas de animación. Fue él quien creó la primera serie de dibujos animados; consistía en una sucesión de figuras muy esquemáticas que daban saltos y se caían, asombrosamente, sin hacerse daño.</p>

</blockquote></blockquote></blockquote>

</body>

</html>

❖ Documento style24.css

```
body {
  background-image:URL("1.jpg");
  background-repeat:no-repeat;
  background-position:right;
  background-attachment:fixed;
}
p{
  color: "black";
  font-family:Monotype Corsiva,verdana;
  font-size: 100%;
  text-align:justify;
}
p.2 {
  color: "green";
  font-family:Monotype Corsiva,verdana;
  font-size: 100%;
  text-align:justify;
}
p.3 {
  color: "blue";
  font-family:Monotype Corsiva,verdana;
  font-size: 100%;
  text-align:justify;
}
```

Este ejercicio lo podemos encontrar en la Carpeta HTML Intermedio, CSS, pagina28.

A continuación en la figura II.27 se muestra el uso de clases:

Tony's Doodleji

Mi primera hoja de estilo

Imagen de Fondo en CSS Desplazada

Para hacer dibujos animados, en primer lugar, el artista dibuja cada movimiento del personaje que quiere representar por separado, en una hoja diferente, de celuloide o de acetato, que son dos tipos de plástico transparente. El fondo, que puede ser un bosque o el mar, se pinta sobre un papel y se coloca detrás de la lámina de plástico.

Cuanto mayor sea el número de dibujos, mayor será la calidad de la animación; para producir un solo segundo de movimiento, el artista tiene que crear doce dibujos. Esto quiere decir que, para hacer una película como El libro de la selva, fueron necesarios alrededor de un millón de dibujos! Para llevar a cabo todo este trabajo, se necesita un equipo entero de ilustradores. El siguiente paso es fotografiar los dibujos en el orden en el que se produce la acción; cuando se proyectan a gran velocidad, el personaje parece que cobra movimiento.

Lo que acabamos de describir es la forma tradicional de hacer dibujos animados. Hoy día, con los ordenadores,



II.27 Uso de Clases



Capítulo III

JavaScript

Capítulo III JavaScript

JavaScript es un lenguaje empleado fundamentalmente en el desarrollo de páginas Web Dinámicas.

Mediante este lenguaje sus páginas pueden reconocer y responder a eventos de usuarios, facilitar la navegación en un sitio Web, verificar los datos del formulario, manipular capas, crear animaciones, desarrollar multimedia, etc.

El objetivo de disponer de un lenguaje de programación sencillo es permitir ejecutar determinadas funciones dentro de los documentos HTML. La sintaxis de este lenguaje se basa en el lenguaje JAVA, aunque JavaScript, es mucho más sencillo de manejar.

Las posibilidades que ofrecen los scripts al documento HTML son las siguientes:

- ✓ A través del script es posible modificar dinámicamente el documento a la hora de cargarlo.
- ✓ El script permite comprobar los datos de entrada en los formularios. El formulario solo se envía al servidor, si todos los datos introducidos se encuentran correctos en su forma.
- ✓ Un script puede reaccionar a eventos, como pulsar un hipervínculo con el ratón y provocar determinadas acciones como activar alguna indicación a realizar.

III.1 Vincular Programas Javascript en HTML.

Para vincular programas Javascript en código HTML, basta con seguir algunas reglas. Dentro de los scripts, podemos distinguir dos variantes:

- ✓ **Instrucciones de script que se ejecutan al cargar el documento.** Estas instrucciones se incorporan al documento a través de la etiqueta **<script>**.
- ✓ **Instrucciones de script que reaccionan a un evento.** Estas instrucciones se incorporan en diferentes atributos dentro de las etiquetas.

III.2 Inserción de Código Javascript en un documento HTML.

El código de Javascript, se encuentra insertado dentro del código HTML de las páginas Web, para ello empleamos la etiqueta **<script>**, la cual tiene su etiqueta de cierre **</script>**. La estructura del script es la siguiente:

```
<script language = "Javascript">  
<!--  
    [ Aquí va el código de Javascript ]  
// -->  
</script>
```

Donde script puede tener los siguientes atributos:

- ✓ **src** : Este atributo es opcional, ya que podemos indicar la dirección de un archivo externo al script.
- ✓ **type**: Este atributo es indispensable, ya que indica el tipo de contenido que tendrá el script correspondiente, por ejemplo `visualscript` y `javascript`. El

navegador automáticamente reconoce el tipo de lenguaje script, con la ayuda de este atributo.

- ✓ **language** : Indica el nombre de lenguaje script utilizado.
- ✓ **defer**: Este atributo es opcional, ya que indica al navegador que el script, no generó ninguna acción visible.

La secuencia de la etiqueta **<script>**, puede aparecer en diferentes lugares dentro del documento HTML. Aunque se recomienda que todo el código de Javascript se integre dentro de la etiqueta **<head>** del código fuente o en la parte del **<body>**. Esta etiqueta **<script>** puede repetirse varias veces dentro de un documento HTML.

Ejemplo:

```
<HTML>
<HEAD>
<TITLE> </TITLE>
<SCRIPT LANGUAGE="JavaScript">
alert("¡Hola Mundo!");
</SCRIPT>
</HEAD>
<BODY background= ". / muñeca.jpg">
<P>
Programa 1 en JavaScript
</P>
</BODY>
</HTML>
```

Donde la palabra alert, nos muestra el texto que tiene dentro de los paréntesis, en una ventana de alerta.

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina1.

A continuación en la figura III.1 se muestra la inserción de código JavaScript:



III.1 Inserción de JavaScript.

III.3 Objetos en Javascript.

El núcleo principal de cualquier programa Javascript son los objetos. Este lenguaje puede aplicar métodos objetos sobre los objetos para poder leer y definir propiedades para los mismos.

Estos objetos se clasifican en una determinada jerarquía a la que también podemos llamar modelos de objetos. Algunos de estos son:

- ✓ **navigator:** Este atributo ofrece información sobre el navegador, ya que nos proporciona información acerca del navegador.
- ✓ **window:** Este objeto permite acceder a la ventana del navegador y al documento. A través de este objeto podemos acceder a la ventana de estado de la ventana.
- ✓ **document:** Este objeto engloba todo el documento HTML, se utiliza para escribir algo dentro del documento.
- ✓ **form:** Un formulario puede ser un objeto de un documento, este objeto puede ser inevitable, para acceder a otros elementos de este formulario.
- ✓ **elements:** Es una clase completa de objetos, entre lo cuales se encuentran los formularios como: botones, cuadros de texto, cuadros de lista, etc.
- ✓ **location:** Este objeto contiene la dirección actual del documento
- ✓ **history:** Se especifica las URL visitadas anteriormente por el usuario.

Dentro de Javascript todavía hay definidos otros objetos. Los objetos como links describen los vínculos existentes en un documento.

Todo lo que en Javascript esta relacionado con el documento HTML o el navegador, debemos desarrollarlo a través de objetos en este lenguaje.

III.4 Propiedades

Los objetos en Javascript, tienen propiedades a través de las cuales indicamos determinadas opciones.

El objeto document representa al documento mostrado en el navegador. Este objeto tiene una serie de propiedades como lo es el color de fondo. Para colocar la propiedad de color de fondo en el documento utilizamos la siguiente sentencia:

```
color = document.bgcolor;
```

El acceso a esta propiedad es de forma muy sencilla, ya que debemos indicar el objeto al que pertenece la propiedad. Al nombre del objeto debemos añadir el nombre de la propiedad. Ambos nombres deben estar separados por un punto. Para definir el color utilizamos:

```
document.bgcolor= "aqua";
```

III.5 Métodos de Javascript

Javascript soporta una serie de métodos concretos que se pueden aplicar a los objetos. Por ejemplo:

```
document.write("Hola Mundo");
```

En esta aplicación utilizamos el objeto **document**. El nombre que se escribe a continuación no corresponde a un atributo sino al método **write**, en el cual indicamos el texto que deseamos visualizar en la página. Una cosa importante que tenemos que resaltar es que todos los parámetros se escriben entre paréntesis.

III.6 Estructura de un Programa Javascript

La sintaxis de Javascript es muy similar a la sintaxis utilizada en lenguaje C, algunas de ellas son:

- ✓ Toda instrucción que se encuentra seguida de otra instrucción debe finalizarse con un punto y coma (;).
- ✓ La única instrucción que no necesita punto y coma es la última instrucción de un bloque o función, que no va seguida de ninguna otra instrucción.
- ✓ Para incluir varias instrucciones dentro de una función o de un bloque, es necesario incluirlo entre llaves ({}).
- ✓ Si necesitamos incluir un comentario dentro del programa de Javascript, cuando comencemos el comentario debemos escribir siempre // seguido del comentario a realizar, para el caso de que solo sea una línea. Si necesitamos comentar más de una línea de utiliza /* comentario */. Este texto escrito seguido de cualquiera de estos dos signos, interpretara que es un comentario y saltará a la siguiente línea de ejecución del programa.

III.7 Definir Funciones.

Una función no se ejecuta hasta que no la activemos nosotros mismos. La estructura de una función es la siguiente:

```
function name()
{
    //Instrucciones de la Función;
    return;
}
```

La función se introduce con la palabra **function**. A cada función debemos asignarle un nombre único. EL nombre debe ir seguido de paréntesis, dentro de los cuales escribiremos los parámetros.

Cuando la función no posea parámetros, los paréntesis deben permanecer vacíos. Las instrucciones de la función debemos escribirlas dentro de las llaves ({ }). Para finalizar la función es necesario incluir una instrucción **return** en el código de Javascript.

Ejemplo:

```
<HTML>
<HEAD>
<SCRIPT LANGUAGE="JavaScript">
function Saludo()
{
```

```

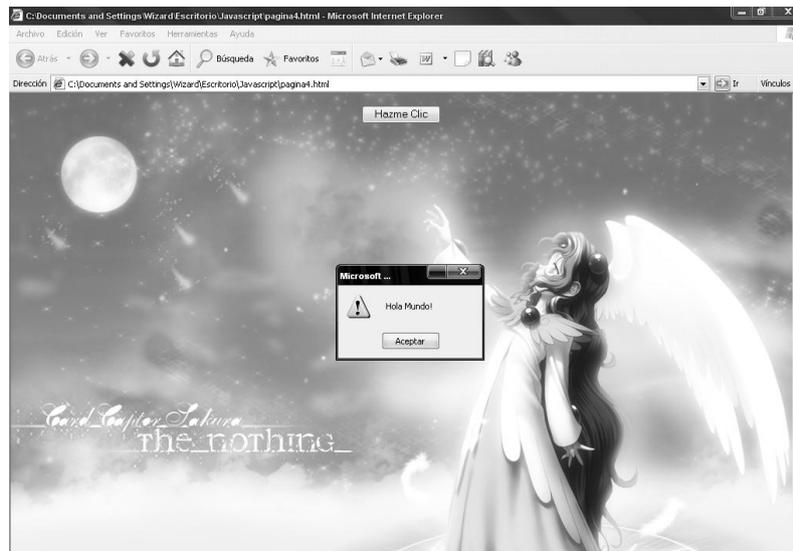
alert("Hola Mundo!");
}
</SCRIPT>
</HEAD>
<BODY background= "./muñeca.jpg">
<P ALIGN="CENTER">
<FORM>
<INPUT TYPE="button" VALUE="Hazme Clic" onClick="Saludo();">
</FORM>
</P>
</BODY>
</HTML>

```

🚦 **Nota:** El nombre de una función se distingue de mayúsculas y minúsculas.

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina2.

A continuación en la figura III.2 se muestra el uso de funciones:



III.2 Uso de Funciones.

III.8 Variables y Constantes en Javascript.

En JavaScript podemos usar constantes y variables. Las variables son áreas de memoria que tienen un nombre asignado y cuyo valor podemos modificar.

🚦 Constantes

Las constantes pueden ser números que aparecen en los programas de JavaScript, además de indicarlos directamente con las instrucciones correspondientes.

Javascript trabaja con varios tipos de constantes que se pueden indicar directamente en la instrucción. Ejemplo:

Costo = 55+100;

✚ Valores y Tipos de Datos.

Javascript contiene muy pocos tipos de datos entre los cuales se encuentra:

Tipo	Características
Número	No hace distinciones entre números reales y números enteros
Valores Lógicos (Booleanos)	True o False
String (Cadena de Caracteres)	Las cadenas de caracteres o strings se escriben en Javascript entre comillas simples o comillas dobles. Como "Esto es un texto"
Null	Es un valor de espacio

✚ Caracteres Especiales en las Cadenas.

JavaScript cuenta con algunos caracteres especiales que cumplen una función especial dentro de las cadenas de caracteres como:

Símbolo	Significado
<code>\b</code>	Espacio en Blanco
<code>\f</code>	Avance de Línea
<code>\n</code>	Línea Nueva
<code>\r</code>	Retroceso de Carro
<code>\t</code>	Carácter de Tabulación

Otro caso que puede producirse es que dentro de una cadena de caracteres aparezcan las comillas normales o una comilla simple. Este carácter cerraría la cadena de caracteres. Para incluirlo dentro de la cadena hay que anteponer una barra invertida o backslash. Ejemplo:

Cadena = "El hombre dijo \ "Vaya Jugada \ " ;

✚ Definir una Variable.

Las variables permiten guardar un valor bajo un nombre y utilizarlo en el programa. Para poder utilizar una variable en Javascript, antes es preciso declararla. Las variables se pueden declarar asignándoles un valor como:

Base= 12;

Altura= 10;

O bien colocándoles la palabra clave "**var**", antes del nombre de la variable, como:

var Texto;

var x = 24;

✚ **Nota:** Cada línea de la instrucción debe finalizarse con punto y coma.

Ejemplo 1:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var nom;
nom=prompt("Escribe tu nombre "," ");
alert("Mucho gusto "+ nom);
</SCRIPT>
<body background="./muñeca.jpg">
</body>
</HTML>
```

Donde el método prompt, nos muestra una ventana en la cual podemos escribir el contenido que tendrá la variable, para después mostrar ésta en la página.

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina3.

A continuación en la figura III.3 se muestra la definición de variables:



III.3 Definición de Variables.

Reglas para el Nombre de las Variables.

- ✓ La primera regla que debemos tener en cuenta es elegir nombres adecuados (lógicos) para las variables, es decir que los nombres sean comprensibles, también fuera del programa.
- ✓ Una variable siempre debe empezar con una letra.
- ✓ El nombre de la variable no puede contener espacios en blanco, ni caracteres especiales, como los acentos.
- ✓ El único carácter especial permitido en las variables es el guión bajo.
- ✓ El nombre no puede contener más de 32 caracteres y no puede coincidir con ninguna de las palabras reservadas.
- ✓ Los nombres de las variables siempre se distinguen entre mayúsculas y minúsculas. Por ejemplo cadena no es lo mismo que Cadena. Estos nombres de variables son distintas.

Palabras Reservadas.

Algunas de las palabras reservadas que tiene Javascript son:

Palabras		Reservadas	
Abstract	extends	interface	synchronize
Boolean	false	long	this

break	Final	native	thrown
byte	finally	new	throws
Case	float	null	translet
cath	for	Package	true
char	function	private	try
class	goto	protecte	Var
const	if	public	Void
continue	implements	return	While
default	import	short	With
do	in	static	
double	Instanceof	super	
else	int	switch	

Validez de los nombres de Variables.

La validez de una variable se puede definir por el tipo de declaración como:

- ✓ Si una variable se declara dentro de una función con la palabra clave “**var**”, la variable solo es válida localmente dentro de la función. Ejemplo:

Var Area=0;

- ✓ Si una variable se declara, dentro o fuera de una función, asignándole un valor, tiene validez global. Ejemplo:

Sueldo = 3000;

- ✓ Se recomienda declarar todas las variables globales y sus valores al principio del script.

III.9 Operadores de Comparación

En las instrucciones que se suelen utilizar con **if**, siempre se utilizan operadores comparativos. Estos operadores se encargan de comparar dos valores, devolviendo un valor lógico como true o false.

JavaScript soporta los siguientes operadores de comparación:

Operador	Significado
==	Igual a
!=	Distinto de
>=	Mayor o igual que
<=	Menor o igual que
<	Menor que
>	Mayor que

Un ejemplo de esto es el operador de comparación:

if(Edad==20)

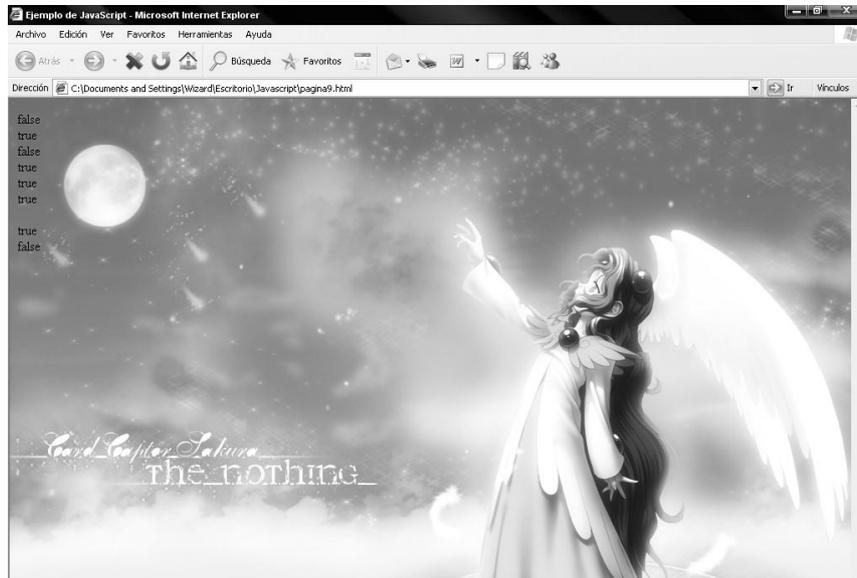
return 1;

Ejemplo:

```
<html>
<head>
  <title>Ejemplo de JavaScript</title>
</head>
<body>
<script language="JavaScript">
  a = 8;
  b = 3;
  c = 3;
  document.write(a == b);document.write("<br>");
  document.write(a != b);document.write("<br>");
  document.write(a < b);document.write("<br>");
  document.write(a > b);document.write("<br>");
  document.write(a >= c);document.write("<br>");
  document.write(b <= c);document.write("<br><br>");
  document.write(3 == "3");document.write("<br>");
  document.write(3 === "3");document.write("<br>");
</script>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina9.

A continuación en la figura III.4 se muestran los operadores de comparación:



III.4 Operadores de Comparación.

III.10 Operadores de Cálculo.

Los cálculos numéricos requieren operadores de cálculo de la forma:

```
var Salario= 30*150;
```

```
var Impuesto= Salario*.15;
```

var Pago = Salario-Impuesto;

En JavaScript son válidas las expresiones de cálculo habituales. Los operadores de cálculo son los siguientes:

Operador	Significado
+	Adición ($a=a+b$)
-	Sustracción ($a=a-b$)
*	Multipliación ($a=a*b$)
/	División ($a=a/b$)
%	Módulo ($a=a\%b$)

Estos operadores los podemos utilizar en todas las instrucciones de Javascript, de manera normal, aunque también existe la posibilidad de combinarlos con el operador de asignación (=).

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
/*Programa que sirve para calcular
el área de un rectángulo */
var base,altura;
base=prompt("Escribe la base del Rectángulo","");
altura=prompt("Escribe la altura del Rectángulo","");
alert("El área del Rectángulo es = "+(base*altura));
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina4.

A continuación en la figura III.5 se muestran los operadores de cálculo:



III.5 Operadores de Cálculo

III.11 Operadores de Incremento y Disminución.

Para incrementar o disminuir los valores de las variables se pueden utilizar los operadores de incremento y disminución. Como:

Operador	Significado
----------	-------------

++i	Incrementa i
--i	Disminuye i
*=	Incrementa la Multiplicación
/=	Incrementa la División
%=	Incrementa la División Modular.

También se puede combinar el operador de asignación con los operadores de cálculo o con los de incremento-disminución y se puede utilizar la sintaxis abreviada, como en lenguaje C. Como:

Sintaxis Abreviada	Equivale a:
a-=b	Equivale a: a=a-b
a*=b	Equivale a: a=a*b
a/=b	Equivale a: a/b
a%=b	Equivale a: a%b (módulo)
a=++i	Equivale a: Incrementa i en 1 y asigna valor a
a=i++	Asigna a "a" el valor de i e incrementa a continuación i en 1
a=-i	Reduce i en 1, y asigna el valor a
a=i-	Asigna a "a" el valor de i y reduce i en 1
a+=b	Equivale a: a=a+b

III.12 Operandos Lógicos.

Para las evaluaciones más complejas requerimos de operadores lógicos. JavaScript dispone de los siguientes operadores lógicos:

Símbolo	Operador
 	Conjunción (O)
>>	Desplazamiento de Bits hacia la Derecha
<<	Desplazamiento de Bits hacia la Izquierda
&	Máscara de bits con conjunción Y
 	Máscara de bits con conjunción O
^	Máscara de bits con conjunción O exclusiva (XOR)
&&	Conjunción (Y)

En las cadenas de caracteres se puede utilizar el operador de adición para realizar un encadenamiento (Concatenación). Las cadenas de caracteres se encadenan entre sí con el signo "+". Ejemplo:

```
var nombre = "Mi" + "Nombre";
```

III.13 Prioridades entre Operadores.

Si una instrucción contiene varios operadores, estos se tendrán que evaluar de acuerdo con determinadas prioridades. Estas prioridades son las que ya conocemos de las fórmulas matemáticas y la regla “El punto va antes que el guión”.

A continuación se muestra una tabla de operadores por orden (ascendente) de prioridad. Las primeras que se enlistan son las de prioridad más baja:

Operador	Orden
Coma	,
Asignaciones	= += -= * =/ =% ==<< ==>> ==>>> =& ^= = \ =
Condición	?
O lógico	
Y lógico	&&
O de bits	
Xor de bits	^
Y de bits	&
Igual, distinto de	, == !=
Relacional	< <= >>=
Desplazamiento de Bits	<< >> >>>
Adición / Sustracción	+ -
Multiplicación/División	* / %
Negación / Incremento	! - ++ --
Call, Member	() []

También se tiene la posibilidad de escribir las expresiones entre paréntesis e influir en el orden de la evaluación.

III.14 Estructuras de Control

Las Estructuras de control permiten incluir bucles y ramificaciones en los programas. En JavaScript existen varias estructuras de control.

III.15 Expresiones de Condición.

A veces es necesario colocar una asignación que defina un valor dependiendo de una condición. Estas asignaciones las podemos realizar con ayuda de las expresiones condicionales. Estas expresiones de condición disponen de dos valores y una expresión comparativa. Ejemplo:

(Condición) valor1:valor2

Si la condición se cumple, a la expresión se la asigna el valor de 1. En otro caso toma el valor de 2.

Estas expresiones condicionales las podemos utilizar como ramificaciones, en el momento en que se requiera. Ejemplo:

Entrada=(Edad>=18)

III.16 Estructura IF

Al igual que en el Lenguaje C, tenemos la estructura if. Esta estructura la utilizamos cuando necesitamos utilizar una condicional para que efectuemos ciertas instrucciones, a la hora que toma la decisión según las especificaciones que le demos. Su sintaxis es la siguiente:

```
if(condición)
{
    Comando cuando se cumple la condición
}
```

Ejemplo:

```
if(Edad<=20)
{
    alert("Seguramente te encuentras estudiando")
}
```

 **Nota:** Si dentro del if, se encuentran varias instrucciones, estas deben finalizarse con un punto y como “;”.

Hay ocasiones en que es necesario realizar una acción y si esta no se cumple realizar otra; esto se hace por medio de la condición else. Ejemplo:

```
if(condición)
{
    Comandos cuando se cumple la condición
}

else
{
    Comandos cuando no se cumple la condición
}
```

Ejemplo:

```
<html>
<head>
<title>      </title>
<SCRIPT LANGUAGE="JavaScript">
var num;
num=parseFloat(prompt("Escribe un número",""));
if(num==100)
{
    alert("El número que has escrito es 100");
}
else
{
```

```

alert("El número que has escrito no es 100");
}
if(num>0)
{
alert("El número que has escrito es positivo");
}
else
{
alert("El número es negativo ó 0");
}
</SCRIPT>
</head>
<body>
</body>
</HTML>

```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina5.

A continuación en la figura III.6 se muestra la estructura if:



III.6 Estructura IF.

✚ **Nota:** Si la afirmación y la negación solo contiene un instrucción podemos omitir el uso de las llaves ({ }) para escribir las instrucciones. Pero se aconseja que siempre se coloquen.

III.17 Ciclo For.

El ciclo for permite repetir determinadas series de instrucciones de la manera definida. El número de repeticiones las indicamos por medio de un contador. Su estructura es la siguiente:

```

for(var contador =1; contador<=100; contador++)
{
    Instrucciones
}

```

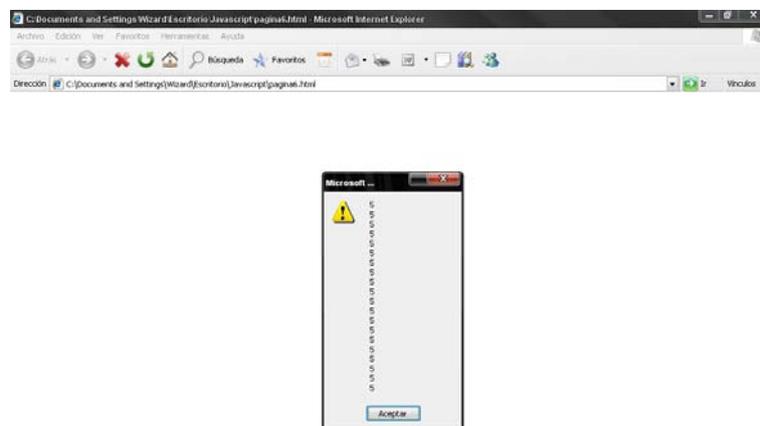
En la primera instrucción el ciclo contiene una variable contador a la que podemos llamar, a la cual se le asigna el valor de 1, cuando aparece por primera vez. Cada vez que recorramos el ciclo, el tercer parámetro de la estructura aumenta en 1 (contador++). La condición final (contador<=100), hace que el ciclo finalice al llegar a las 100 repeticiones.

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var texto;
var num;
var salida="";
texto=prompt("Escribe un texto","");
num=prompt("¿Cuántas veces quieres que lo repita?","");
num=parseInt(num,10);
for(i=1;i<=num;i++)
{
salida=salida+texto+"\n";
}
alert(salida);
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina6.

A continuación en la figura III.7 se muestra el ciclo for:



III.7 Ciclo For.

III.18 Ciclo While.

Con el ciclo while podemos repetir varias veces las instrucciones de Javascript. Las repeticiones se interrumpen cuando no se cumpla la condición del ciclo. Su estructura es la siguiente:

```
while(condición)
{
    Instrucciones
}
```

```
}
```

La condición se comprueba en la cabecera del ciclo de repetición. Mientras la condición se cumpla, se repiten las instrucciones de ciclo. Es importante también incluir una condición que finalice el ciclo después de un número determinado de repeticiones.

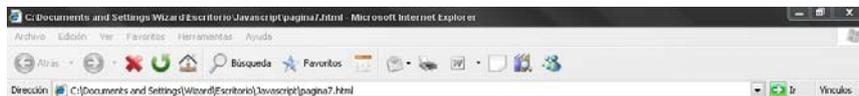
Para impedir que el bucle se convierta en un ciclo infinito, debemos introducir una variable la cual aumentará su valor de 1 en 1, cada vez que se recorra el ciclo. Cuando el valor de la variable es mayor al de la condición, se termina de repetir el ciclo. El interprete continúa entonces ejecutando las demás instrucciones que vienen a continuación del ciclo.

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var x=0;
while(x<6)
{
alert("El valor de x es= "+x);
x=x+1;
}
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina6.

A continuación en la figura III.8 se muestra el ciclo while:



III.8 Ciclo While.

Algo muy importante que debemos recordar es que al introducir una condición debemos asegurarnos que esta tenga un fin. De lo contrario el usuario solo podrá cancelar la ejecución del programa en el navegador de una manera muy violenta. Lo cual nunca es aconsejable. Para esto se aconseja, que durante la terminación del ciclo se coloque la palabra **break**, al finalizar el ciclo, de esta manera nos aseguramos que termine solo.

III.19 Sentencia do.....while

La sentencia do...while es muy similar a la sentencia while, con la diferencia de que la condición se comprueban solo tras la ejecución de las sentencias; así como una sentencia while se ejecuta cero o más veces, mientras que una sentencia do...while se ejecuta una o más veces.. Su estructura es:

```
do
{
    Sentencias
}while(condición);
```

III.20 Sentencia Switch

La sentencia switch, la utilizamos, cuando necesitamos trabajar con ciertos casos, los cuales si utilizamos la sentencia if, puede ser muy grande el código además de provocar alguna confusión durante la ejecución del programa, para esto se utiliza la sentencia switch, con la cual podemos evaluar varios casos, sin la necesidad de tanto código, y con función del explorador. Para terminar cada caso, de utiliza la palabra **break**; la cual indica, que ha encontrado la condición, por lo cual ya no es necesario seguir evaluando las demás condiciones, de lo contrario seguirá evaluándolas, en caso de no encontrarla se pasará a la sentencia default, para avisar que no hay ninguna condición que se cumpla. Su estructura es:

```
switch(expresión){
case etiqueta:
sentencias
break;
case etiqueta2:
sentencias
break;
.....
default:
sentencias
}
```

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var num,bien,notanum,notacual;
num=parseInt(prompt("Escribe el número total de preguntas",""));
bien=parseInt(prompt("Escribe el número de respuestas acertadas",""));
notanum=parseInt(10*bien/num);
switch(notanum)
{
case 0:
notacual="Muy Deficiente";
break;
case 1:
notacual="Muy Deficiente";
break;
case 2:
```

```
notacual="Deficiente";
break;
case 3:
notacual="Deficiente";
break;
case 4:
notacual="Insuficiente";
break;
case 5:
notacual="Suficiente";
break;
case 6:
notacual="Bien";
break;
case 7:
notacual="Notable";
break;
case 8:
notacual="Notable";
break;
case 9:
notacual="Excelente";
break;
case 10:
notacual="Matricula de Honor";
break;
}
alert("La nota cualitativa es "+notacual);
</SCRIPT>
</HTML>
```

Donde:

- ✓ **parseInt:** Es una función devuelve un valor numérico y espera un argumento también numérico. Además que esta función devuelve valores enteros.

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina8.

A continuación en la figura III.9 y III. 10 se muestra la sentencia do while:



III.9 Sentencia Do while



III.10 Sentencia Do While.

III.21 Comentarios

En JavaScript se pueden escribir comentarios de dos tipos: de una sola línea, van precedidos por dos barras inclinadas hacia la derecha (*//*) y comentarios que pueden ocupar varias líneas, determinadas por los símbolos */* comentario */*. Hay que señalar que todo lo que este después de estos símbolos o entre los símbolos no será visualizado en el explorador. **Ejemplo:**

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
/* Programa que utiliza una variable explícita
y dos implícitas */
var Expli;
var pi=3.141592;
var radio=7;
Expli=pi*radio*radio;
alert("Área del Círculo = "+Expli); // calcula el área del círculo
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina10.

A continuación en la figura III. 11 se muestran los comentarios:



III.11 Comentarios.

III.22 Utilización del Objeto Document.

El objeto document posee una serie de métodos y propiedades muy interesantes, sin embargo solo se usan dos métodos comúnmente los cuales son write y writeln, los cuales tienen la función de escribir dentro del navegador.

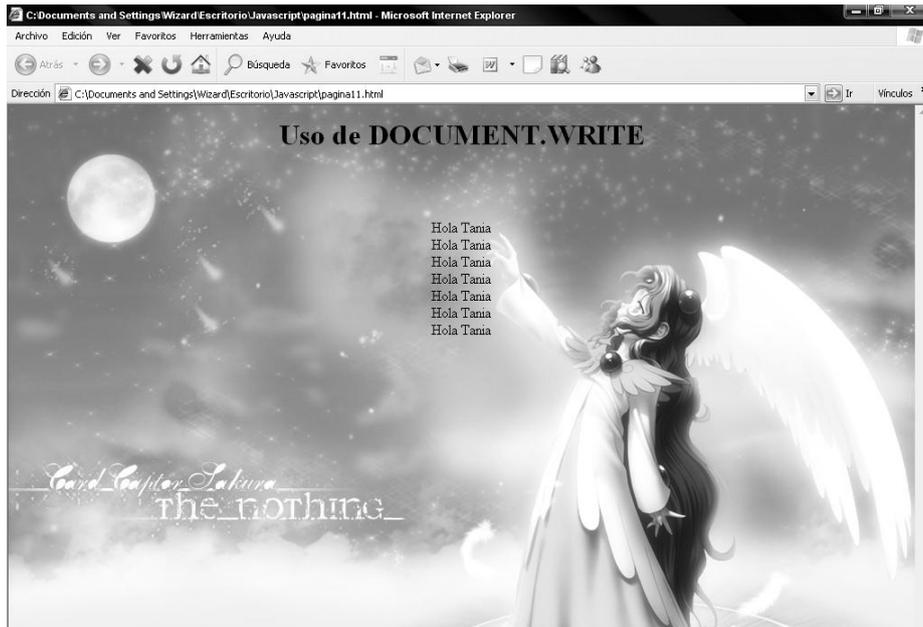
El argumento que deben recibir es una cadena o concatenación de cadenas, que serán mostradas dentro de HTML.

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var nom;
nom=prompt("Escribe tu nombre","");
var salida="";
for(i=2;i<=14;i=i+2)
{
salida=salida+"<BR>"+ " Hola "+nom;
}
document.write("<H1><P align='CENTER'>Uso de DOCUMENT.WRITE</H1>");
document.write("<BR><BR>");
document.write(salida);
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina11.

A continuación en la figura III. 12 se muestra el objeto document:



III. 12 Objeto Document

III.23 Utilización del Objeto Window.

El objeto window tiene una serie de métodos útiles para crear nuevas ventanas y diálogos. Uno de estos métodos es alert, el cual ya hemos estado utilizando a lo largo de este capítulo.

Método Alert.

Este método muestra una caja de dialogo, en una especie de mensaje de alerta.

Ejemplo:

```
<HTML>
<SCRIPT LANGUAGE="JavaScript">
var num1,num2;
num1=prompt("Escribe un número","");
num2=prompt("Escribe otro número","");
alert("La suma es "+(num1+num2));
</SCRIPT>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina12.

A continuación en la figura III. 13 se muestra el objeto window:



III.13 Objeto Window.

III.24 Movimiento del Titulo

Por medio de JavaScript podemos comenzar a tener movimientos dinámicos dentro de una página un ejemplo de ello es hacer que el título que se presenta en el navegador pueda tener un movimiento específico.

Ejemplo:

```
<HTML>
<HEAD>
  <title>Ejemplo de JavaScript</title>
  <script LANGUAGE="JavaScript">
<!--
  var i=1;
  var titulooriginal = document.title;
  function cambiarTitulo()
  {
    document.title = titulooriginal.substring(0,i);
    i++;
    if (i>titulooriginal.length) i=0;
    setTimeout('cambiarTitulo()',100);
  }
  cambiarTitulo();
  //-->
</script>
</HEAD>
<BODY>

</script>
</BODY>
</HTML>
```

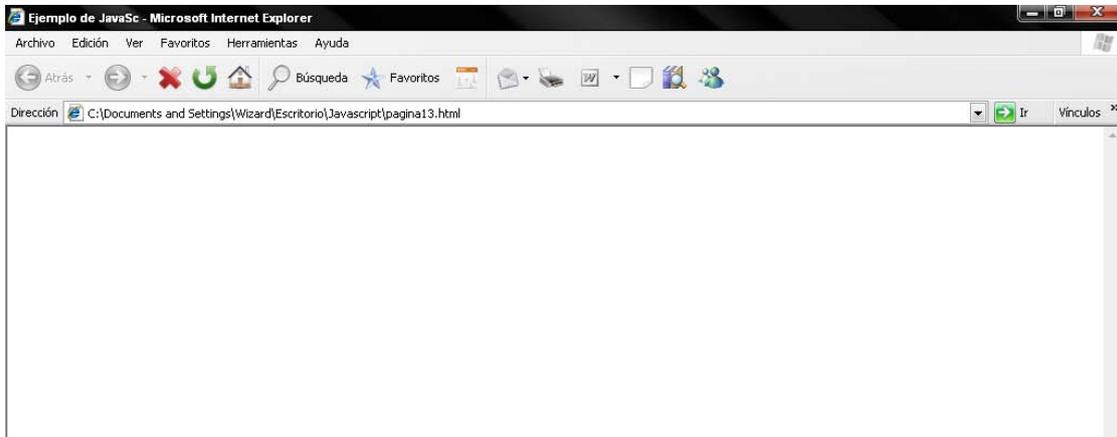
Donde:

Los únicos métodos que no hemos conocido a lo largo de este capítulo son el método title, el cual se encarga de asignarle todos los movimientos a la etiqueta <title>, la

función length, para el tamaño de la cadena y por último setTimeout, El cual se encarga de especificar el tiempo en que se recorrerá el texto.

Este ejercicio lo podemos ver en la carpeta de HTML Intermedio, Javascript, pagina13.

A continuación en la figura III. 14 se muestra el movimiento del título:



III.14 Movimiento del Título.

III.25 Consultar los Valores de Entrada en JavaScript

Para consultar un valor determinado en Javascript, podemos incluir en el documento un formulario con un cuadro de texto. Algunas veces se prefiere no utilizar un formulario, para consultar algunos de los valores introducidos por el usuario. En este caso lo mejor es utilizar la función prompt. Su estructura es la siguiente:

Area = Prom.(texto,valor_predefinido);

Esta entrada ya la hemos utilizado a lo largo del capítulo con la cual se ha demostrado su funcionamiento.

A continuación en la figura III. 15 se muestran los valores de entrada:



III.15 Valores de Entrada



Capítulo IV

DHTML Dinámico

Capítulo IV HTML Dinámico.

Hay que señalar que el lenguaje denominado DHTML¹, no se trata de una extensión de HTML.

HTML Dinámico es la unión de HTML, hojas de estilo y JavaScript. El uso de estas tres tecnologías nos permiten introducir movimientos dinámicos a una página para volverla más atractiva.

Desarrollar una página Web con HTML Dinámico no es difícil, para ello se recomienda desarrollar algunos bocetos iniciales, con lo que se tiene planeado desarrollar la página, para después pensar en la mejor manera de implementarlo con la ayuda de HTML, CSS y el desarrollo de código en JavaScript necesario para gestionar todo.

Muchas personas que se dedican a trabajar con páginas dinámicas prefieren definir dentro del mismo documento Web, todas las líneas de estilo y otras prefieren definir primero todas las hojas de estilo. Aunque se recomienda tener todas las hojas de estilo, externas para después enlazarlas todas directamente desde el código HTML.

HTML Dinámico está basado en los lenguajes considerados hasta ahora como estándar de HTML. Pero incluye funciones que van más allá de las características de estos lenguajes de programación y nos permiten mostrar contenidos dinámicos en las páginas Web.

El núcleo común de estos lenguajes son las hojas de estilo utilizadas para definir el diseño del documento. A través de plantillas de diseño, los elementos del documento se pueden posicionar de manera absoluta dentro de la página.

HTML Dinámico soporta bases de datos y controles multimedia. Contiene un modelo dinámico, el cual permite integrar scripts en las páginas y modificar las propiedades de los estilos y los elementos de la página. Con las funciones de base de datos integradas los diseñadores de páginas pueden acceder directamente a las bases de datos.

Las plantillas de estilo que contiene Javascript permiten manipular las propiedades de la plantilla. Con las fuentes Dinámicas para poder crear efectos especiales en el documento.

Si queremos crear contenidos dinámicos en las páginas Web, para todas las familias de navegadores, se recomienda utilizar algunas de las herramientas existentes como Dreamweaver. Debido a que sin alguna de estas herramientas, necesitaremos hacer una página para cada versión de navegador, ya que cada uno de estos navegadores cuenta con propiedades diferentes, las cuales no se pueden visualizar en otro explorador, si fueron diseñados para uno en específico. Por lo que la herramienta Dreamweaver genera automáticamente las páginas HTML necesarias para cada tipo de navegador.

¹ DHTML: Dinamic HTML

IV.1 Desactivar las Barras de Desplazamiento.

Los exploradores activan automáticamente una barra de desplazamiento en el margen derecho de la ventana del navegador. Existen casos en los que el autor de la página Web prefiere eliminar estas barras de la pantalla. Para ello se integra el atributo scroll en la etiqueta <body>, con la cual podemos suprimir la barra de desplazamiento. Su estructura es la siguiente:

```
<body bgcolor= "blue" scroll= "no">
```

✚ **NOTA:** Al atributo scroll, se le asigna el valor de no, para evitar que se muestren las barras de desplazamiento.

IV.2 Trabajar con Eventos.

Los navegadores pueden trabajar con eventos. Un evento se puede producir al cargar una página, pulsar un botón, cerrar un formulario, etc.

IV.3 Vincular un evento a un Procedimiento

En JavaScript, el tratamiento de eventos se debe vincular al elemento directamente a través de los atributos del evento.

Un ejemplo de esto es una página Web simple con un botón. Cuando el usuario pulse el botón, produce el evento clic.

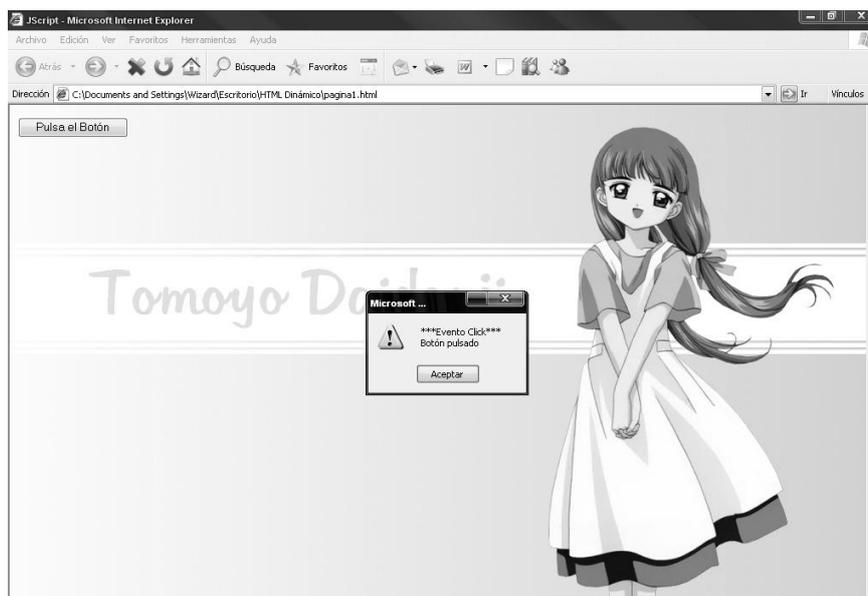
Ejemplo:

```
<html>
<head>
<title>JScript</title>
</head>
<body background="/.1.jpg">
<script language="JavaScript">
<!--
function Pulsar_Boton()
{
  alert ("***Evento Click***\nBotón pulsado")
}
//-->
</script>
<form>
<p><input type="button" name="TestBut"
  onClick="Pulsar_Boton()" value="Pulsa el Bot&ocute;n"></p>
</form>
</body>
</html>
```

Para vincular el procedimiento al botón, utilizamos la etiqueta <input>, incluyendo el atributo onClick. A este atributo asignamos comandos de Javascript, para poder activar la función de Javascript.

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina1.

A continuación en la figura IV. 1 se muestra como vincular un evento a un procedimiento:



IV.1 Vincular un Evento a un Procedimiento.

IV.4 Barra de Estado Desplazable (scrollbar) A través del evento timer.

En algunas páginas Web, la barra de estado se utiliza para emitir determinada información. A veces, el texto de la barra de estado del navegador se mueve para llamar la atención del usuario.

Para reproducir un movimiento de texto en la barra de estado, el script correspondiente recurre a la propiedad status del objeto window. Ejemplo:

```
window.status="Mi texto"; //en la barra de estado
```

No obstante el navegador suele sobrescribir este texto con sus propios mensajes de estado. Para que el propio texto permanezca visible, habría que emitirlo cíclicamente en la barra de estado. Esta repetición cíclica la podemos conseguir a través de un evento con temporizador. El navegador ofrece para ello el método setTimeout del objeto Windows. Este método lo podemos activar de la siguiente manera:

```
window.setTimeout("scroll ( )",speed);
```

Donde el primer parámetro es una cadena de caracteres que indica la función dentro del script. Después transmitimos el nombre de la función scroll al evento setTimeout. En donde este método suspende la ejecución del script durante el tiempo de espera indicando el segundo parámetro en milisegundos. Una vez transcurrido el tiempo de espera indicado, el navegador activa de nuevo la función correspondiente.

Ejemplo:

```
<html>  
<head>  
<title>Barra de estado dinámica</title>
```

```

</head>
<body bgcolor="white" onload="scroll()" background= "./sakura.jpg">
<script language="JavaScript">
<!--
// Inicializar variable
var text = "Barra de estado animada";
var txtlen = text.length;
var speed=25;
var ancho=100;
var pos=1-ancho;
function scroll()
{
  pos++; // inicializar posición para la emisión
  var strScroll=""; // Texto a emitir
  // Mover ahora hacia la derecha a través del string
  if (pos==txtlen)
  {
    pos=1-ancho; // empezar de nuevo una vez procesado el string
  }
  // Scrolling
  if (pos<0)
  {
    // añadir caracteres en blanco si es necesario
    for (var i=1; i<=Math.abs(pos); i++)
    {
      strScroll=strScroll+" ";
    }
    strScroll=strScroll+text.substring(0, ancho-i+1);
  }
  else
  {
    strScroll=strScroll+text.substring(pos,ancho+pos);
  }
  window.status = strScroll; // en barra de estado
  setTimeout ("scroll()", speed);
}
//-->
</script>
<p>Ejemplo de una barra de estado animada</p>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina2.

A continuación en la figura IV. 2 se muestra la barra de estado despegable:



IV. 2 Barra de Estado Despegable.

IV.5 Barra de Estado Parpadeante.

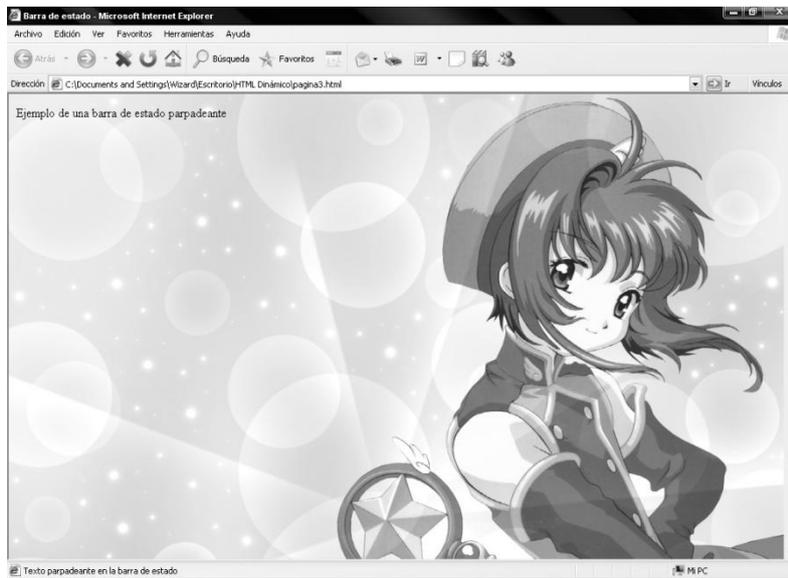
Para insertar una barra de estado parpadeante, lo podemos conseguir con un sencillo truco lo podemos conseguir fácilmente, utilizando algunas técnicas para escribir el texto cíclico, al texto original y un carácter en blanco en la barra de estado. El código es el siguiente:

```
<html>
<head>
<title>Barra de estado</title>
</head>
<body bgcolor="white" onload="parpadeo()" background="/sakura.jpg">
<script language="JavaScript">
<!--
// Inicializa variable
var text = "Texto parpadeante en la barra de estado";
var speed=500;
var state = 0;
function parpadeo()
{
if (state == 0)
{
window.status = text; // Texto de la barra de estado
state = 1;
setTimeout ("parpadeo()", 2 * speed);
}
else
{
window.status = ""; // Borrar barra de estado
state = 0; // reset
setTimeout ("parpadeo()", speed);
}
}
}
//-->
```

```
</script>
<p>Ejemplo de una barra de estado parpadeante</p>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina3.

A continuación en la figura IV. 3 se muestra la barra de estado parpadeante:



IV.3 Barra de Estado Parpadeante.

IV.6 Ajustar el Valor de un Atributo.

Con las propiedades de HTML y Javascript, se puede acceder a los valores de los atributos en las etiquetas. Esto permite modificar dinámicamente el destino de un hipervínculo, modificar el color de fondo de una página Web o una imagen cualquiera. A través de un ejemplo podemos especificar esto, con la utilización de 4 botones, los cuales definirán el color de fondo del documento.

Para definir el color de fondo se utiliza la siguiente instrucción de Javascript:

```
document.bgColor= "#FF0000";
```

La instrucción define el valor del atributo bgColor asignándole un valor de color. Este valor de color lo podemos indicar por su nombre en inglés o de la forma hexadecimal. Su estructura es la siguiente:

```
function chBGColor(color)
{
    //Define el color de fondo
    document.bgColor=color;
}
```

En Javascript se hace la distinción entre mayúsculas y minúsculas. Los nombres de atributos se deben escribir en Javascript de acuerdo con las normas establecidas por el modelo de objetos. En el caso del atributo bgColor, en el nombre hay una letra que se debe escribir en Mayúscula, aunque no lo creamos, esta simple letra nos puede causar una pérdida de tiempo considerable, hasta poder identificarla como una fuente de error.

Para modificar el color al pulsar un botón del documento HTML, se puede utilizar la siguiente instrucción:

```
<input type= "button" value= "verde" onClick = "chBGColor(, #00FF00');">
```

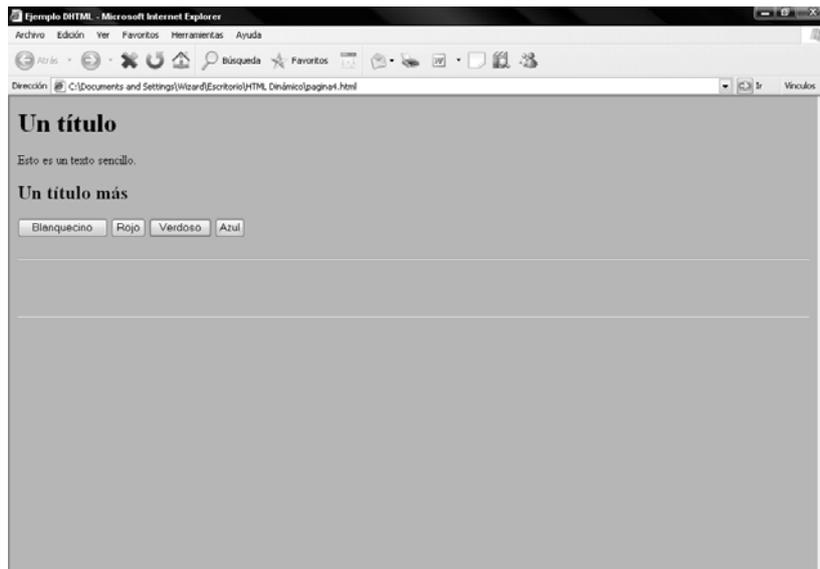
Al evento onClick se le asigna la activación de función chBGColor. El evento provoca a continuación el cambio de color a través del procedimiento.

Ejemplo:

```
<html>
<head>
<title>Ejemplo DHTML</title>
</head>
<script language="JavaScript">
<!--
function chBGColor(color)
{ // define color de fondo
  document.bgColor = color;
}
//-->
</script>
<body bgcolor="white" onload="tick();">
<h1 id="h1">Un título</h1>
<p>Esto es un texto sencillo.</p>
<h2 id="h2">Un título más</h2>
<form>
<input type="button" value="Blanquecino" onclick="document.bgColor = '#FFFFFF';">
<input type="button" value="Rojo" onclick="document.bgColor = '#FF0000';">
<input type="button" value="Verdoso" onclick="chBGColor('#00FF00');">
<input type="button" value="Azul" onclick="chBGColor('blue');">
</form>
<hr>
<div id="clock" align="center"
style="font-family: Comic Sans MS; font-size: '40'; color: '#0000FF'">&nbsp;  </div>
<hr>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina4.

A continuación en la figura IV. 4 se muestra el ajuste de valores:



IV.4 Ajuste de Valores.

IV.7 Posicionar Imágen por Detrás del Texto.

Para posicionar una imágen detrás de un texto, HTML dinámico nos permite hacerlo. Por medio de la imágen que se emite al principio del documento simplemente la colocamos con la etiqueta ``. Para colocar el texto sobre la imágen utilizamos la etiqueta `<layer>`. Ejemplo:

`<layer top= "yy" left= "xx">`

Con los atributos `top` y `left` se definen las coordenadas del vértice superior izquierdo de la capa.

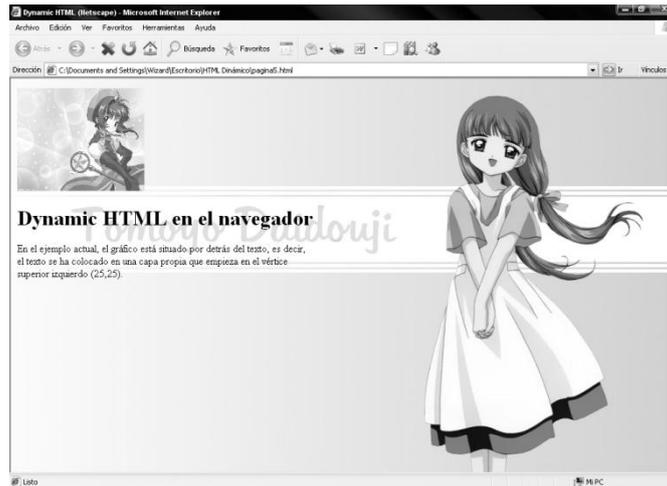
Ejemplo:

```
<html>
<head>
<title>Dynamic HTML </title>
</head>
<body background="/.1.jpg">

<layer top="25" left="25">
<h1>Dynamic HTML en el navegador </h1>
<p>En el ejemplo actual, el gráfico está situado por detrás del texto, es decir,
<br>
el texto se ha colocado en una capa propia que empieza en el vértice
<br>superior izquierdo (25,25).</p>
</layer></body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina5.

A continuación en la figura IV. 5 se muestra la imágen detrás del texto:



IV.5 Imagen Detrás del Texto,

IV.8 Definir una Imagen y un Texto de Fondo.

Además de colocar imágenes de fondo también podemos colocar texto de fondo a través de la etiqueta <layer>, podemos posicionar elementos de HTML libremente. Se puede posicionar en la etiqueta <layer> los atributos top y left, para posicionar la imagen.

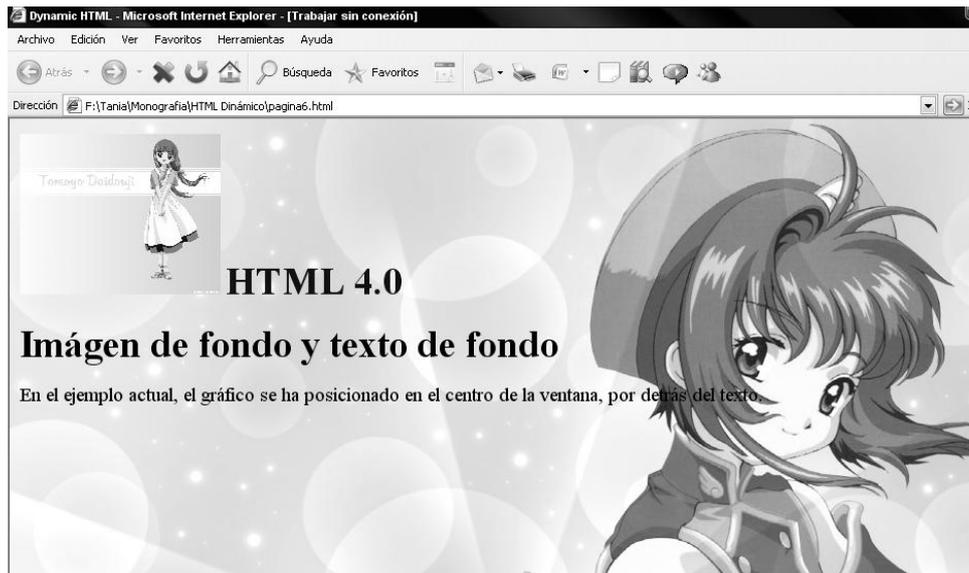
Ejemplo:

```
<html>
<head>
<title>Dynamic HTML</title>
</head>
<body bgcolor="#FFFFFF" background="/sakura.jpg">
<layer left="150" top="70">

</layer>
<layer left="150" top="120">
<font color="blue" size="6"><b>HTML 4.0</b></font>
</layer>
<layer top="50" left="25">
<h1>Imagen de fondo y texto de fondo</h1>
<p>En el ejemplo actual, el gráfico se ha posicionado
en el centro de la ventana, por detrás del texto.</p>
</layer>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina6.

A continuación en la figura IV. 6 se muestra texto e imagen de fondo:



IV.6 Texto e Imágen de Fondo

IV.9 Textos Sombreados.

Para crear textos con sombras se utiliza el filtro Shadow en el atributo style de la etiqueta <div>. Con los valores width y height se definen las dimensiones del área rectangular que ocupará el texto sombreado. Los parámetros que debemos transmitir son el color de sombra, los valores offset de x e y y el valor positivo=1, todos ellos en paréntesis y separados por comas.

Ejemplo:

```
<html>
<head>
<title>Dynamic HTML</title>
</head>
<body background="/sakura.jpg">
<h1>Prueba de filtro Shadow</h1>
<div style = "font-size: 26pt; font-family: Arial; color: silver">
Texto sin sombreado
<div style = "width: 420; height: 50; font-size: 36pt;
font-family: Arial; color: #FFFF00;
filter:dropShadow (color=#009000,
  offX=5, offY=5, positive=1)">
Texto con sombreado</div><br>
<div style=" font-family: 'Times New Roman'; font-size:36pt;
color: maroon; width: 420; height: 50;
filter:dropShadow (color=#009000,
  offX=10, offY=10, positive=-1)">
Texto con sombreado. <br>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina7.

A continuación en la figura IV. 7 se muestran textos sombreados:



IV.7 Textos Sombreados.

IV.10 Textos con Efectos de Brillo.

El filtro glow permite aplicar un efecto espectacular sobre el texto. Al borde exterior del texto se le aplica un color que produce el efecto de un borde brillante.

Para aplicar este filtro se requieren muy pocas líneas de código HTML. El atributo style se asigna en la etiqueta <div>. Esta etiqueta es la encargada de que los efectos se apliquen a todo el texto incluido dentro de la etiqueta.

El atributo id sólo se necesita dentro de la etiqueta cuando se quiera acceder al elemento correspondiente desde un script. A través de las propiedades width y height del atributo style podemos indicar en primer lugar, el área en la que se ha de producir el efecto. Su estructura es la siguiente:

filter:glow(color=#ff0000, strength=10)

El primer parámetro que espera el filtro es el de color para el efecto de brillo. En el segundo parámetro se indica la anchura de la línea para el borde brillante.

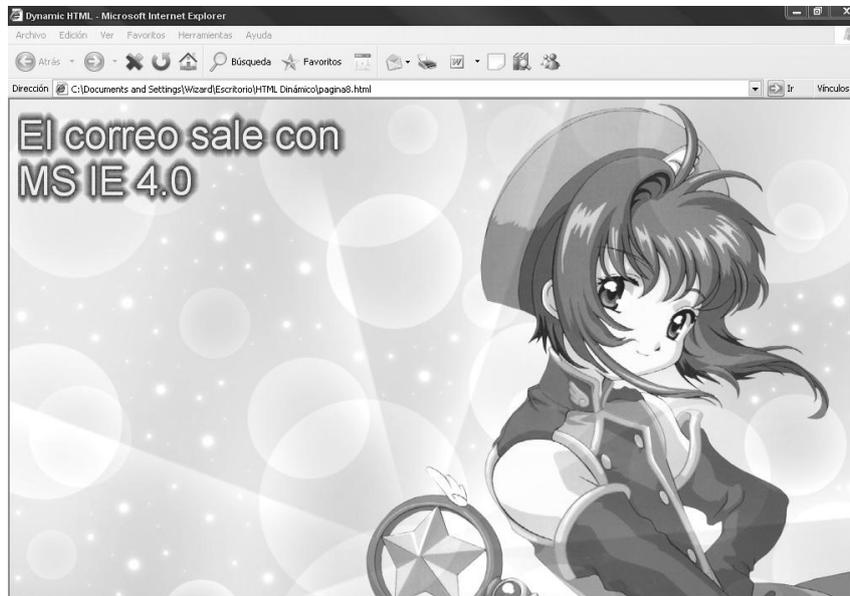
Ejemplo:

```
<html>
<head>
<title>Dynamic HTML</title>
</head>
<body background="/sakura.jpg">
<div id="Brillo"
style = "width: 420; height: 90; font-size: 36pt;
font-family: Arial; color: #FFFF00;
filter:glow (color=#ff0000, strength=10)"><p>
El correo sale con MS IE 4.0
</div>
```

```
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina8.

A continuación en la figura IV. 8 se muestran textos con efectos de brillo:



IV.8 Textos con Efectos de Brillo.

IV.11 Filtro con Efecto de Sombra.

Hay otro filtro que permite aplicar un efecto de sombreado sobre textos y todo tipo de objetos en general. Para conseguir este efecto se aplica el filtro shadow en la etiqueta <div>, sobre el área del documento que se desea.

Ejemplo:

```
<html>
<head>
<title>Dynamic HTML</title>
</head>
<body background= "./sakura.jpg">
<div id = "sText"
style = "width: 500; height: 40; font-size: 24pt;
font-family: Arial; color: #000090;
filter: shadow(color=#663399, direction=120)">
El hombre de las sombras...
</div>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina9.

A continuación en la figura IV. 9 se muestran los filtros con efectos de sombra:



IV.9 Filtros con Efectos de Sombra.

IV.12 Filtro con Efecto de Olas.

El efecto de olas en textos y objetos en general podemos conseguirlo a través del filtro wave y con muy poco esfuerzo. Todo lo que se debe hacer es definir un área para el texto con ayuda de las propiedades width y height del atributo style. A continuación se aplica el filtro wave en una etiqueta <div> sobre el área del documento. El filtro solo necesita como parámetros, un grosor de ola, una frecuencia, una frase y el parámetro add.

Ejemplo:

```
<html>
<head>
<title>Dynamic HTML</title>
</head>
<body bgcolor="#FFFFFF">
<div id = "Texto con olas"
style = "width: 400; height: 50;
font-size: 46pt; color: blue;
filter: wave(add=0, phase=4, freq=5, strength=2)">
Interferencia de imagen en Internet ...
</div>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta HTML Dinámico, pagina10.

A continuación en la figura IV. 10 se muestran los filtros con efectos de olas:



IV.10 Filtros con Efectos de Olas.

IV.13 Modificar el Estilo del Cursor.

Con la propiedad `cursor` se puede modificar fácilmente la forma del cursor del ratón. Todo lo que debemos hacer es crear una etiqueta para el objeto deseado y asignarle a la etiqueta el atributo `style` con la propiedad del cursor.

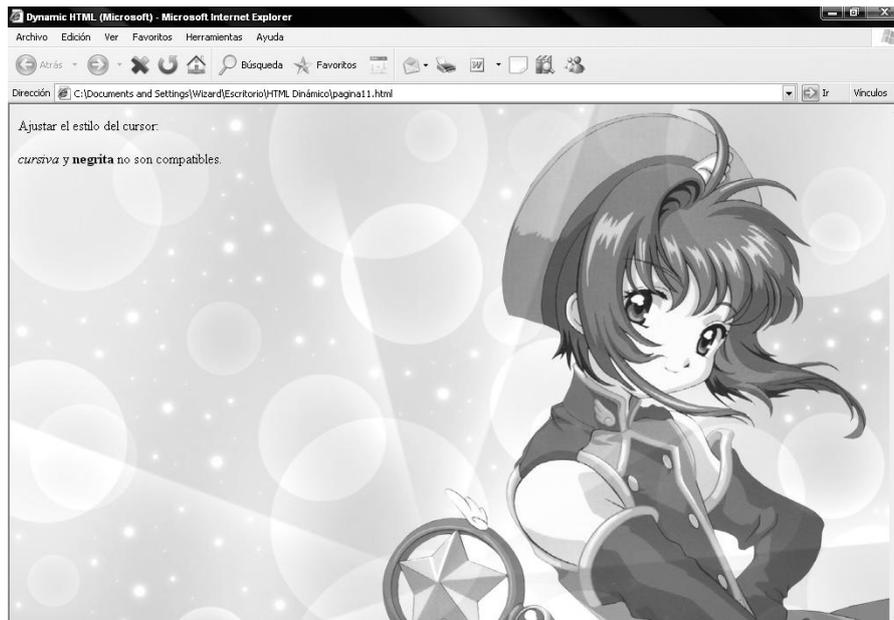
El código JavaScript y las instrucciones HTML se pueden ajustar igualmente en otros lugares.

Ejemplo:

```
<html>
<head>
<title>Dynamic HTML (Microsoft)</title>
</head>
<body bgcolor="#FFFFFF">
Ajustar el estilo del cursor:<p>
<i style="cursor:hand">cursiva</i> y <b style="cursor:move">negrita</b>
<span style="cursor:wait">no son</span> compatibles.
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de HTML Dinámico, pagina11.

A continuación en la figura IV. 11 se muestran la modificación del estilo del cursor:



IV.11 Modificación del Estilo del Cursor.



Capítulo V

PHP y MySQL

Capítulo V PHP y MySQL.

El lenguaje PHP es un lenguaje de programación de estilo clásico, es decir un lenguaje de programación con variables, sentencias, condicionales, ciclos y funciones. No es un lenguaje de marcado como lo puede ser HTML, XML ó WML. Está más cercano a ser Javascript o lenguaje C.

Los programas hechos en PHP son ejecutados en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML.

Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del browser, sin embargo para que las páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

V.1 Programación con PHP

En el código PHP se incluyen etiquetas especiales de comienzo y final que nos permite entrar y salir del modo o estructura de PHP.

La estructura de PHP a modo de HTML es la siguiente:

```
<html>
<head>
<title> </title>
</head>
<body>
<?php
    //Instrucciones de PHP
?>
</body>
</html>
```

O también la podemos escribir a modo de PHP, como se muestra a continuación:

```
<?php
    //Instrucciones de PHP
?>
```

Cabe señalar que cualquiera de las dos instrucciones es válida, solo hay que saber utilizarlas conforme las necesitemos, como lo explicaremos durante este capítulo.

Estas etiquetas especiales de comienzo y final nos llevan a la existencia de 4 modalidades diferentes de poder acceder al modo PHP, de las cuales solo trabajaremos con una, la cual se muestra continuación:

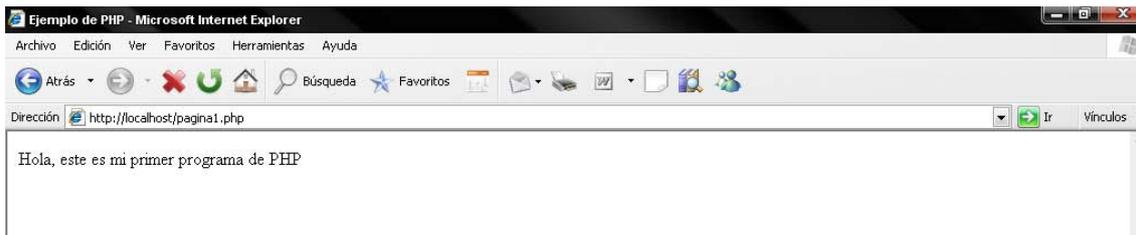
```
<html>
<head>
<title> Ejemplo de PHP </title>
</head>
<body>
<?php echo "Hola, este es mi primer programa de PHP";?>
</body>
```

</html>

Cabe mencionar que todo el código de estas páginas debe estar guardado en un archivo con la extensión “.php”, para poderlo ejecutar con la ayuda de un servidor de PHP.

La instrucción echo, es la que nos permite escribir dentro de la pantalla del navegador, todo lo que escribamos dentro de sus respectivas comillas, será visto.

A continuación en la figura V. 1 se muestra la programación en PHP:



V.1 Programación en PHP

Este ejercicio lo podemos ver en la carpeta de PHP, pagina1.

V.2 Separación de Instrucciones.

Las instrucciones se separan igual que en C ó Pascal terminando cada una de las sentencias con un punto y coma.

La etiqueta de cierre (?>) también implica el fin de una sentencia.

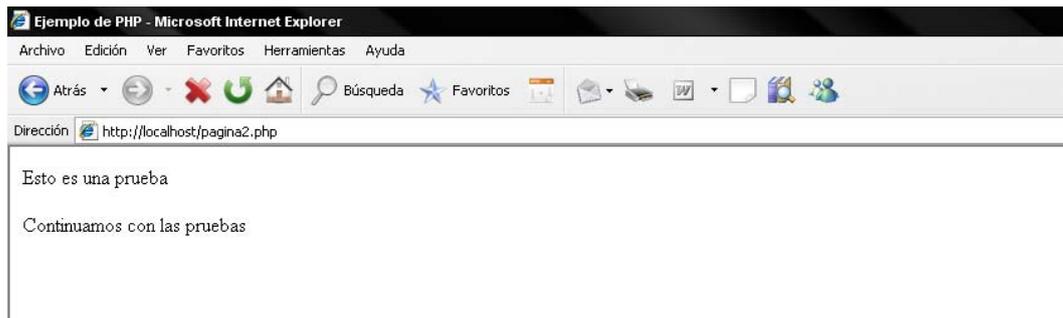
Ejemplo:

```
<html>
<head>
<title> Ejemplo de PHP </title>
</head>
<body>
<?php echo "Esto es una prueba";?>
<?php echo"<br> <br>Continuamos con las pruebas";?>
</body>
</html>
```

 **Nota:** Tenemos que señalar que podemos ocupar todas las etiquetas que conocemos de HTML, dentro de la instrucción **echo**, las cuales tendrán el mismo funcionamiento que en HTML.

Este ejercicio lo podemos ver en la carpeta de PHP, pagina2.

A continuación en la figura V. 2 se muestra la separación de instrucciones:



V.2 Separación de Instrucciones.

V.3 Comentarios de PHP

A diferencia de algunos lenguajes de programación, PHP no soporta comentarios tipo C, C++ ó Unix. Un ejemplo de esto es el siguiente:

```
<?php echo "Prueba de Comentarios"; //Comentario en lenguaje C++?>
```

Un comentario correcto en PHP es el siguiente:

```
<?php  
/*Esto es un comentario*/  
echo "Esto es una demostración para colocar comentarios";  
?>
```

V.4 Variables

Una variable es un elemento al cual se le asigna un nombre y atribulle un determinado tipo de información. Las variables son consideradas como una de las bases fundamentales de la programación.

En PHP las variables se representan con un signo de pesos, seguido por el nombre de la variable que se le dará. El nombre en las variables es sensible a minúsculas y mayúsculas. Ejemplo:

```
$var = "Carlos";  
$var2 = "Wendy";  
echo "$var, $var2";  
//Muestra en la Pantalla "Carlos,Wendy"
```

Las variables son nombres de variables que se pueden establecer y usar dinámicamente. Una variable normal se establece como una sentencia. Ejemplo:

```
$a= "Hola";
```

Una variable variable toma el valor de una variable y lo trata como el nombre de otra variable. Ejemplo:

```
$$a= "mundo"  
echo "$a${$a}";  
echo "$Hola$a";
```

Como resultado obtendremos "Hola Mundo".

Ejemplo:

```
<html>  
<head>  
<title> Ejemplo de PHP </title>  
</head>  
<body>  
<?php  
$variable=2;  
$variable1="Carlos";  
$variable2="Wendy";  
echo"<br>";  
echo"<br>";  
echo "<h1>$variable1, $variable2</h1>";  
echo"<br>";  
echo"<h1>";  
echo " Esta es una variable $variable";  
echo"</h1>";  
?>  
</body>  
</html>
```

A continuación en la figura V. 3 se muestra la utilización de variables:



V.3 Utilización de Variables.

Este ejercicio lo podemos ver en la carpeta de PHP, pagina3.

V.5 Tipos de Datos.

Dentro del lenguaje PHP, existen diferentes tipos de datos los cuales se mencionan a continuación:

Enteros:

Los enteros se pueden especificar usando las siguientes sintaxis:

`$a=1234;` #Número Decimal

`$a =-123;` #Número Negativo

`$a = 0123;` # Número Octal (equivale a 83 en decimal)

`$a = 0X12;` #Número Hexadecimal (equivale a 18 decimal)

Números Enteros Flotantes.

Los números en punto flotante (“double”) se pueden especificar utilizando cualquiera de las siguientes sintaxis:

`$a = 1.234;`

`$a=1.2e3;`

Cadenas.

El único operador de cadenas que existen dentro de PHP es la concatenación y se realiza por medio de un punto(.). Ejemplo:

`$a= “Hola”;`

`$b= $a. “Mundo”;` //Ahora \$b contiene “Hola Mundo”.

Debemos hacer una distinción en la interpretación que hace PHP, con las comillas simples y las comillas dobles. Para que PHP interprete el contenido de la cadena. Ejemplo:

`$a= “Mundo”;`

echo = „Hola\$a”; // De esta manera el navegador escribirá “Hola\$a”

echo = “Hola\$a”; // De esta manera se escribirá en el navegador “Hola Mundo”

V.6 Caracteres Protegidos.

Dentro de PHP, tenemos algunas limitaciones a la hora de utilizar algunos caracteres, debido, a que el lenguaje PHP, los ocupa como sentencias o caracteres especiales, los cuales tienen alguna función en este lenguaje, para poder utilizarlos dentro de PHP, y poder visualizarlos sin ningún problema en el navegador utilizamos el carácter barra invertida (“\”), el cual nos permitirá utilizar estos caracteres especiales.

Algunos de ellos son:

Carácter	Significado
\n	Nueva Línea
\r	Retorno de Carro
\t	Tabulación Horizontal
\\	Barra Invertida
\\$	Signo de Dólar
\ “	Comillas Dobles
\ [0-7]{1-3}	La secuencia de caracteres que coincidan con la expresión regular es un carácter en notación octal.
\x[0-9 ^a -Fa]	La secuencia de caracteres que coincida con la expresión regular es f1[1,2] un carácter notación hexadecimal.

V.7 Operadores de Comparación.

En las instrucciones que se suelen utilizar con **if**, siempre se utilizan operadores comparativos. Estos operadores se encargan de comparar dos valores, devolviendo un valor lógico como true o false.

PHP soporta los siguientes operadores de comparación:

Sintaxis	Significado
\$ a > \$b	\$a mayor que \$b
\$a <= \$b	\$a menor o igual que \$b

<code>\$a >= \$b</code>	<i>\$a mayor o igual que \$b</i>
<code>\$a == \$b</code>	<i>\$a igual que \$b</i>
<code>\$a != \$b</code>	<i>\$a distinto de \$b</i>
<code>\$a < \$b</code>	<i>\$a menor que \$b</i>

V.8 Operadores Lógicos.

Para las evaluaciones más complejas requerimos de operadores lógicos. PHP dispone de los siguientes operadores lógicos:

Operador	Significado
<code>\$a && \$b</code>	Verdadero si ambos son verdaderos
<code>\$a OR \$b</code>	Verdadero si alguno de los dos es verdadero.
<code>\$a !! \$b</code>	Verdadero si alguno de los dos es verdadero.
<code>\$a XOR \$b</code>	Verdadero solo si uno de los dos es verdadero
<code>!\$a</code>	Verdadero si \$a es falso.
<code>\$a AND \$b</code>	Verdadero si ambos son verdaderos

V.9 Operadores de Asignación.

Para asignar a una variable algún valor podemos utilizar los operadores de asignación y efectuar todas las operaciones que podemos realizar al igual que en el lenguaje C. Algunos de ellos son:

Operador	Significado
<code>\$a = \$b</code>	<i>Asigna a \$a el contenido de \$b</i>
<code>\$a += \$b</code>	Suma a \$b a \$a
<code>\$a -= \$b</code>	Resta a \$b a \$a
<code>\$a *= \$b</code>	Multiplica \$a por \$b y lo asigna a \$a
<code>\$a /= \$b</code>	Divide \$a por \$b y lo asigna a \$a

\$a. = \$b

Añade la cadena \$b con la cadena \$a

Ejemplo:

```
<html>
<head>
<title> Ejemplo de PHP </title>
</head>
<body>
<?php
$salario=3500;
$impuesto=20; //porcentaje
$sueldo=$salario-(($salario/100)*$impuesto);
echo "<h1>Sueldo del trabajador sin impuesto: $salario <br>";
echo "Con impuesto: $sueldo</h1>";
?>
</body>
</html>
```

A continuación en la figura V. 4 se muestran los operadores de asignación:

Este ejercicio lo podemos ver en la carpeta de PHP, pagina4.



V.4 Operadores de Asignación.

V.10 Expresiones de Condición.

A veces es necesario colocar una asignación que defina un valor dependiendo de una condición. Estas asignaciones las podemos realizar con ayuda de las expresiones condicionales. Estas expresiones de condición disponen de dos valores y una expresión comparativa. Ejemplo:

(Condición) valor1:valor2

Si la condición se cumple, a la expresión se la asigna el valor de valor1. En otro caso toma el valor de valor2.

Estas expresiones condicionales las podemos utilizar como ramificaciones, en el momento en que se requiera. Ejemplo:

Entrada=(Edad>=18)

V.11 Estructura IF

Al igual que en el Lenguaje C, tenemos la estructura if. La cual permite ejecutar un bloque de instrucciones si la condición es verdadera y otro bloque de instrucciones si esta es falsa.

La condición que evaluemos ha de estar encerrada entre paréntesis (esto es aplicable a todas). Su estructura es:

```
if (condición)
{
    /* Este bloque se ejecuta si la condición es verdadera*/
}
else
{
    /*Este bloque se ejecuta si la condición es falsa*/
}
```

Podemos utilizar el if sin la necesidad de utilizar else se usa, la siguiente estructura:

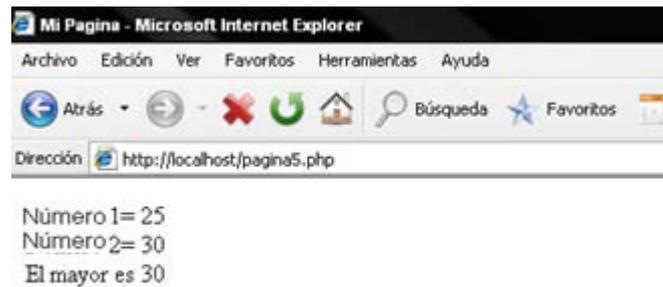
```
if( $a > 4)
echo “$a es mayor que 4”;
```

Ejemplo:

```
<HTML>
<HEAD>
<TITLE> Mi Página </TITLE>
</HEAD>
<BODY>
<?php
$numero1=25; $numero2=30;
$mayor=0;
if ($numero1 > $numero2)
$mayor=$numero1;
else
$mayor=$numero2;
echo "Número 1= $numero1 <BR>";
echo "Número 2= $numero2 <BR>";
echo "El mayor es $mayor";
?>
</BODY>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina5.

A continuación en la figura V. 5 se muestran la estructura if:



V.5 Estructura IF

V.12 Condición IF...ELSEIF...ELSE

Esta sentencia permite ejecutar varias condiciones if en cascada.

Ejemplo:

```
<?php
if ($nombre == "")
{
    echo "Tú no tienes nombre";
} elseif (($nombre=="Tania") OR ($nombre=="tania")) {
    echo "Tu nombre es Tania";
} else
{
    echo "Tu nombre es " . $nombre;
}
?>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina7.

A continuación en la figura V. 6 se muestran la estructura if elseif:



V.6 Estructura IF ELSEIF.

V.13 Sentencia Switch

La sentencia switch evalúa y compara cada expresión de la sentencia case, con la que si llega al final de la lista del case y no se encuentra una condición verdadera ejecuta el bloque default. Si encontramos una condición verdadera colocamos un

break, al termino del case, para que la sentencia switch, no siga buscando en las demás listas case.

Ejemplo:

```
<?php
switch ($dia) {
case "Lunes":
echo "Hoy es Lunes";
break;
case "Martes":
echo "Hoy es Martes";
break;
case "Miercoles":
echo "Hoy es Miercoles";
break;
case "Jueves":
echo "Hoy es Jueves";
break;
case "Viernes":
echo "Hoy es Viernes";
break;
case "Sábado":
echo "Hoy es Sábado";
break;
case "Domingo":
echo "Hoy es Domingo";
break;
default:
echo "Esa cadena no corresponde
a ningún día de la semana";
}
?>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina8.

A continuación en la figura V. 7 se muestra la sentencia switch:



V.7 Sentencia Switch.

V.14 Sentencia While

La sentencia while ejecuta un bloque de código mientras se cumpla una determinada condición.

Ejemplo:

```
<?php
$num = 1;
while ($num < 5)
{
echo $num;
$num++;
}
?>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina9.

A continuación en la figura V. .8 se muestra la sentencia while:



V.8 Sentencia While.

V.15 Sentencia Do While

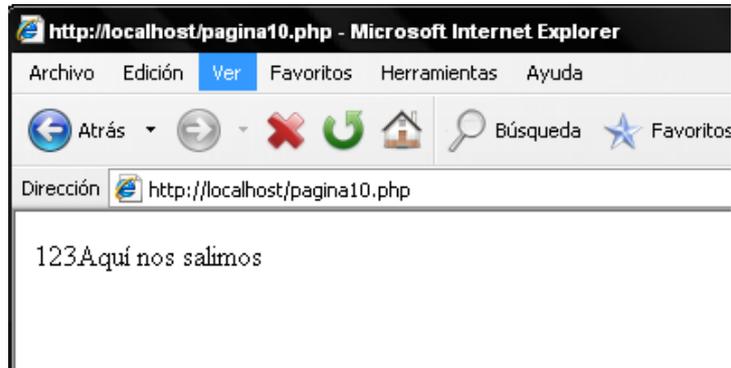
Esta sentencia es similar a la sentencia while, salvo que con esta sentencia primero ejecutamos el bloque de código y después se evalúa la condición, por lo que el código se ejecuta al menos una vez.

Ejemplo:

```
<?php
$num = 1;
do {
echo $num;
if ($num == 3){
echo "Aquí nos salimos \n";
break;
}
$num++;
} while ($num < 5);
?>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina10.

A continuación en la figura V.9 se muestra la sentencia do while:



V.9 Sentencia Do While.

V.16 Ciclo For.

El ciclo for no es estrictamente necesario, cualquier ciclo for puede ser sustituido fácilmente por otro while. Sin embargo el ciclo for resulta muy útil cuando ejecutamos un bloque de código a una condición de una variable para que encuentre el valor mínimo y máximo.

Ejemplo:

```
<?php
for ($num = 1; $num <=5;
$num++){
echo $num;
if ($num == 3){
echo "Aquí nos salimos \n";
break;
}
}
?>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina11.

A continuación en la figura V.10 se muestra el ciclo for:



V.10 Ciclo For.

V.17 Funciones

El uso de funciones nos da la capacidad de agrupar varias instrucciones bajo un solo nombre y poder llamarlas tantas veces las necesitemos, ahorrándonos la necesidad de escribirlas de nuevo.

Su estructura básica es:

```
<?php
function Nombre(parametro1, parametro2...)
{
instrucción1;
instrucción2;
instrucción3;
instrucción4;
return valor_de_retorno;
}
?>
```

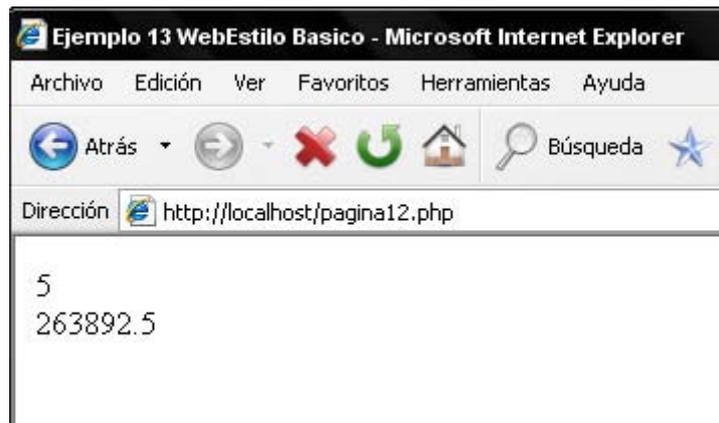
Opcionalmente podemos pasarle parámetros a las funciones las cuales se tratan como variables locales y así mismo podemos devolver este resultado con la instrucción return. Esto produce la terminación de la función retornando un valor.

Ejemplo:

```
<html>
<head>
<title>          </title>
</head>
<body>
<?php
function media_aritmetica($a,$b)
{
$media=($a+$b)/2;
return $media;
}
echo media_aritmetica(4,6),"<br>";
echo media_aritmetica(3242,524543),"<br>";
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, pagina12.

A continuación en la figura V.11 se muestra el uso de funciones:



V.11 Uso de Funciones.

V.18 Formularios desde PHP

Los formularios no forman parte de PHP, sino del lenguaje estándar de Internet HTML.

Utilizando PHP, podemos acceder fácilmente a los datos introducidos desde un formulario de HTML

Hay que mencionar que utilizando PHP dentro de los formularios se recomienda utilizar un archivo para todo el código HTML y otro específico para el código de PHP, como se muestra a continuación:

Ejemplo:

✓ Archivo formulario.html

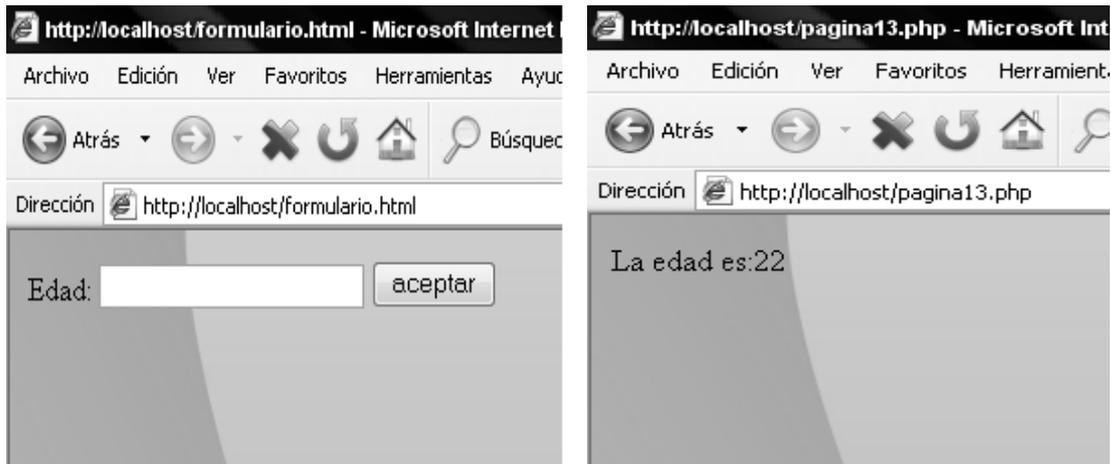
```
<HTML>
<BODY>
<FORM ACTION="pagina13.php" METHOD="POST">
Edad: <INPUT TYPE="text" NAME="edad">
<INPUT TYPE="submit" VALUE="aceptar">
</FORM>
</BODY>
</HTML>
```

✓ Archivo pagina13.php

```
<HTML>
<BODY background="3.jpg">
<?PHP
$Edad=$_POST['edad'];
echo "La edad es:". $Edad;
?>
</BODY>
</HTML>
```

Este ejercicio lo podemos ver en la carpeta de PHP, formulario y pagina13.

A continuación en la figura V.12 y V.13 se muestra el uso de formularios y PHP:



V.12 y V.13 Uso de Formularios y PHP.

Un formulario nos permite tener acceso a los diferentes tipos de elementos de entrada del formulario, cabe mencionar que estas herramientas se vieron a detalle en el capítulo I:

Algunos de los elementos que podemos utilizar para introducir datos son:

- Ⓞ TEXT
- Ⓞ RADIO
- Ⓞ CHECKBOX
- Ⓞ BUTTON
- Ⓞ FILE
- Ⓞ HIDDEN
- Ⓞ PASSWORD
- Ⓞ SUBMIT
 - Elemento SELECT
- Ⓞ Simple / múltiple
 - Elemento TEXTAREA

A continuación mostraremos algunos ejemplos utilizando estos elementos:

TEXT

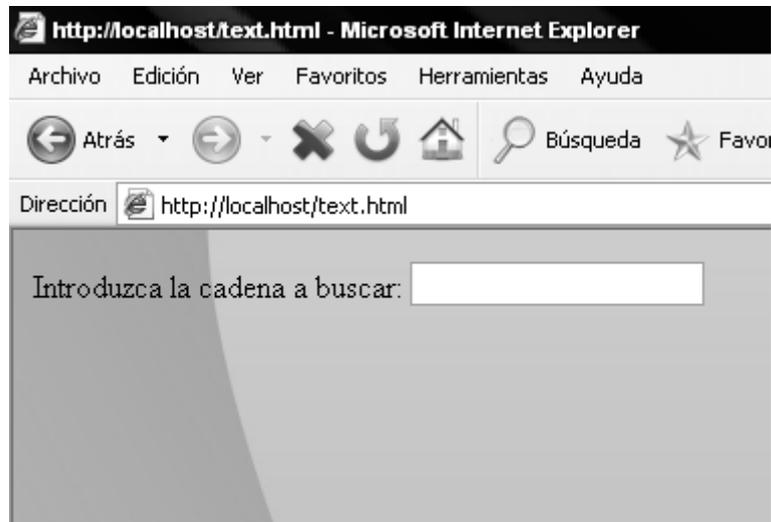
Ejemplo text.php:

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
Introduzca la cadena a buscar:
<INPUT TYPE="text" NAME="cadena" VALUE="valor por defecto" SIZE="20">
<?PHP
$cadena = $_REQUEST[„cadena”];
print ($cadena);
?>
```

```
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, text.php.

A continuación en la figura V.14 se muestra el uso de caja de texto:



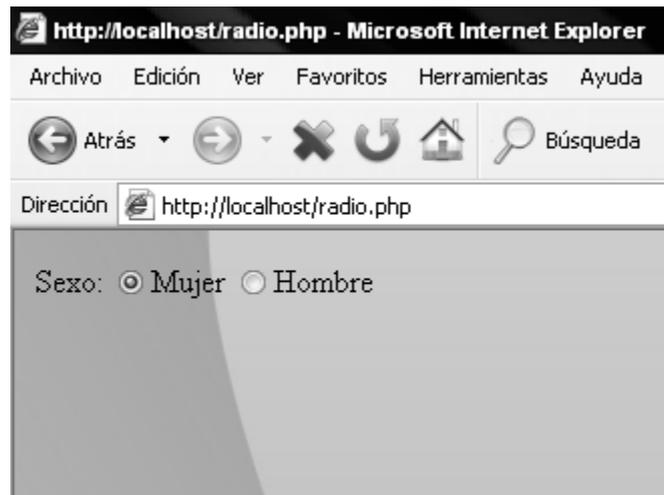
V.14 Uso de Caja de Texto

RADIO

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
Sexo:
<INPUT TYPE="radio" NAME="sexo" VALUE="M" CHECKED>Mujer
<INPUT TYPE="radio" NAME="sexo" VALUE="H">Hombre
<?PHP
$sexo = $_REQUEST[,'sexo'];
print ($sexo);
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, radio.php.

A continuación en la figura V.15 se muestra el uso de las casillas de radio:



V.15 Uso de Casillas de Radio.

CHECKBOX

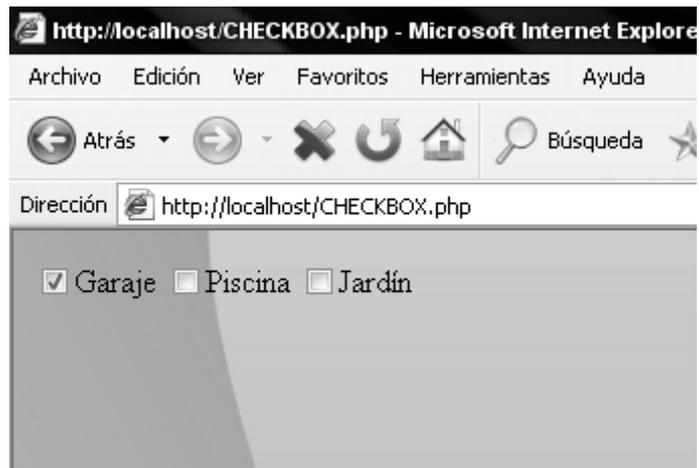
```

<HTML>
<HEAD>
<TITLE>          </TITLE>
</HEAD>
<BODY BACKGROUND="/.3.JPG">
<INPUT TYPE="checkbox" NAME="extras[]" VALUE="garaje" CHECKED>Garaje
<INPUT TYPE="checkbox" NAME="extras[]" VALUE="piscina">Piscina
<INPUT TYPE="checkbox" NAME="extras[]" VALUE="jardin">Jardín
<?PHP
$extras = $_REQUEST[„extras“];
$n = count ($extras);
for ($i=0; $i<$n; $i++)
ECHO "$extras[$i]<BR>";
?>
</BODY>
</HTML>

```

Este ejercicio lo podemos ver en la carpeta de PHP, CHECKBOX.php.

A continuación en la figura V.16 se muestra el uso de las casillas checkbox:



V.16 Uso de Casillas Checkbox.

BUTTON

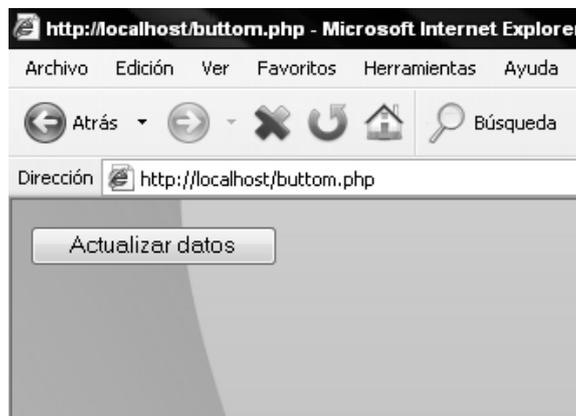
```

<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<INPUT TYPE="button" NAME="actualizar" VALUE="Actualizar datos">
<?PHP
$actualizar = $_REQUEST[,'actualizar'];
if ($actualizar)
print ("Se han actualizado los datos");
?>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de PHP, button.php.

A continuación en la figura V.17 se muestra el uso de botones:



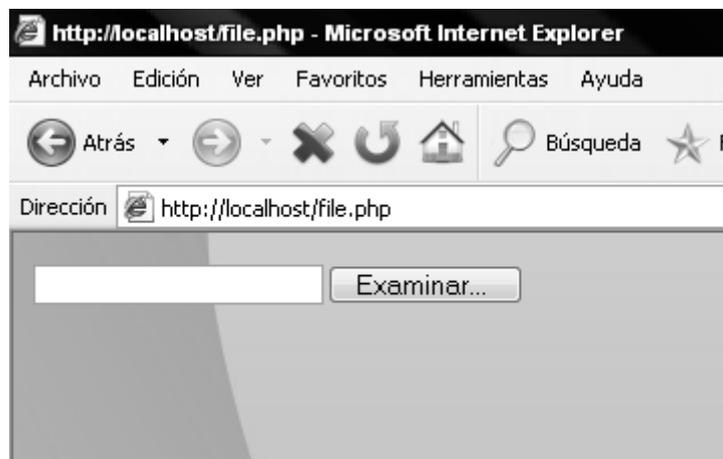
V.17 Uso de Botones.

File

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<FORM ACTION="procesa.php" METHOD="post" ENCTYPE="multipart/form-data">
<INPUT TYPE="file" NAME="fichero">
</FORM>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, file.php.

A continuación en la figura V.18 se muestra como localizar un archivo:



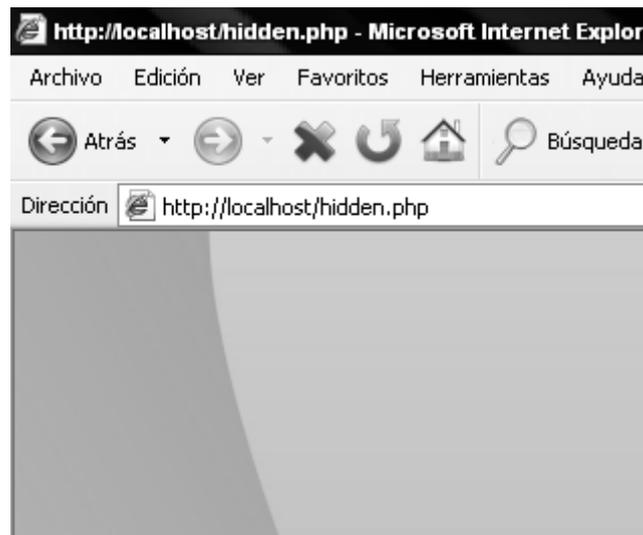
V.18 Localizar un Archivo.

Hidden

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<?PHP
echo "<INPUT TYPE='hidden' NAME='username' VALUE='$usuario'>";
?>
<?PHP
$username = $_REQUEST[„username“];
print ($username);
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, hidden.php.

A continuación en la figura V.19 se muestran los controles ocultos:



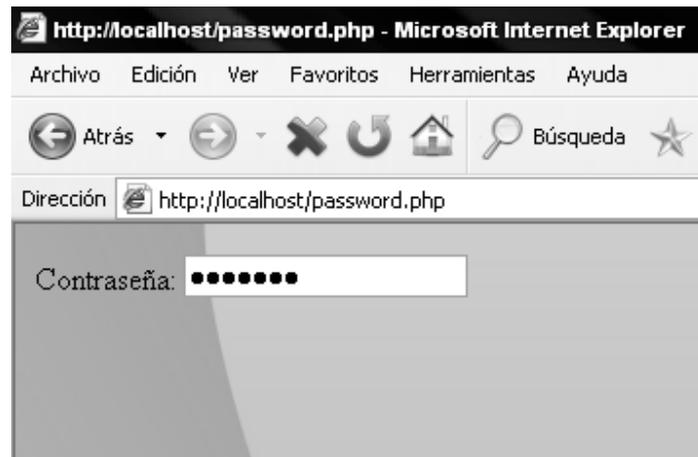
V.19 Controles Ocultos.

Password

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<form>
Contraseña: <INPUT TYPE="password" NAME="clave">
</form>
<?PHP
$clave = $_REQUEST[,'dave'];
print ($clave);
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, password.php.

A continuación en la figura V.20 se muestra la introducción de password:



V.20 Introducción de Password.

Submit

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<form>
<INPUT TYPE="submit" NAME="enviar" VALUE="Enviar datos">
</form>
<?PHP
$enviar = $_REQUEST[„enviar”];
if ($enviar)
print ("Se ha pulsado el botón de enviar");
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, submit.php.

Select

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<form>
Color:
<SELECT NAME="color">
<OPTION VALUE="rojo" SELECTED>Rojo
<OPTION VALUE="verde">Verde
<OPTION VALUE="azul">Azul
</SELECT>
</form>
```

```

<?PHP
$color = $_REQUEST[„color”];
print ($color);
?>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de PHP, select.php.

A continuación en la figura V.21 se muestra la casilla de selección:



V.21 Casilla de Selección.

Select Múltiple

```

<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<form>
<SELECT MULTIPLE SIZE="3" NAME="idiomas[]">
<OPTION VALUE="ingles" SELECTED>Inglés
<OPTION VALUE="frances">Francés
<OPTION VALUE="aleman">Alemán
<OPTION VALUE="Español">Español
</SELECT>
</form>
<?PHP
$idiomas = $_REQUEST[„idiomas”];
$n = count ($idiomas);
for ($i=0; $i<$n; $i++)
echo "$idiomas[$i]<BR>";
?>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de PHP, select_multiple.php.

A continuación en la figura V.22 se muestra la casilla de multiple selección:



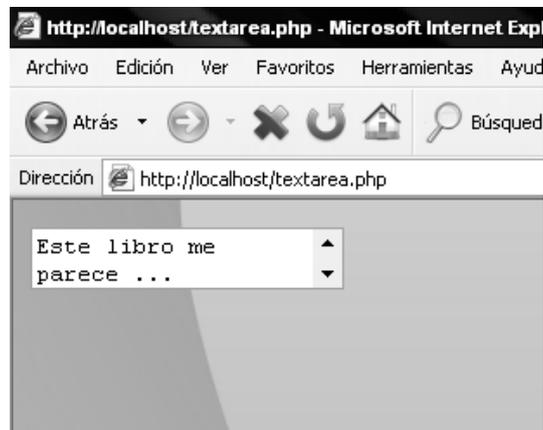
V.22 Casilla de Multiple Selección.

TEXTAREA

```
<html>
<head>
<title> </title>
</head>
<body background="/.3.jpg">
<form>
<TEXTAREA COLS="50" ROWS="4" NAME="comentario">
Este libro me parece ...
</TEXTAREA>
</form>
<?PHP
$comentario = $_REQUEST[,"comentario"];
print ($comentario);
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, textarea.php.

A continuación en la figura V.23 se muestra la caja de texto multilinea:



V.23 Caja de Texto Multilinea.

V.19 MYSQL

Para trabajar con PHP con una base de datos necesitamos definir primero lo que es una base de datos.

- ✚ Base de Datos: Es un conjunto de datos estructurados.
- ✚ Base de Datos Relacional: Almacena los datos en tablas separadas, colocando todos los datos dentro de un almacén.

Para agregar, acceder y procesar los datos almacenados en una base de datos se necesita un sistema administrador de base de datos como MYSQL.

SQL forma parte de MYSQL conocido como Lenguaje de Consultas Estructurado, este es el lenguaje estandarizado más usado para acceder a las bases de datos.

✓ Creación de una Base de Datos.

Para comenzar a crear una base de datos se necesita utilizar el comando **CREATE DATABASE**, seguido del nombre que contendrá la tabla y para terminar la sentencia se utiliza un punto y coma:

Ejemplo:

```
create database mercancia;
```

Con esta sentencia se crea la base datos, en la cual se albergará toda la información.

✓ Selección de la Base de Datos a Utilizar.

Por medio de esta instrucción podemos seleccionar la base de datos, que utilizaremos en un momento específico utilizando el comando **USE**, seguido del nombre de la base de datos a utilizar, cerrando con un punto y coma.

Ejemplo:

```
use mercancia;
```

✓ Tipos de Datos.

Dentro de una base de datos se puede colocar datos de diferente forma como:

Ⓜ Números

- ✓ **Int (Enteros)**
- ✓ **Float (Flotantes)**
- ✓ **Numeric (Numéricos)**

@ Caracteres

- ✓ **char (Cadenas)**
- ✓ **varchar (Cadenas)**

Nota:

- ❖ **CHAR:** Rellena con espacios en blanco, si no utilizamos todo el espacio.
- ❖ **VARCHAR:** Solo ocupa el espacio que necesita, aunque tenga para más almacenamiento.

@ Fechas

- ✓ **DATE TIME**
- ✓ **DATE**

✓ Creación de una Tabla de Datos con Atributos

Cuando ya estamos seguros de saber cuáles son los atributos que contendrá la tabla podemos incluirlos mediante el comando **CREATE TABLE**, seguido del nombre de la tabla, los podemos incluir desde que comenzamos a crearla, separando cada uno de los atributos con una coma (.). Ejemplo:

```
create table alumno(  
id_alumno numeric(2,0)  
primary key not null,  
nombre char(20),  
paterno varchar(20),  
materno varchar(20),  
edad date);
```

Donde:

- **Primary Key**, es una llave primaria, la cual explicaremos más adelante
- ✓ **Agregar Atributos o Agregar Columnas.**

EL agregar atributos y columnas suele ser muy utilizado, en algunas bases de datos, las cuales ya contenga sus propias bases de datos, en las que solo se les desee agregar un dato o columna más a lo existente, para evitar, tener que volver hacer la tabla se tiene el comando **ALTER TABLE**, con el cual podemos desde agregar una columna a la tabla o solamente un atributo. **Ejemplo:**

✓ **Agregar Atributo**

```
alter table alumno  
add comentarios varchar(50);
```

✓ **Agregar Columna.**

```
alter table alumno  
add column fecha_nac char(50);
```

 **Borrar una columna de la Tabla.**

Cuando deseamos borrar alguna de las columnas que contiene la tabla, lo podemos hacer mediante el comando **DROP** seguido del nombre de la columna, de la siguiente manera:

```
alter table alumno  
drop fecha_nac;
```

Por medio de esta sentencia estamos indicando con el comando **ALTER**, que seleccionaremos la tabla alumno, después con el comando **DROP**, que borraremos la columna fecha_nac.

 **Borrar una Tabla**

Por medio del comando **DROP** seguido de la palabra **TABLE** y el nombre de la tabla, podemos indicarle que deseamos borrar toda la tabla, con todo lo que tenga de información de la siguiente manera:

```
drop table curso2;
```

 **Formas de Generar Llaves Primarias.**

Una llave primaria es una marca con la cual podemos indicar a la base de datos, que este será nuestro identificador único, a la hora de buscar datos dentro de una tabla., simbolizado como **PK**.

✓ **Creación de la llave primaria.**

```
create table alumno(  
id_alumno numeric(2,0)  
primary key not null,  
nombre char(20),  
paterno varchar(20),
```

materno varchar(20),
edad date);

Por medio de las palabras **PRIMARY KEY**, le indicamos a la base de datos, que esa columna será la llave primaria.

Insertar Datos en la Tabla

Cuando estamos listos, para comenzar a insertar datos dentro de la tabla, lo podemos hacer mediante el comando **INSERT TO**, seguido del nombre de la tabla, e indicándole, las columnas a las cuales insertaremos estos datos. Existen dos formas de insertar datos los cuales son inserción explícita e inserción implícita.

✓ Inserción Explícita.

Con este tipo de inserción tenemos que indicarle a la tabla las columnas a las cuales insertaremos datos, debemos tener mucho cuidado a la hora de insertar los datos, ya que estos se insertaran de la forma en que indicamos dentro del **INSERT TO**, una vez definidas las columnas, por medio del atributo **VALUE**, indicamos los datos a insertar, es aquí donde es importante la forma de coloca los datos, que se si nosotros en materno, colocamos los datos de edad, nos marcará un error a la hora de insertar el dato.

Ejemplo:

```
INSERT INTO alumno (id_alumno, nombre, paterno, materno, edad)  
VALUES (1,'Tania','Garcia','Sanchez',22 );
```

✓ Inserción Implícita.

Con este tipo de inserción solo colocamos los datos a insertar sin indicar las columnas, ya que por medio de esta inserción estamos seguros de conocer la estructura de la base de datos.

Ejemplo:

```
INSERT INTO alumno  
values (2,'Alejandro','Martinez','Rosales',25 );
```

Mostrar los Datos Insertados en una Tabla.

Si queremos ver los datos que tenemos insertados en la tabla lo podemos hacer por medio del comando **SELECT**, en donde se puede indicar que datos queremos ver en específico, o simplemente muestre toda la tabla.

✓ **Mostrar toda la Tabla**

```
select *  
  
from alumno;
```

Por medio de esta sentencia se muestra toda la tabla.

	id_alumno	nombre	paterno	materno	edad
1	1	Tania	Garcia	Sanchez	22
2	2	Alejandro	Martinez	Rosales	25

Donde el * nos dice que mostremos toda la tabla, con el atributo from, indicamos el nombre de la tabla a mostrar.

✓ **Mostrar algunos atributos de la tabla.**

Además de mostrarnos toda la tabla el atributo select, también podemos indicarle qué columnas queremos ver solamente, esta indicación se coloca dentro del select, indicándole las columnas y el nombre de la tabla a mostrar.

Ejemplo:

```
select id_alumno,nombre,edad  
  
from alumno;
```

La vista que nos muestra el programa es la siguiente:

	id_alumno	nombre	edad
1	1	Tania	22
2	2	Alejandro	25

✚ **Borrar un Registro**

Si deseamos borrar un registro el cual no necesitamos o introducimos erróneamente, lo podemos hacer mediante el comando DELETE, borramos toda la información que contenga este registro.

Ejemplo:

```
delete from alumno  
  
where id_alumno=3;
```

Donde **from** nos indica la tabla a la cual se le borrara el registro, y **where** nos muestra la condición que debe tener el registro, en este caso nos indica que se borrará el registro donde el id_alumno sea igual a 3.

V.20 PHP Y MYSQL

Para poder trabajar con una base de datos dentro de una página WEB. Php puede interactuar con cualquier gestor de base de datos. Los más populares son: SQL, ORACLE, POSTGRESQL, MYSQL. Las más usadas por php es mysql y postgresql

Cada uno de estos gestores tienen formas diferentes de conectarse con php

Sentencias Básicas de Conexión.

✓ **mysql_connect**

mysql_connect("servidor:puerto","usuario","contraseña");

Devuelve: un identificador de conexión, o FALSE en caso de error.

mysql_connect establece una conexión a un servidor de MySQL. Todos los argumentos son optativos, y si no se especifican, los valores por defecto son (' el servidor', nombre del usuario del usuario que posee el proceso del servidor, la contraseña). La cadena servidor también puede incluir un número del puerto, "servidor:puerto".

En caso de realizar una segunda llamada a mysql_connect con los mismos argumentos, no se establecerá ninguna nueva conexión, sino se devolverá el identificador de conexión de la ya existente.

La conexión al servidor se cerrará en cuanto la ejecución del script acabe, a menos que la cerremos antes con la función mysql_close.

✓ **mysql_close**

int mysql_close(int [link_identifier]);

Devuelve: TRUE si se ha cerrado correctamente, FALSE en caso de error.

mysql_close es la conexión a la base de datos MySQL asociada al identificador de conexión especificado. Si no se especifica un identificador de conexión, se asume la conexión abierta.

Esta función no es normalmente necesaria en conexiones no-persistentes (abiertas con mysql_connect) ya que ésta se cerrará automáticamente al final de la ejecución del script o página. La función mysql_close no cierra una conexión persistente.

✓ **mysql_db_query**

mysql_db_query("bd","query_consulta",identificador);

Devuelve: un identificador de conexión, o FALSE en caso de error.

Ejecuta una consulta en una base de datos. Si el identificador no se especifica, la función intenta encontrar una conexión abierta con el servidor. Si no encuentra una conexión, intentará crear una (similar a `mysql_connect()` sin argumentos).

✓ **mysql_free_result**

mysql_free_result(conexión);

Sólo debe ser utilizada si la cantidad de memoria utilizada para almacenar el resultado de una consulta es muy grande. Cuando se ejecuta esta función, toda la memoria asociada al resultado se libera.

✓ **mysql_query**

mysql_query(consulta);

Ejecuta una consulta a la base de datos activa en el servidor asociado al identificador de conexión. Si no se especifica, se utiliza la última conexión abierta. Si no hay conexiones abiertas la función intenta establecer una.

Esta función devuelve **TRUE** o **FALSE** para indicar si las operaciones **UPDATE**, **INSERT** o **DELETE** han tenido éxito. Para la operación **SELECT** devuelve un nuevo identificador de resultado.

✓ **mysql_select_db**

mysql_select_db(sentencia, identificador);

Establece la base de datos activa en el servidor. Si no se especifica identificador de conexión se utiliza la última conexión abierta. Si no hay conexiones activas, la función intenta establecer una. A partir de la llamada a `mysql_select_db` las llamadas a `mysql_query()` actúan sobre la nueva base de datos activa.

✓ **mysql_fetch_array**

mysql_fetch_array (consulta_conexión);

Devuelve un array con la información correspondiente al resultado de una consulta especificado por su identificador o 'false' si ya no hay más filas.

Además de almacenar los datos a través de índices numéricos del array, también lo hace a través de índices asociativos, utilizando los nombres de los campos como claves.

Si dos o más columnas del resultado tienen el mismo nombre de campo, la última es la que tiene preferencia.

V.21 Agrupando PHP y MYSQL

Ahora que ya conocemos algunas de las sentencias que tiene PHP y MYSQL, podemos llegar a crear un programa el cual pueda registrar productos dentro de la base de datos, por medio de la página web, ya sea que estemos en otro lado de la ciudad o dentro de la misma empresa que contenga la base de datos. El ejemplo que se muestra a continuación, sirve para poder visualizar cómo debemos colocar cada una de las sentencias de conexión entre la base de datos y la página web. Además de ver la forma de colocar las sentencias para que la página Web, y la base de datos se sincronice.

✓ Index.php

```
<html>
<head>
<title>Tienda de Productos</title>
</head>
<body background="./zorro.bmp">
  <center>
    <h1><font color="white">Bodega Rozen Maiden</h1></font>
    <br><br>
  </center>
  <center>
    <font color="white"><h1>
    <ul>
    <li>Catalogo de Productos</li>
    </ul></h1>
    </font>
    <hr color="white">
    <?php
    $con=mysql_connect("localhost","root","cursophp");
    $sen=mysql_db_query("producto","select * from productos");
    while($col=mysql_fetch_array($sen))
    {
    echo"<Table>";
    echo"<TABLE BORDER=0>";
    echo"<TH COLSPAN=3>";
    echo"<TR>";
    echo "<td><form action='actualizar.php' method='POST'></td>";
    echo "<td><input type='hidden' value=".". $col[Clave]." name='Clave'></td>";
    echo "<td><input type='text' value=".". $col[Producto]."
    name='Descripcion'></td>";
    echo "<td><input type='submit' value='actualizar'></td>";
    echo "<td></form></td>";
    echo "<td><form action='eliminar.php' method='POST'></td>";
    echo "<td><input type='hidden' value=".". $col[Clave]." name='Clave'></td>";
    echo "<td><input type='hidden' value=".". $col[Producto]."
    name='Descripcion'></td>";
```

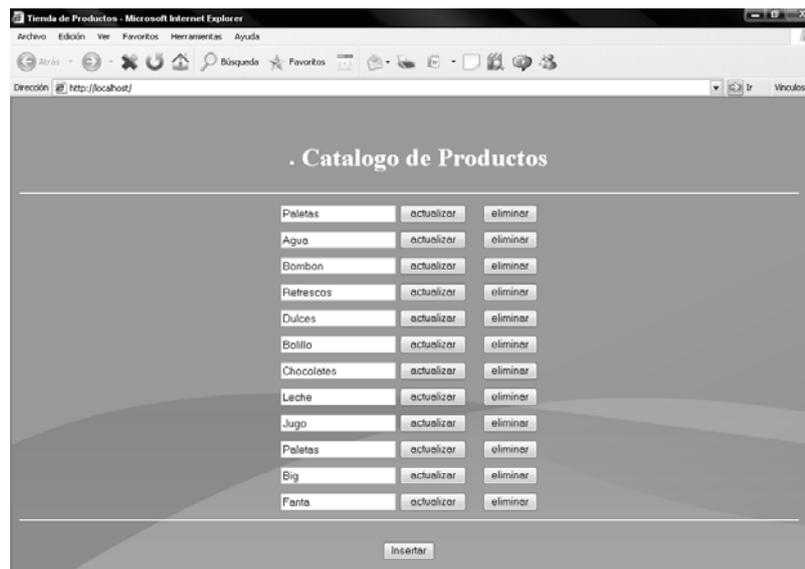
```

echo "<td><input type='submit' value='eliminar'></td>";
echo "<td></form></td>";
echo "<td></FORM></td>";
echo "</tr>";
echo "</TABLE>";
}
echo "<hr color='white'>";
echo "<center>";
echo "<form action='formulario2.html' method='POST'>";
echo "<input type='submit' value='Insertar'>";
echo "</form>";
echo "</center>";
?>
</center>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de PHP, Proyecto,index.php.

A continuación en la figura V.24 se muestra la agrupación de PHP y MySQL en un catálogo de productos:



V.24 Agrupación de PHP y MySQL en un Catálogo de Productos.

✓ Formulario2.html

```

<html>
<head>
<title>Inserta Productos</title>
</head>
<body background="./sprincess.jpg" text="darkblue">
<blockquote>
<blockquote>
<blockquote>
<blockquote>

```

```

<blockquote>
<h1>Insertar Productos</h1>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
<br>
<br>
<br>
<blockquote>
<blockquote>
<form action='conexion.php' method='POST'>
Clave : <input type='text' name='Clave'>
<br>
<br>
Producto :<input type='text' name='Producto'>
<br>
<br>
<input type='submit' value='Insertar'>
</form>
</blockquote>
</blockquote>
</body>
</html>

```

Este ejercicio lo podemos ver en la carpeta de PHP, Proyecto, formulario2.html.

A continuación en la figura V.25 se muestra la inserción de Productos:



V.25 Inserción de Productos.

Una vez que terminamos de insertar el producto la siguiente ventana que nos muestra es:

✓ Conexion.php

```
<html>
<body background="/.9.jpg">
<?php
    $Producto=$_POST['Producto'];
    $Clave=$_POST['Clave'];
    $r=mysql_connect("localhost","root","cursophp");
    mysql_db_query("producto","insert into productos values($Clave,$Producto)");
    echo "<br><br><br>";
    echo "<center><font color='darkblue'>";
    echo "<h1><blockquote>El Producto ha sido insertado
correctamente</blockquote></h1>";
    echo "<br><br><br>";
    echo "<center><img src='10.bmp' widht=150 height=150></center>";
    echo "<br><br><br>";
    echo "<h1><a href='./index.php'>Regresar</a></h1>";
    echo "</font></center>";
?>
</body>
</html>
```

Este ejercicio lo podemos ver en la carpeta de PHP, Proyecto, conexión.php.

A continuación en la figura V.26 se muestra mensaje de aceptación del producto:



V.26 Mensaje de Aceptación del Producto.

De esta manera acabamos de insertar ya un producto dentro de la base de datos de MYSQL, debemos recordar que hay que tenerla encendida, en estos momentos tal y como se explica en el anexo 1.

✓ Actualizar.php

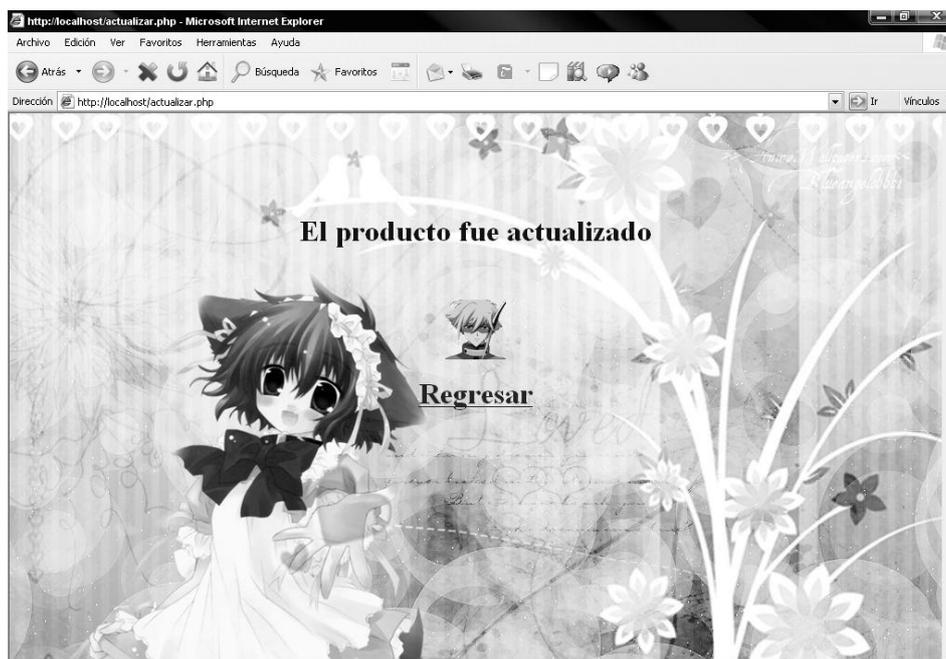
<?php

```
echo "<body background='./actualizar.jpg'>";  
$nom_produc=$_POST['Descripcion'];  
$Clave=$_POST['Clave'];  
$r=mysql_connect("localhost","root","cursophp");  
mysql_db_query("producto","update productos set Producto='$nom_produc'  
where Clave=$Clave");  
echo "<br><br><br><br><br>";  
echo "<center>";  
echo "<font color='blue'><h1>El producto fue actualizado</h1></font>";  
echo "<br><br>";  
echo "<img src='./wizard.gif'>";  
echo "<br><br>";  
echo "<h1><a href='./index.php'>Regresar</a></h1>";  
echo "</center>";  
echo "</body>";
```

?>

Este ejercicio lo podemos ver en la carpeta de PHP, conexión.php.

A continuación en la figura V.27 se muestra mensaje de actualización del producto:



V.27 Mensaje de Actualización del Producto.

✓ Eliminar.php

<?php

```
echo "<body background='./9.jpg' link='darkblue' vlink='green'>";  
$nom_produc=$_POST['Descripcion'];  
$Clave=$_POST['Clave'];  
$r=mysql_connect("localhost","root","cursophp");
```

```

mysql_db_query("producto"," delete fromsprincessproductos where
Clave=$Clave");
echo "<br><br><br>";
echo "<center><font color='darkblue'>";
echo "<h1><blockquote>El Producto ha sido borrado</blockquote></h1>";
echo "<br><br><br>";
echo "<center><img src='10.bmp' widht=150 height=150></center>";
echo "<br><br><br>";
echo "<h1><a href='./index.php'>Regresar</a></h1>";
echo "</font></center>";

```

?>

De esta manera eliminamos el producto de la base de datos automáticamente.

Este ejercicio lo podemos ver en la carpeta de PHP, Proyecto, eliminar.php.

A continuación en la figura V.28 se muestra mensaje de eliminación del producto:



V.28 Mensaje de Eliminación del Producto.



Conclusiones

Conclusiones.

Se puede concluir el trabajo diciendo, que las tecnologías Web, se han convertido en algo indispensable, para nuestras vidas, ya que con el tiempo, cada vez más, las personas están utilizando el internet, con lo cual las tecnologías Web, han tomando un gran papel.

El futuro de la Web debe tratar de los defectos que tiene la Web actual, algunas de las soluciones que se están haciendo es por medio de la **Web semántica**. El término "Web semántica" corresponde a una visión en que tanto los ordenadores –software– como las personas podrán encontrar, leer y comprender la información.

Cada vez más dispositivos portátiles permiten tomar imágenes digitales y video. Los cuales pueden ser subidos a diferentes páginas y ser mostrados dentro de la web. En este campo, ya empezamos a ver herramientas que nos permiten mezclar videos, música e imágenes directamente desde los sitios. Dándonos la oportunidad de crear no sólo nuevo contenido en forma de texto, sino también como multimedia. A su vez, es ya muy común el uso de etiquetas para clasificar el contenido que se suba a estos sitios. Permitiendo así que el contenido pertenezca a diferentes categorías al mismo tiempo, agilizando futuras búsquedas.

Personal y profesionalmente, el uso de las tecnologías web me parece una estupenda forma de mejorar la manera de comunicarse de las personas y de algunos negocios, los cuales actualmente, ya se encuentran haciendo uso de este medio, para promocionar sus productos.

Con la realización de varios cursos de programación en Web, observé que muchos de los estudiantes de la Facultad de Estudios Superiores Aragón, tienen la idea de que un sitio Web, es algo muy difícil de realizar, pero a través de varios cursos que se impartieron, pudieron observar que no es nada complicado realizarlo, solo hace falta un manual, el cual los lleve paso a paso, en la realización de su página y algo de creatividad.

Durante la realización de estos cursos participaron muchos estudiantes y algunos profesores de la Facultad de Estudios Superiores Aragón, entre la carreras que participaron están: Ingeniería en Computación, Ingeniería Mecánica Eléctrica, Pedagogía, Relaciones Internacionales, en las cuales, las Tecnologías Web están muy relacionadas, ya que con estas pueden difundir su información a muchos lugares.

Como se puede observar no solo el ingeniero necesita de esta herramienta cualquier profesionalista la puede usar y ese el motivo por el cual se realizó este manual, como un apoyo a la comunidad de la Facultad de Estudios Superiores Aragón, para todas aquellas personas quieran adentrarse a este tipo de programación.

Además de observar que conforme las personas van conociendo cada vez más lo que puede llegar a realizar, comienza a querer empaparse más sobre estas tecnologías, al menos eso fue lo que puede observar, ya que al finalizar los cursos ellos mismos proponían que se impartieran nuevos cursos, en los cuales ellos ya habían leído el tema,

pero no lograban comprenderlo del todo, con lo cual proponían que se diera otro curso referente al tema.

Cabe destacar que estas personas, lograron realizar sitios con mucha creatividad y muy llamativos para los usuarios de internet, además de mencionar que ciertas que tomaron los cursos ahora han podido realizar sitios para negocios, empresas, o simplemente para promocionar las habilidades que cada uno tiene.

Conforme se fue elaborando este manual, me di cuenta de la necesidad de documentar, hasta lo más esencial, ya que para las personas que manejamos la programación algunos de los términos utilizados en este manual, pueden ser muy claros, pero también hay que pensar en las personas que no conocen nada de programación y quieren realizar un sitio, lo puedan hacer, explicando algunos puntos que son claves en la programación, con los cuales espero, que las generaciones que lean este manual, tengan las bases para poder programar estas tecnologías más avanzadas, como lo es SMIL, la cual utiliza como base fundamental la programación de todas las tecnologías presentadas en este trabajo, además de que este tipo de programación en la actualidad y por lo que puedo percibir en un futuro, estará integrada, como una base muy fuerte en los compiladores más destacados como: Visual Studio, Quicktime, QT, Open GL, etc.

Ahora gracias a las experiencias que me ha dejado el realizar este manual, y la realización de estos cursos, me di cuenta de mi gusto por programar este tipo de tecnologías, y que han servido como base fundamental de todo lo que he realizado durante mis estudios profesionales, así como de los logros que he tenido desde que finalice mis estudios, con los cuales pienso dedicarme a la programación con este tipo de tecnologías y poder realizar programas, los cuales puedan ayudar a la comunidad de la Facultad de Estudios Superiores Aragón y porque no a la comunidad de la UNAM.

Además de convertirme en una programadora exitosa, tanto el lenguajes de programación estructurada, como el uso de las tecnologías Web para poderlas combinar y realizar buenas aplicaciones en realidad virtual para el uso de la comunidad.

Finalmente puedo decir que dentro de este trabajo se encuentran todas, las sentencias más importantes en la realización de sitios Web, sin embargo me queda muy claro, que se pueden añadir, más tecnologías Web, las cuales harían mucho más consistentes nuestros sitios Web algunos de ellas son: XHTML, AJAX, SMIL, SVG, XSL, OPEN GL ETC.



Anexo1

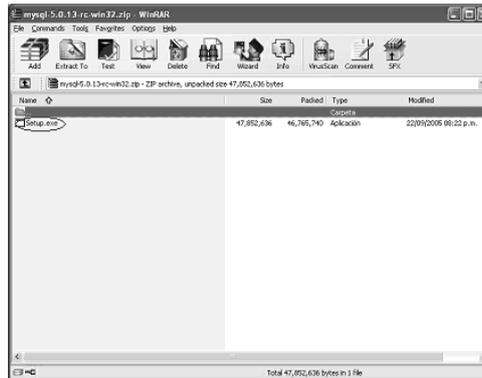
Instalación de WAMP y MySQL

Anexo I

Instalación de WAMP y MYSQL.

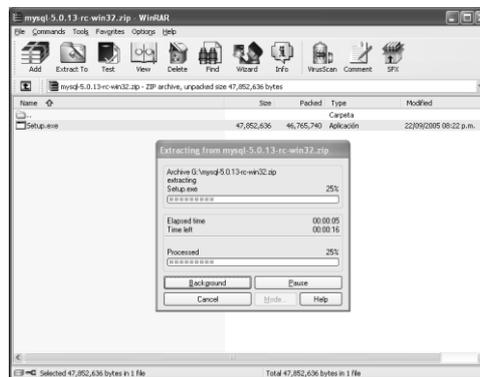
Instalación de MYSQL.

1. Dentro del disco se encuentra una carpeta llamada Disco PHP, aquí se encuentra un comprimido llamado mysql-5.0.13-rc-win32, debemos abrirlo, para ver su contenido como se muestra en la figura AI.1.



AI.1 Contenido del comprimido mysql-5.0.13-rc-win32

2. Cuando podamos ver esta ventana nos encontraremos, con un archivo llamado SETUP.EXE, al cual debemos dar doble clic, para comenzar la instalación como se muestra en la figura AI.2 .



AI.2 Comienzo de la instalación

3. Una vez que podamos observar esta ventana, debemos dar next, para continuar con la instalación, como se muestra en la figura AI.3.



AI.3 Presentación de Instalación.

4. Cuando se nos presente la siguiente ventana debemos escoger la opción típica y dar clic en next, como se muestra en la figura A1.4.



A1.4 Selección de Instalación

5. A continuación se mostrará otra ventana en donde podremos ver el proceso de la instalación como se muestra en la figura A1.5.



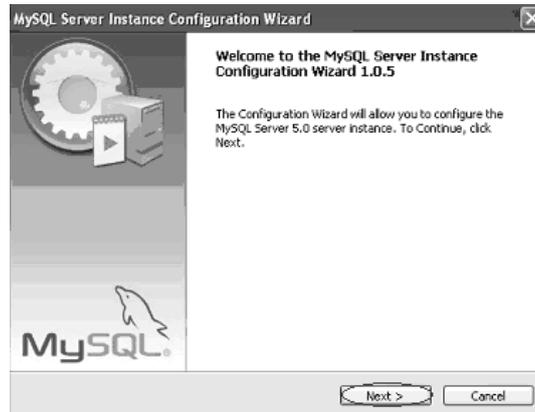
A1.5 Proceso de Instalación.

6. Una vez que termina de instalar nos muestra la siguiente ventana, de la cual debemos escoger la opción de SKIP SIGN-UP y dar clic en next como se muestra en la figura A1.6.



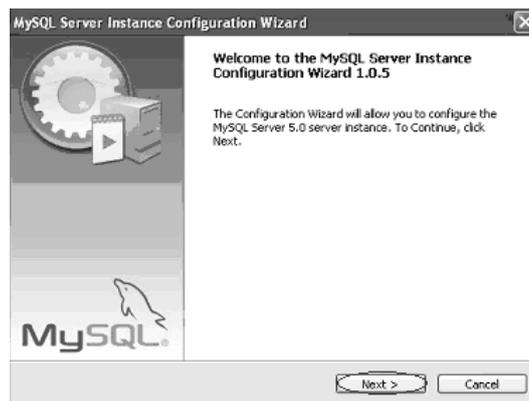
A1.6 Creación de Cuenta

7. Cuando damos clic en next, nos aparece la ventana de finalizar, si damos clic en finalizar nos aparecerá la siguiente ventana como se muestra en la figura A1.7.



A1.7 Finalización de MySQL

8. Una vez que finalizamos, a continuación nos parece la siguiente ventana, con la que continuaremos con nuestra instalación. Damos clic en next como se muestra en la figura A1.8.



A1.8 Configuración de MySQL

9. En la siguiente ventana que nos muestra instalador debemos escoger la opción **DETAIL CONFIGURATION**, y damos clic en next como se muestra en la figura A1.9.



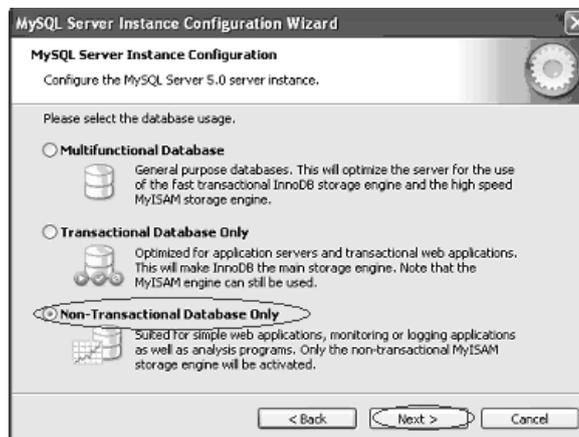
A1.9 Tipo de Configuración.

10. A continuación seleccionamos la opción DEVELOPER MACHINE, damos clic en next como se muestra en la figura A1.10.



A1.10 Tipo de Servidor

11. Ahora debemos escoger la opción NON-TRANSACTIONAL DATABASE, y damos clic en next como se muestra en la figura A1.11.



A1.11 Selección de Base de Datos

12. En la siguiente ventana seleccionamos la opción Decision Support , y damos clic en next como se muestra en la figura A1.12.



A1.12 Asignación de Usuarios para Utilizar el Software.

13. En esta nueva ventana seleccionamos el puerto que deseamos ocupar para la aplicación o simplemente dejamos el que se tiene por default como se muestra en la figura A1.13.



AI.13 Asignación del Puerto a Utilizar.

14. Después seleccionamos Standard Character Set como se muestra en la figura AI.14



AI.14 Selección del Idioma del Software

15. A continuación seleccionamos la opción INCLUDE BIN DIRECTORY IN WINDOWS PATH y damos clic en next como se muestra en la figura AI.15.



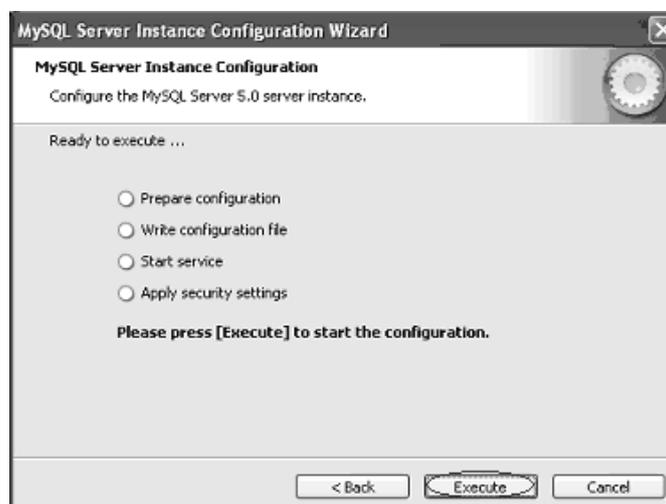
AI.15 Incluir Consola de Comandos de MySQL

16. En la siguiente ventana pedirá una contraseña, la cual debemos recordar ya que cada que iniciemos una en MYSQL, nos la pedirá, de igual manera la debemos colocar a la hora de trabajar con PHP. Una vez colocada la contraseña, damos clic en next como se muestra en la figura AI.16.



AI.16 Colocación de Contraseña.

17. En la siguiente ventana muestra las opciones que hemos especificado, pero pide la autorización para poderlas ejecutar, así que debemos dar doble clic en execute, las cuales aparecerán, en paloma azul las aplicaciones que ya fueron instaladas, si existe algún problema marca una x, y tenemos que verificar la configuración que hemos hecho, de lo contrario, aparecerá una ventana para poder finalizar como se muestra en la figura AI.17 y AI.18.

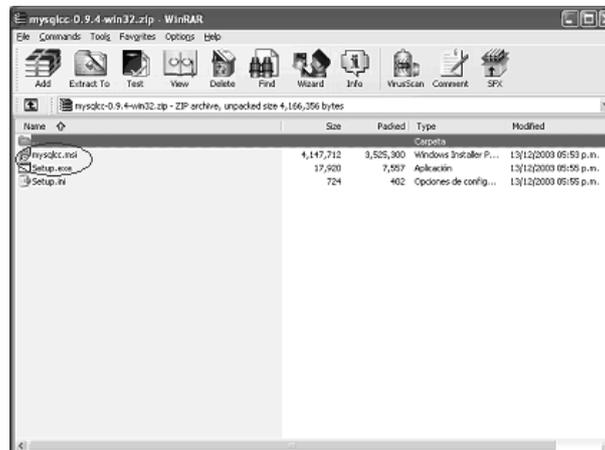


AI.17 Autorización para instalar las propiedades del software.



AI.18 Finalización correcta de instalación de propiedades.

18. Ahora debemos abrir el comprimido mysqlcc-0.9.4-win32 y dar doble clic en el setup como se muestra en la figura AI.19.



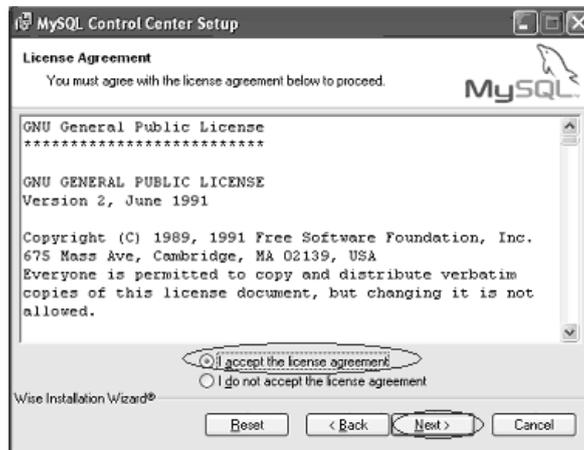
AI.19 Contenido del Comprimido mysqlcc-0.9.4-win32

19. A continuación aparece la siguiente ventana a la cual le debemos dar clic en next como se muestra en la figura AI.20.



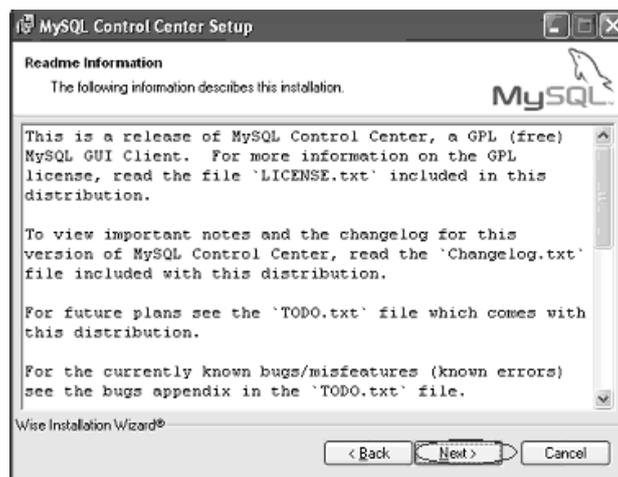
AI.20 Instalación de MySQL

20. Ahora pide que aceptemos las condiciones de uso del programa y damos clic en next. como se muestra en la figura AI.21.



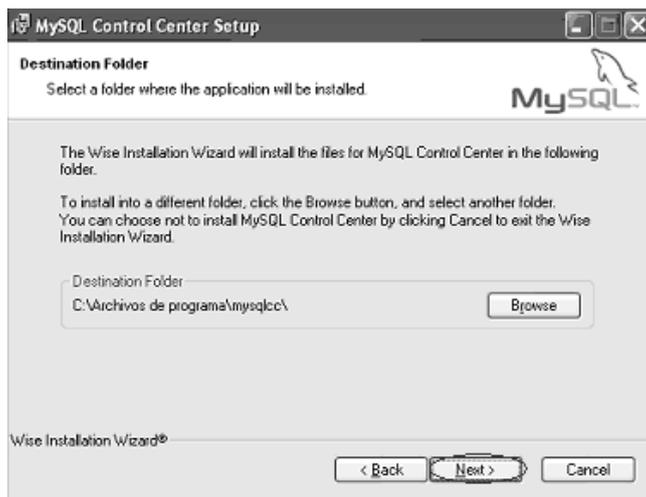
AI.21 Aceptar Licencia.

21. En la siguiente ventana damos clic en next como se muestra en la figura AI.22.



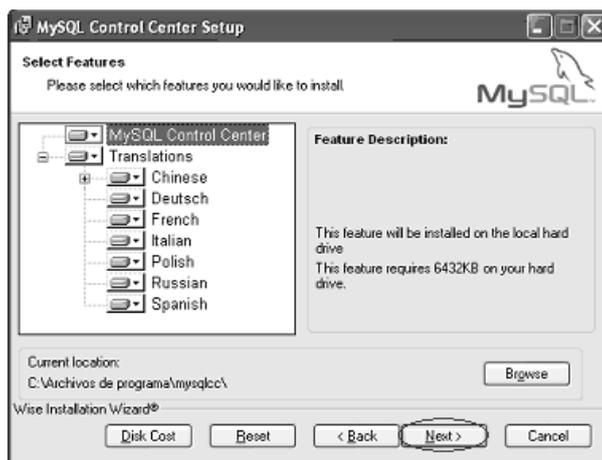
AI.22 Terminos y Condiciones.

22. En la siguiente ventana seleccionamos la dirección en donde queremos que se instale nuestro programa o podemos dejar, la que tiene por default como se muestra en la figura AI.23.



AI.23 Ubicación donde se instala el programa.

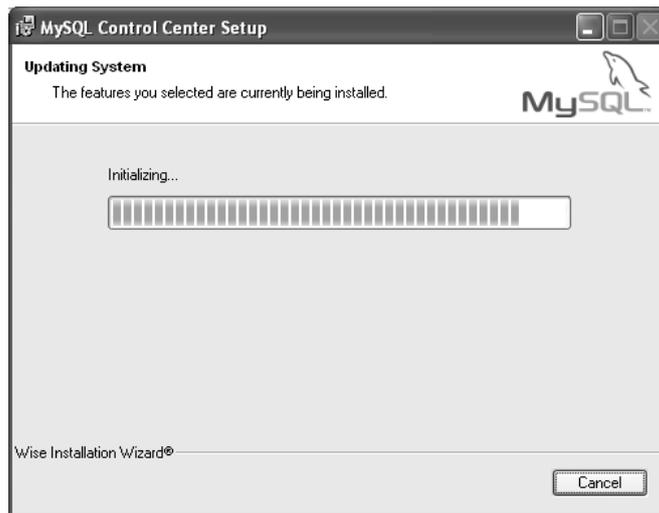
23. En las siguientes dos ventanas damos next, y comenzara la instalación del programa MYSQL, en forma gráfica como se muestra en la figura AI.24, AI.25 y AI.26.



AI.24 Instalación de los Controles de MySQL



AI.25 Confirmación de Instalación



AI.26 Proceso de Instalación.

24. A continuación muestra la pantalla para poder finalizar la instalación a la cual damos doble clic como se muestra en la figura AI.27.



AI.27 Fin de la Instalación.

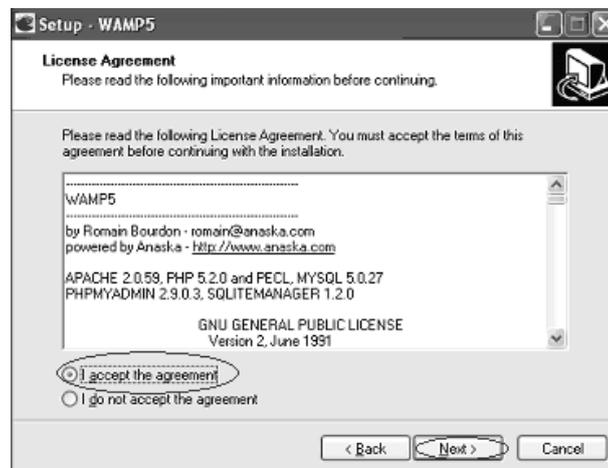
🚦 Instalación de WAMP¹.

1. Dentro del disco se encuentra el ejecutable wamp5_16.6, al cual damos doble clic y cuando nos aparezca la pantalla, damos clic en next como se muestra en la figura A1.28.



A1.28 Inicio de Instalación de WAMP 5

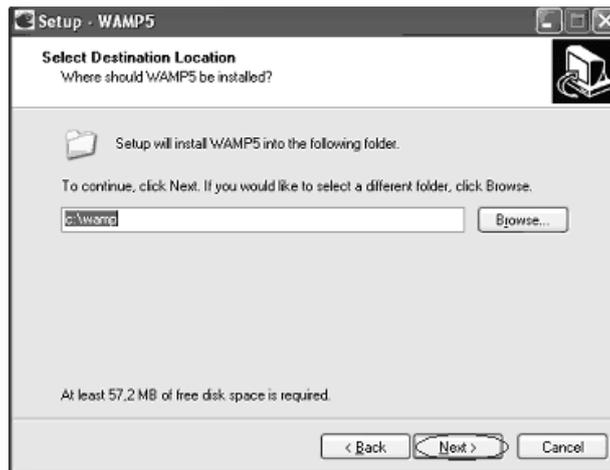
2. Aceptamos los términos de programa y damos clic en next como se muestra en la figura A1.29.



A1.29 Aceptación de Licencia.

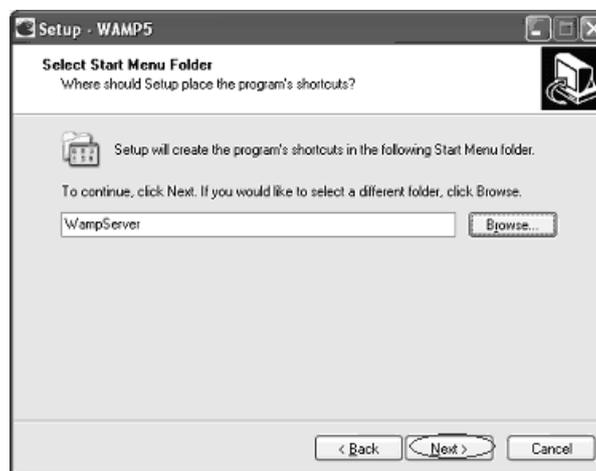
3. Seleccionamos la dirección en donde queremos que se instale el programa, y damos clic en siguiente como se muestra en la figura A1.30.

¹ Windows-Apache-MySQL- PHP/Python/PERL



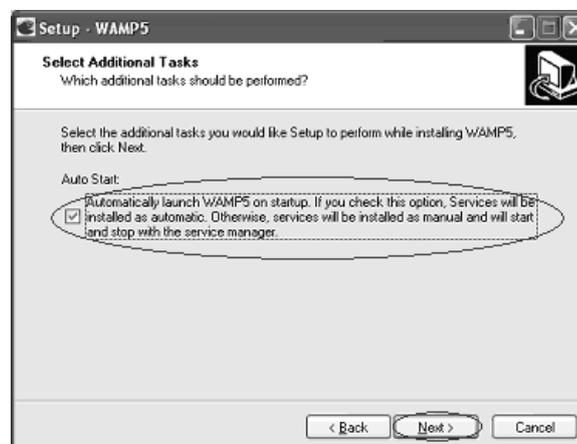
AI.30 Ubicación de Instalación del Programa

4. En la siguiente ventana damos clic en next como se muestra en la figura AI.31.



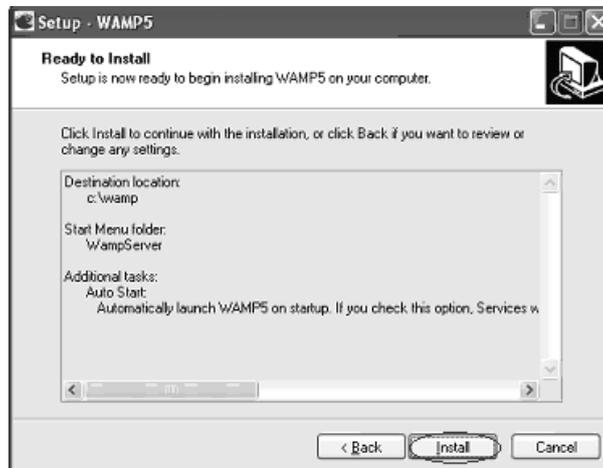
AI.31 Tipo de Servidor que utilizará

5. En la siguiente ventana seleccionamos la opción que nos marca, esto es para que la aplicación inicie automáticamente, cuando prendamos la computadora. Una vez seleccionada damos clic en next como se muestra en la figura AI.32.

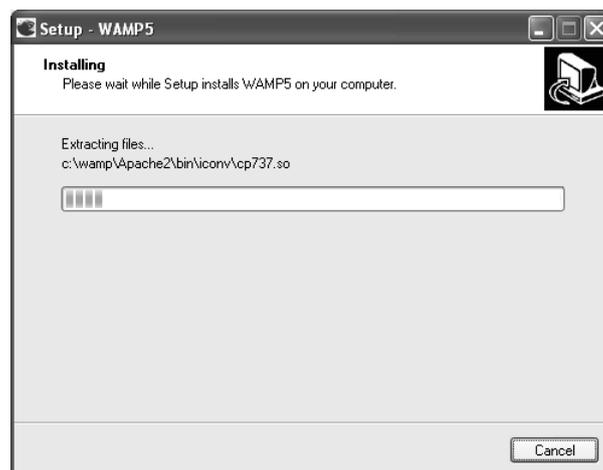


AI.32 Inicio Automático de WAMP5

6. A continuación aparece la ventana con la opción de instalar. En donde damos clic para que el programa se comience a instalar como se muestra en la figura AI.33 y AI.34.



AI.33 Instalación de WAMP 5



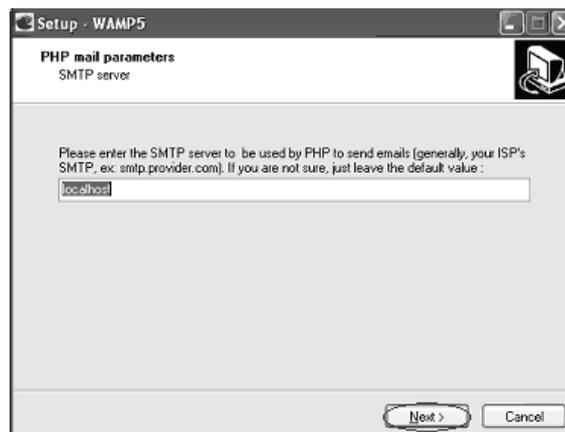
AI.34 Proceso de Instalación.

7. A continuación aparece una ventana en la cual correremos todos los programas de PHP, elegimos la carpeta www, que es la designada por php, y damos clic en aceptar como se muestra en la figura AI.35.



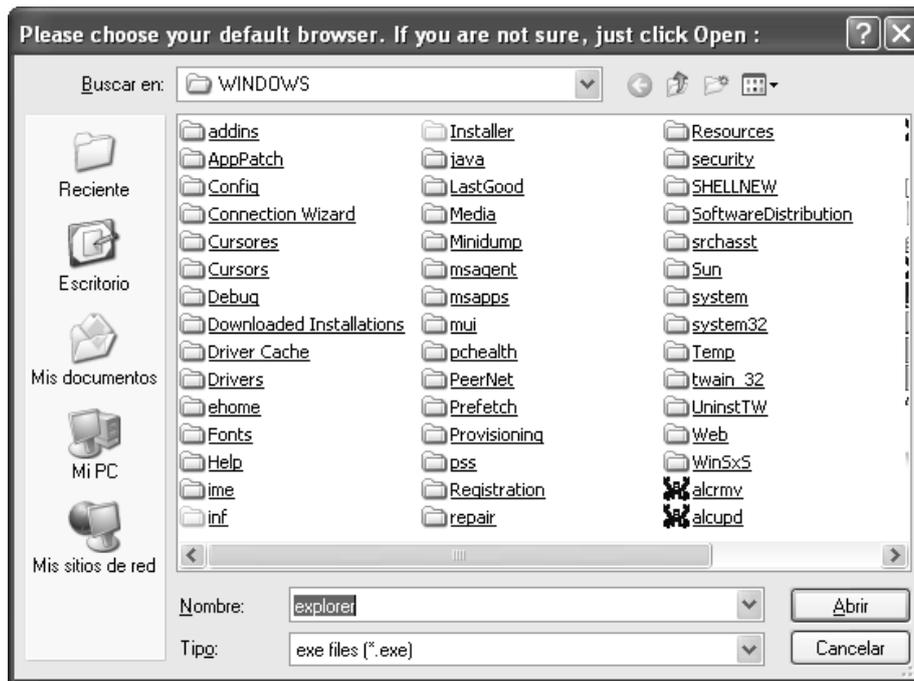
Al.35 Selección de Carpeta para guardar los programas elaborados

8. En la siguiente ventana nos muestra el directorio raíz de la aplicación, el cual es localhost, en esta ventana no debemos modificar nada solo dar clic en next como se muestra en la figura Al..36



Al.36 Directorio raíz de ejecución de programas.

9. Cuando damos clic en next aparece una ventana en donde debemos seleccionar la aplicación explorer, y damos clic en abrir, esto es para seleccionar el explorador con el cual correremos php como se muestra en la figura Al.37.



AI.37 Selección del explorador a utilizar.

10. A continuación muestra la ventana de finalización del programa en donde damos clic en finalizar como se muestra en la figura AI.38.



AI.38 Fin de la Instalación.



Anexo II

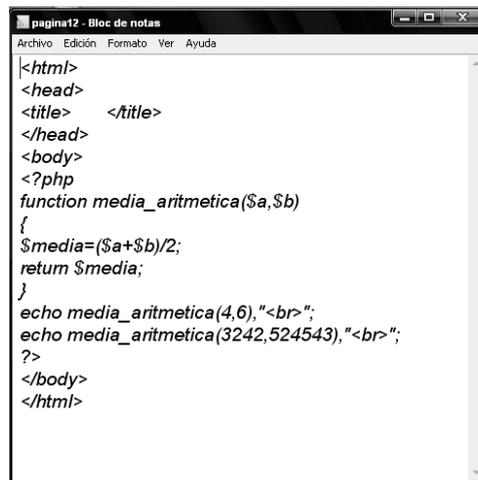
Utilización de WAMP y MySQL

Anexo II

Utilización de WAMP y MYSQL.

WAMP

1. Para poder correr los programas hechos en PHP necesitamos del programa WAMP, a continuación se muestra cómo se debe utilizar.
2. Antes de utilizar el programa debemos tener el programa capturado en un bloc de notas como se muestra a continuación en la figura All.1.



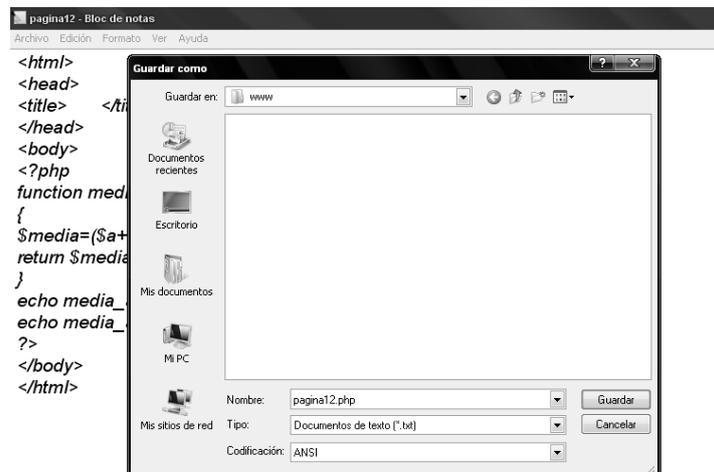
```
pagina12 - Bloc de notas
Archivo  Edición  Formato  Ver  Ayuda
</html>
<head>
<title>  </title>
</head>
<body>
<?php
function media_aritmetica($a,$b)
{
$media=($a+$b)/2;
return $media;
}
echo media_aritmetica(4,6),"<br>";
echo media_aritmetica(3242,524543),"<br>";
?>
</body>
</html>
```

All.1 Captura de Programa

3. A la hora de guardar la página debemos guardarla dentro de la carpeta WAMP, que seleccionamos durante la instalación, la ruta para llegar a ella es:

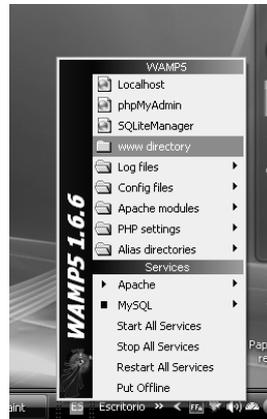
C:\wamp\www

4. Debemos colocarle un nombre y la extensión con la que guardaremos es .php como se muestra a continuación en la figura All.2.

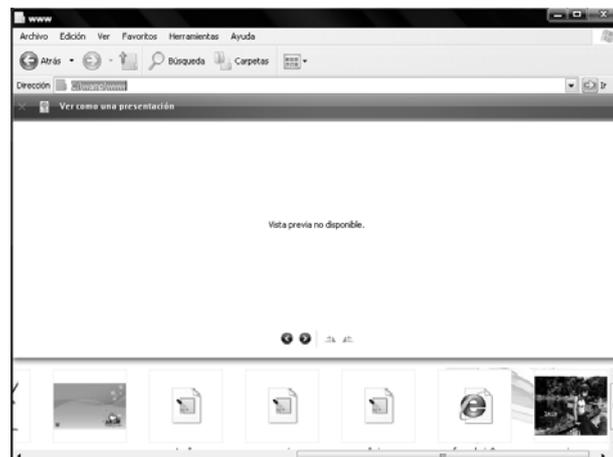


All.2 Guardar con extensión .php

5. Para estar seguros que la página se guardó correctamente lo podemos ver desde el acceso rápido que nos da WAMP. Seleccionamos la opción de www directory, y nos abrirá la carpeta de www como se muestra en la figura AII.3 y AII.4



AII.3 Directorio de WAMP



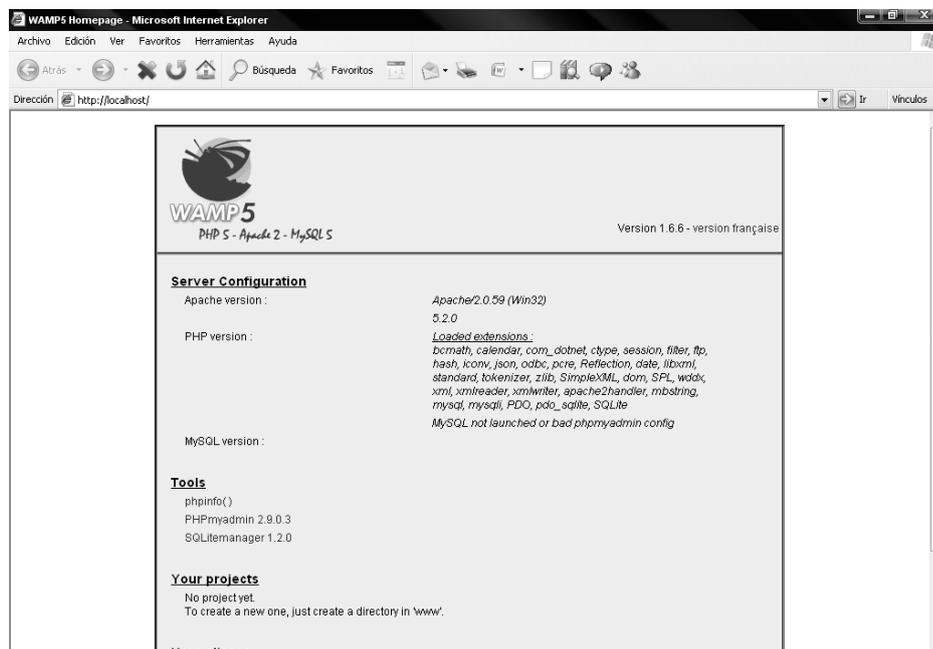
AII.4 Carpeta de Alojamiento de Programas.

6. Una vez que estamos seguros que la página se encuentra dentro de esta carpeta, lo último que se debe hacer es ir directo al programa WAMP, y ubicar la opción de localhost, la cual nos envía al directorio raíz como se muestra en la figura AII.5.



All.5 Ubicación de localhost dentro del directorio.

7. Aquí es muy importante mencionar que cualquier página que realicemos con PHP, solo podrá ser vista dentro de este programa, si la tratamos cargar en un explorador sin el programa no la veremos, en caso de colocarla en un servidor, el servidor debe tener las herramientas necesarias, para correr páginas hechas con PHP.
8. Una vez que seleccionamos la opción de localhost esta aparece el servidor de PHP, como se muestra a continuación en la figura All.6.



All.6 Servidor de PHP

9. Ya por último debemos colocar el nombre de la página junto con su extensión, en la barra de direcciones, como se muestra a continuación.

- ✓ Esta es la dirección que contiene la página actualmente como se muestra en la figura All.7



All.7 Dirección de localhost

- ✓ Colocaremos el nombre de la página como se muestra a continuación en la figura All.8.



All.8 Colocación del nombre de la página a compilar.

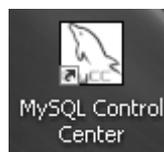
10. El resultado final se muestra a continuación el la figura All.9:.



All.9 Visualización de Página.

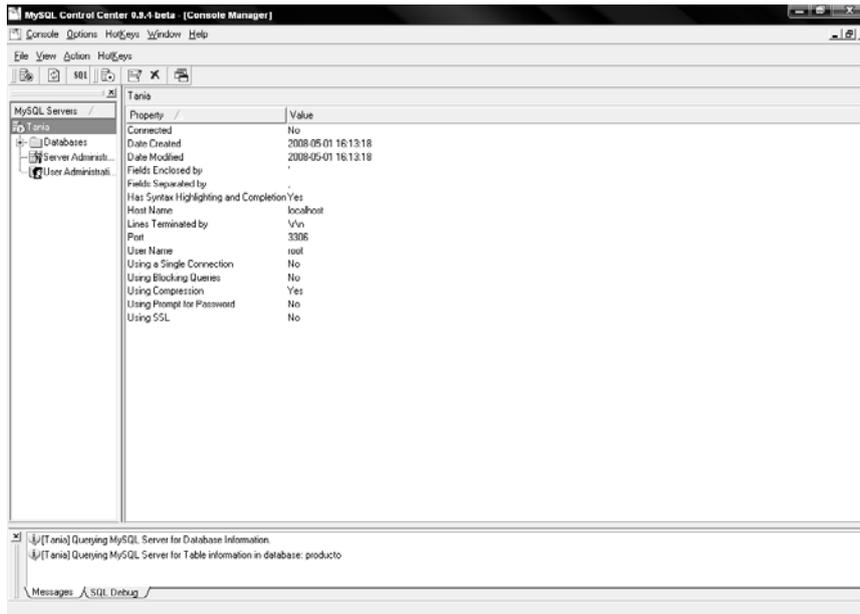
MySQL

1. Para comenzar a trabajar con MySQL, debemos dar doble clic en el icono que se muestra a continuación en la figura All.10.



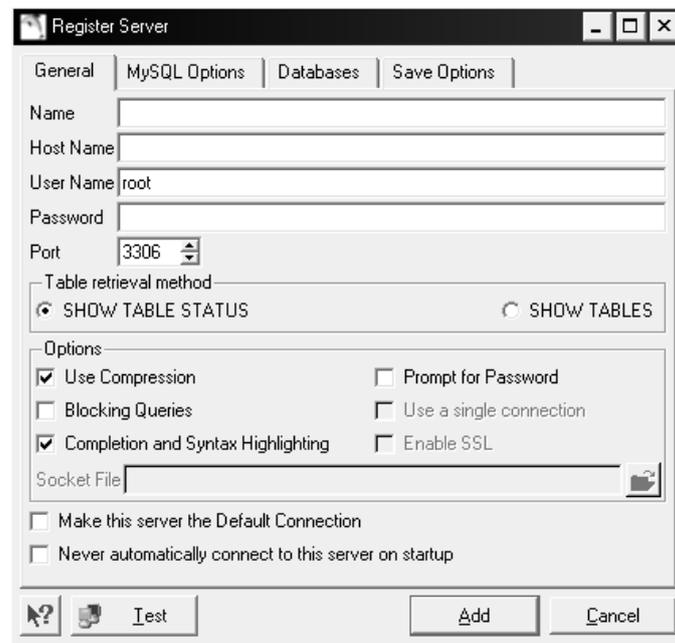
All.10 Ícono de MySQL

2. Con esto abrimos la aplicación en forma gráfica como se muestra a continuación en la figura All.11.



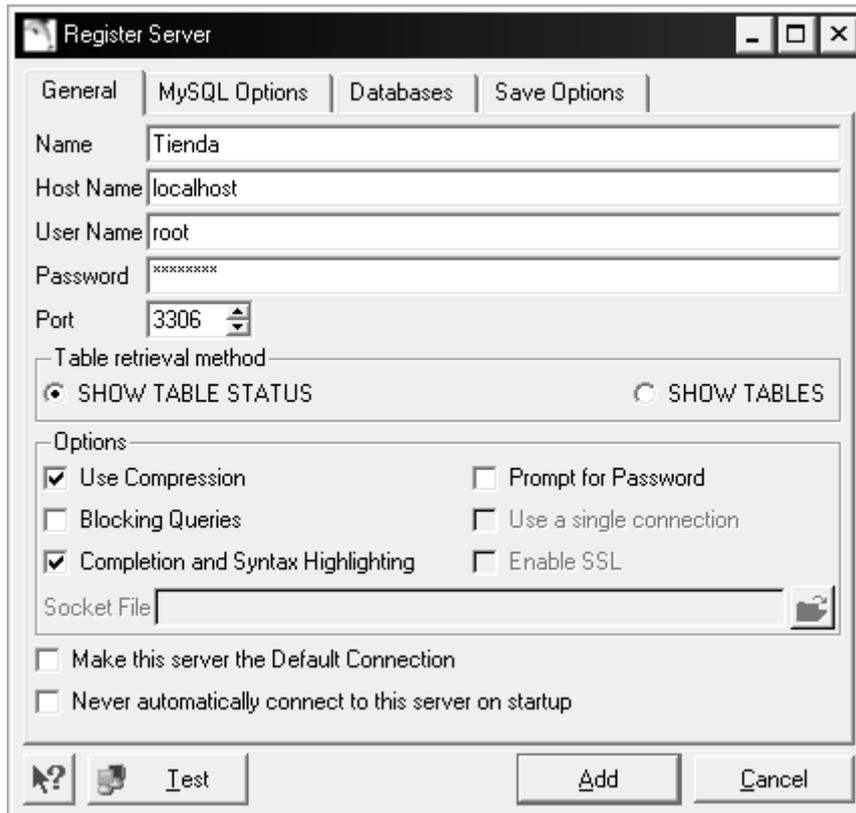
All.11 Vista de Aplicación

3. A continuación manda una ventana en donde pide que debemos especificar el servidor a utilizar, como se muestra a continuación en la figura All.12.



All.12 Registro de servidor

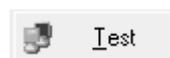
4. La forma de llenado de esta ventana es la siguiente, como se muestra en la figura All.13:



All.13 Llenado de ventana

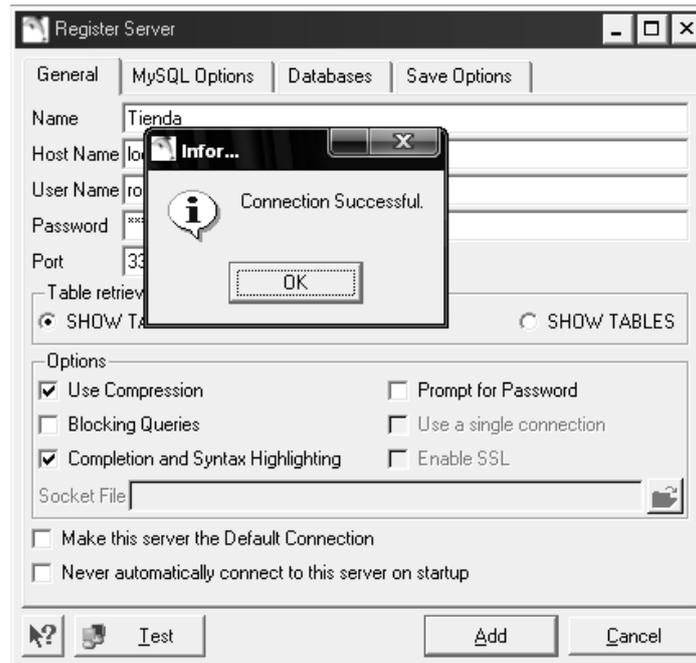
En la opción name se coloca el nombre que tendrá el servidor, en host name, se coloca el nombre del directorio raíz de WAMP que es localhost, y en password, colocamos la contraseña que colocamos durante la instalación de MYSQL.

5. Para comprobar que estos datos son correctos, se tiene una herramienta dentro de la misma ventana: se tiene la opción Test, a la cual le damos clic como se muestra en la figura All.14.



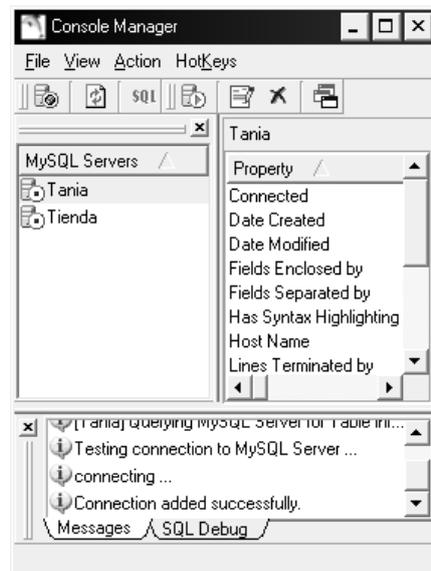
All.14 Opción de introducción de datos correctos.

6. Si cuando damos clic nos aparece una ventana la cual nos indica que la conexión ha sido exitosa, damos clic en aceptar, y después clic en ADD. De lo contrario nos mostrará un error y se tendrá que realizar nuevamente la configuración anterior. Como se muestra en la figura All.15.



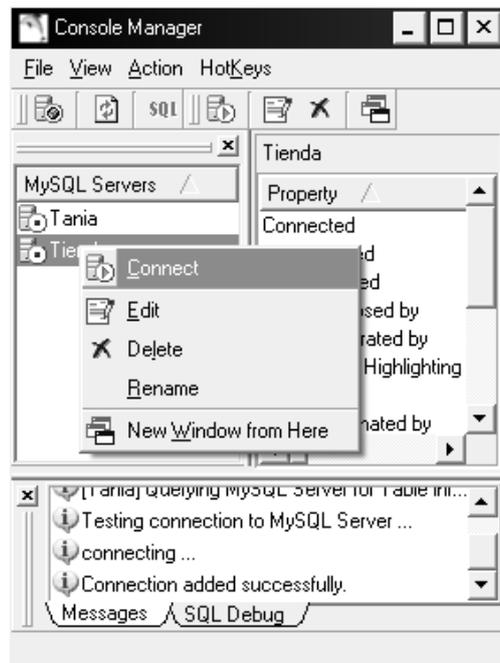
All.15 Conexión exitosa.

7. Ahora tenemos el servidor instalado en nuestro programa, como se muestra en la figura All.16.



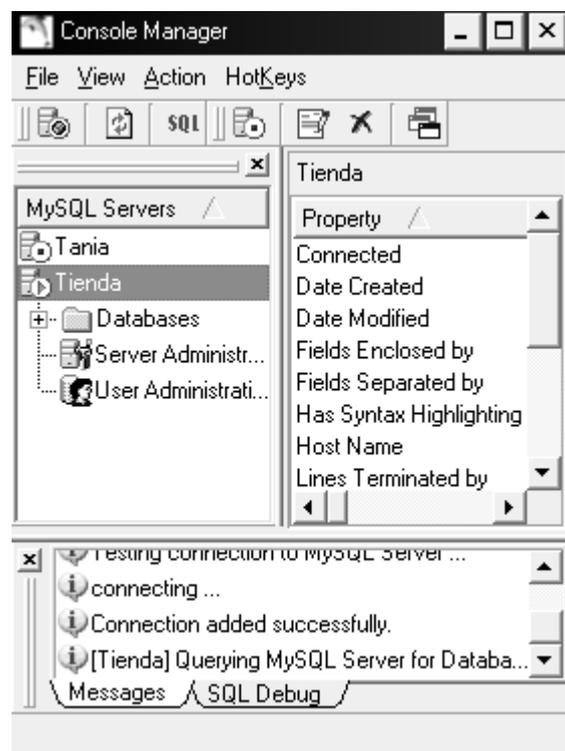
All.16 Servidor Instalado

8. Ahora solo queda, conectar el servidor para que comience a funcionar, seleccionado el servidor que deseamos conectar, damos clic derecho y a continuación aparece una ventana en la cual seleccionaremos la opción connect como se muestra en la figura All.17.



All.17 Conexión del Servidor.

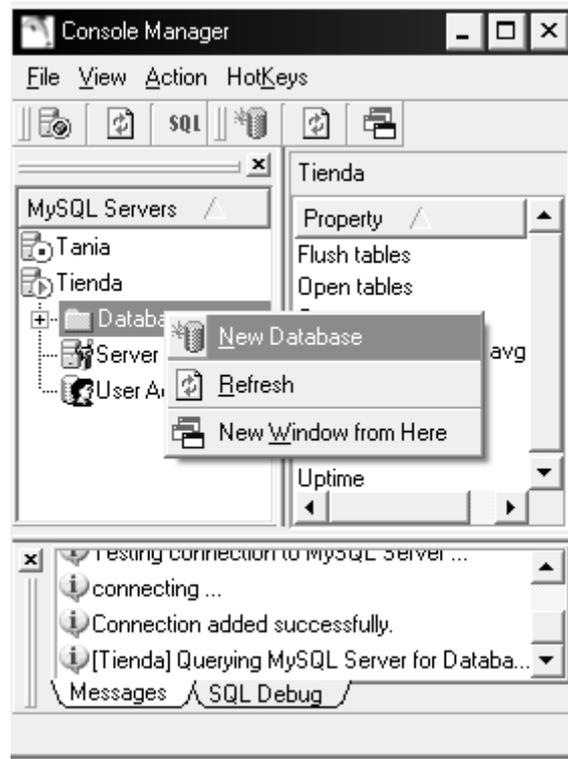
9. Una vez que hemos conectado el servidor, nos muestra la carpeta de base de datos, que contiene, además de otras herramientas que se necesitan para administrar una base de datos, como se muestra en la figura All.18.



All.18 Vista del servidor en funcionamiento.

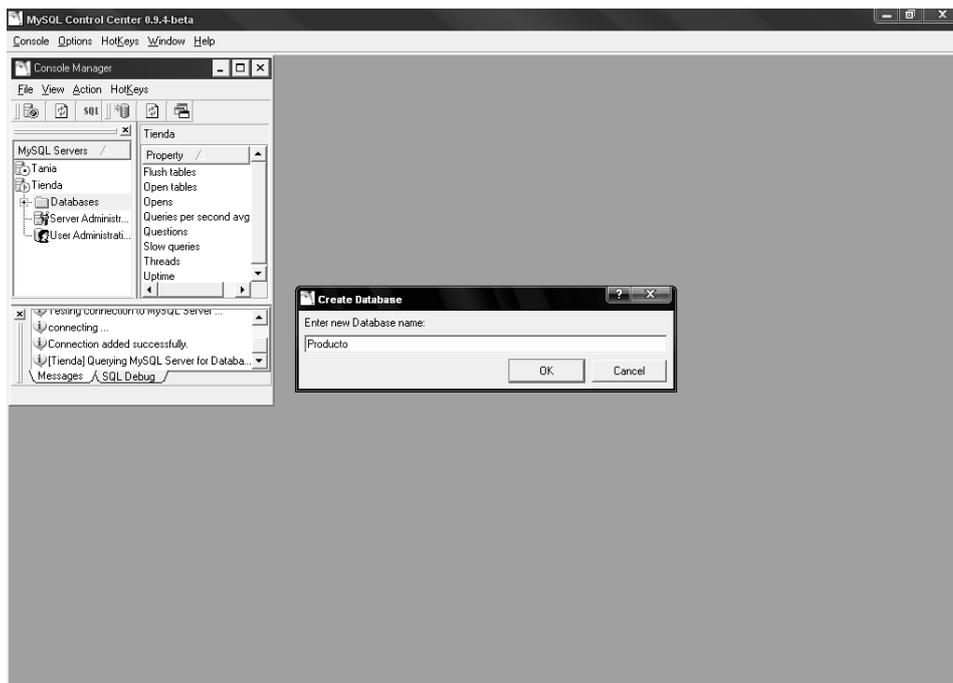
10. Ahora debemos crear la base de datos, para hacerlo necesitamos seleccionar la carpeta Databases, que se encuentra dentro del servidor, y damos clic derecho, para que muestre las opciones que tenemos, dentro de estas

opciones muestra la opción New Database, la cual seleccionamos como se muestra en la figura AII.19.



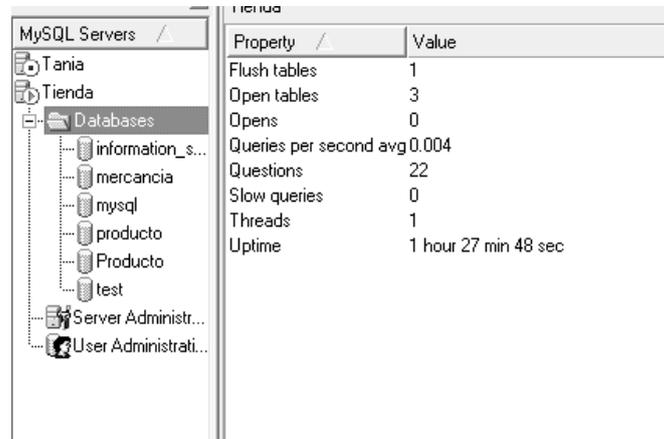
AII.19 Crear base de datos.

11. Una vez que la seleccionamos, manda una ventana en la cual pide el nombre que tendrá la base de datos, colocamos el nombre y damos aceptar, para crear la base de datos, como se muestra en la figura A2.20.



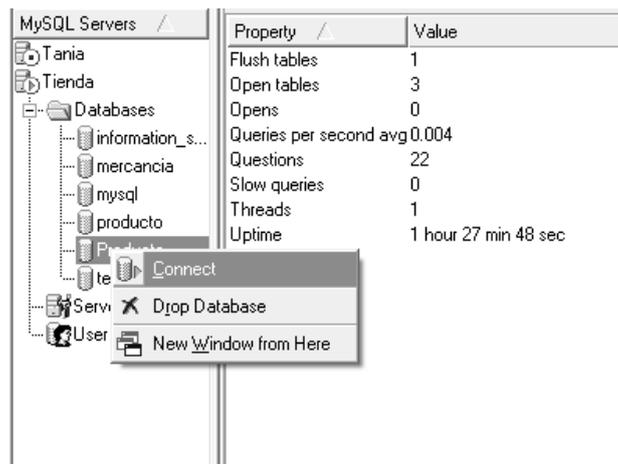
AII.20 Nombrar base de datos.

12. Una vez que damos aceptar, crea la base de datos, cuando desplegamos la carpeta de Databases, aparece la base de datos que acabamos de crear, como se muestra en la figura A2.21.



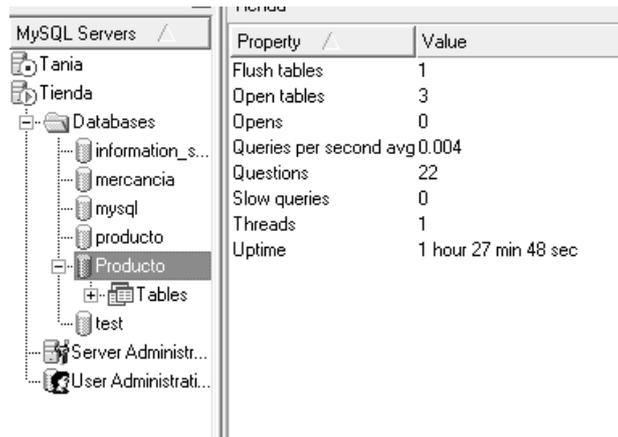
All.21 Inserción de base de datos.

13. Seleccionamos la base de datos, damos clic derecho, y seleccionamos la opción de conectar base de datos, como se muestra en la figura All.22.



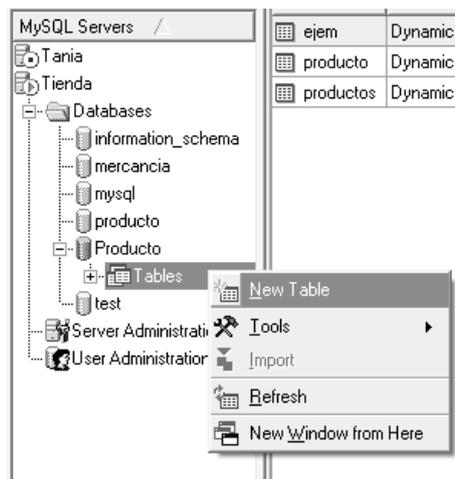
All.22 Conexión de la base de datos.

14. Una vez que está conectada la muestra de la siguiente como en la figura All.23.



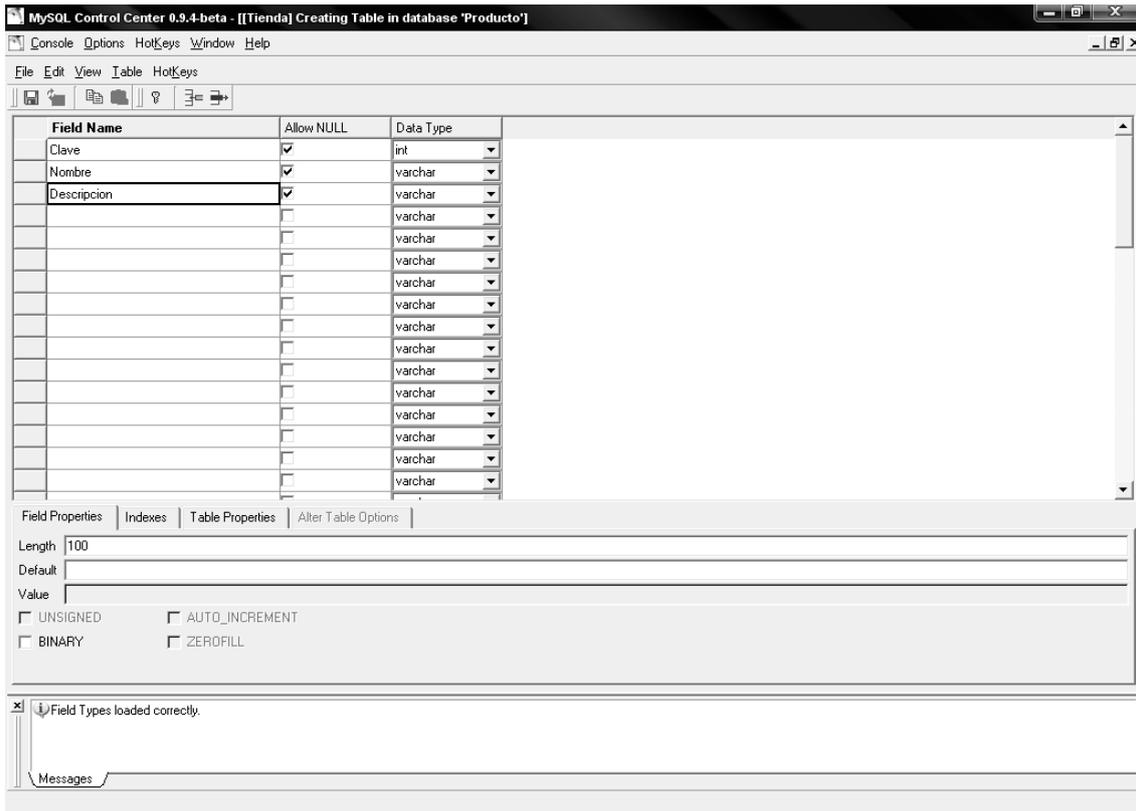
All.23 Base de datos conectada

15. Una vez que la base de datos está conectada, permite ver la opción de tablas, en donde, si seleccionamos la opción de tables y damos clic derecho sobre ella, nos aparece una ventana con la opción new table, esto permitirá crear una tabla para la base de datos, como se muestra en la figura A2.24.



All.24 Crear tabla de datos.

16. Aparece una ventana en donde podemos colocar los atributos que tendrán la tabla, así como seleccionar el tipo de variable que utilizaremos, como se muestra en la figura All.25.



AII.25 Inserción de Atributos a la Tabla

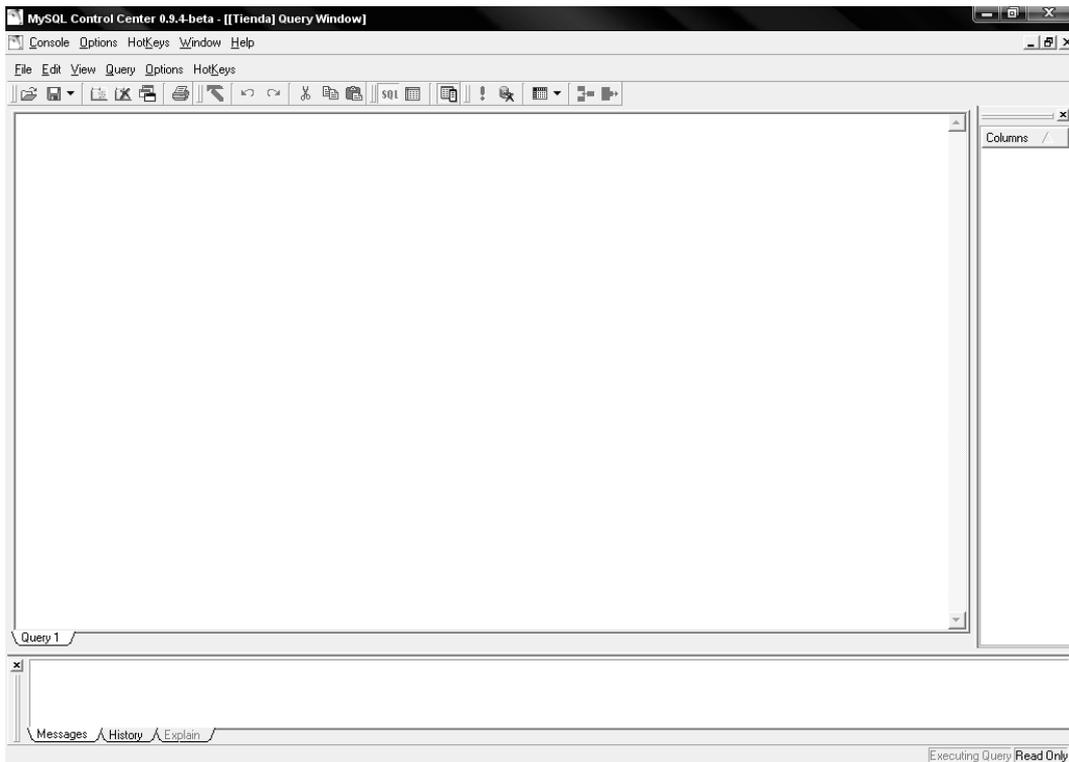
17. Para salvar esta tabla debemos dar clic en el siguiente símbolo de la figura AII.26



AII.26 Salvar Tabla.

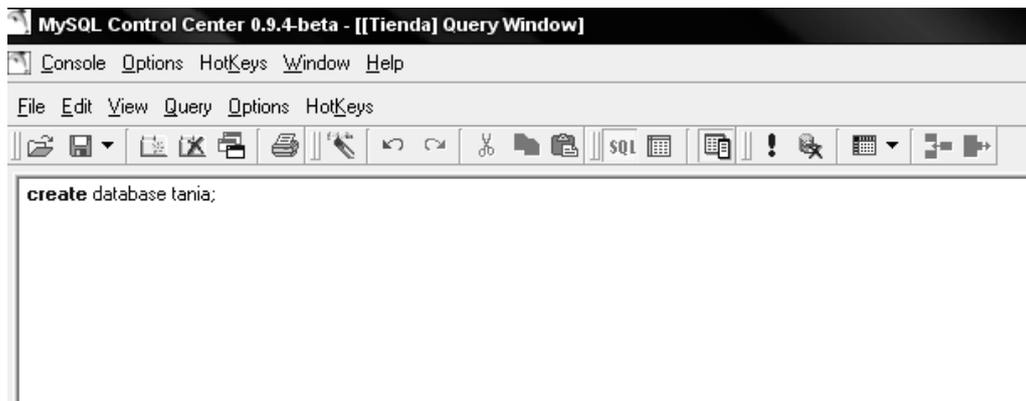
18. A continuación pedirá el nombre de la tabla como se muestra en la figura AII.27.

21. A continuación aparecerá una ventana como la que se muestra en la figura A2.30:



All.30 Vista del Editor de Comandos de SQL.

22. Dentro de esta ventana colocaremos todo el código de SQL, como se muestra en la figura All.31:

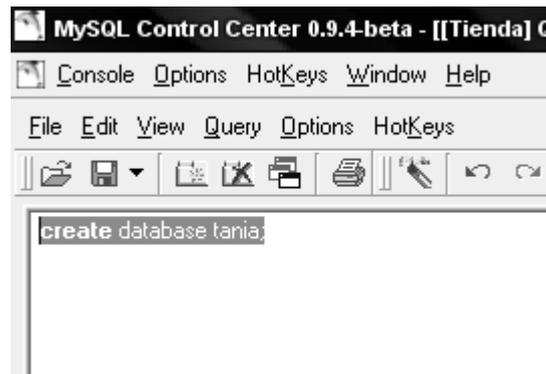


All.31 Inserción del código MySQL.

23. Una vez que hemos escrito el código necesarios, debemos compilarlo, para que pueda hacer las instrucciones indicadas, por nosotros. Esto lo hacemos seleccionando todo el código que deseamos ejecutar, además de dar clic en el siguiente icono que se muestra en la figura All.32 y All.33.

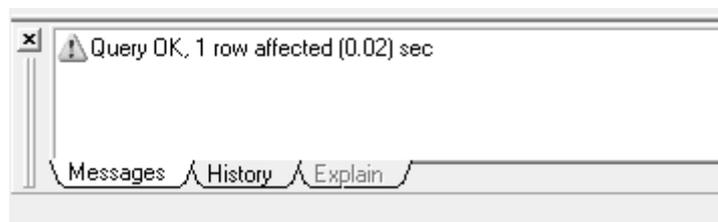


All.32 Ejecutar código.

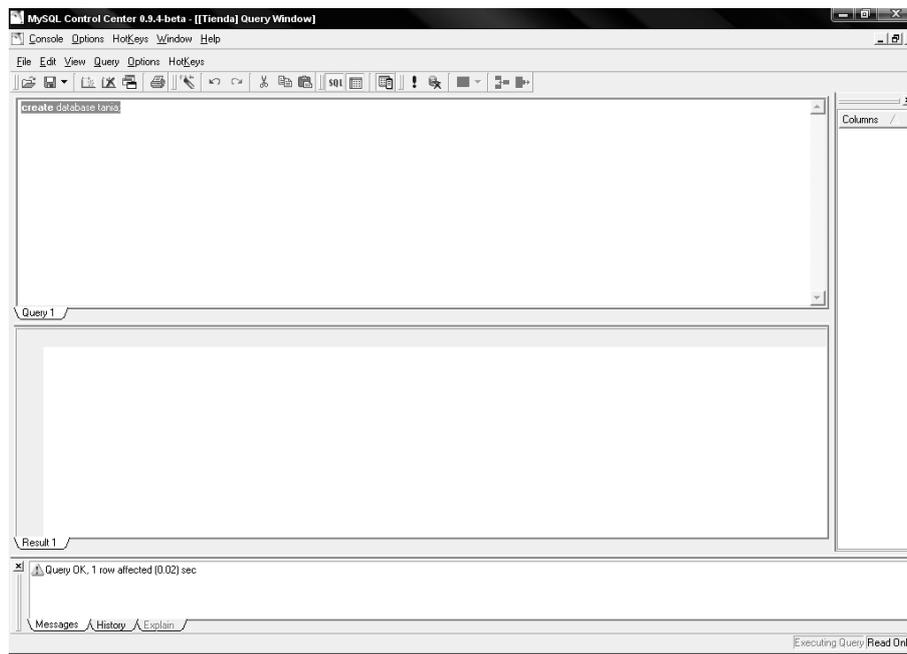


All.33 Selección del código a ejecutar.

24. Si la sentencia es correcta nos mandará un mensaje de aceptación como el que se muestra en la figura All.34 y All.35:

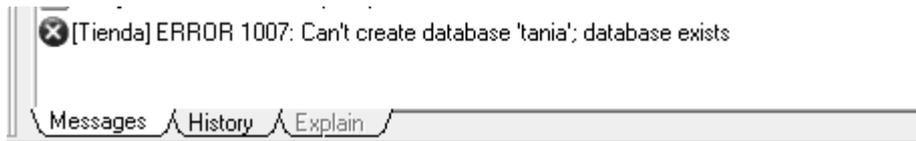


All.34 Mensaje de Aceptación de comandos.



All.35 Vista completa del compilador

25. En caso de tener un error de sintaxis en el código nos mandará un error, dando doble clic sobre el error, nos posicionará sobre la línea que tiene error, para poderla corregir, hasta que tengamos la sintaxis correcta quitará este error, como se muestra en la figura All.36:



All.36 Mensaje de introducción errónea de comandos.

26. Todas las instrucciones que se manejan dentro del capítulo V, se manejan de esta manera.



Anexo III

Colores en Hexadecimal

Anexo III

Colores en Hexadecimal.

nombre HTML	código hex R G B	código decimal R G B	nombre HTML	código hex R G B	código decimal R G B
Colores Rojos			Colores Marrones		
IndianRed	CD 5C 5C	205 92 92	Cornsilk	FF F8 DC	255 248 220
LightCoral	F0 80 80	240 128 128	BlanchedAlmond	FF EB CD	255 235 205
Salmon	FA 80 72	250 128 114	Bisque	FF E4 C4	255 228 196
DarkSalmon	E9 96 7A	233 150 122	NavajoWhite	FF DE AD	255 222 173
LightSalmon	FF A0 7A	255 160 122	Wheat	F5 DE B3	245 222 179
Crimson	DC 14 3C	220 20 60	BurlyWood	DE B8 87	222 184 135
Red	FF 00 00	255 0 0	Tan	D2 B4 8C	210 180 140
FireBrick	B2 22 22	178 34 34	RosyBrown	BC 8F 8F	188 143 143
DarkRed	8B 00 00	139 0 0	SandyBrown	F4 A4 60	244 164 96
Colores Rosas			Goldenrod	DA A5 20	218 165 32
Pink	FF C0 CB	255 192 203	DarkGoldenrod	B8 86 0B	184 134 11
LightPink	FF B6 C1	255 182 193	Peru	CD 85 3F	205 133 63
HotPink	FF 69 B4	255 105 180	Chocolate	D2 69 1E	210 105 30
DeepPink	FF 14 93	255 20 147	SaddleBrown	8B 45 13	139 69 19
MediumVioletRed	C7 15 85	199 21 133	Sienna	A0 52 2D	160 82 45
PaleVioletRed	DB 70 93	219 112 147	Brown	A5 2A 2A	165 42 42
Colores Naranjas			Maroon	80 00 00	128 0 0
LightSalmon	FF A0 7A	255 160 122	Colores Blancos		
			White	FF FF FF	255 255 255
			Snow	FF FA FA	255 250 250
			Honeydew	F0 FF F0	240 255 240
			MintCream	F5 FF FA	245 255 250
			Azure	F0 FF FF	240 255 255
			AliceBlue	F0 F8 FF	240 248 255
			GhostWhite	F8 F8 FF	248 248 255
			WhiteSmoke	F5 F5 F5	245 245 245
			Seashell	FF F5 EE	255 245 238
			Beige	F5 F5 DC	245 245 220
			OldLace	FD F5 E6	253 245 230
			FloralWhite	FF FA F0	255 250 240
			Ivory	FF FF F0	255 255 240
			AntiqueWhite	FA EB D7	250 235 215

Coral	FF 7F 50	255 127 80
Tomato	FF 63 47	255 99 71
OrangeRed	FF 45 00	255 69 0
DarkOrange	FF 8C 00	255 140 0
Orange	FF A5 00	255 165 0

Colores Amarillos

Gold	FF D7 00	255 215 0
Yellow	FF FF 00	255 255 0
LightYellow	FF FF E0	255 255 224
LemonChiffon	FF FA CD	255 250 205
LightGoldenrodYellow	FA FA D2	250 250 210
PapayaWhip	FF EF D5	255 239 213
Moccasin	FF E4 B5	255 228 181
PeachPuff	FF DA B9	255 218 185
PaleGoldenrod	EE E8 AA	238 232 170
Khaki	F0 E6 8C	240 230 140
DarkKhaki	BD B7 6B	189 183 107

Colores Púrpuras

Lavender	E6 E6 FA	230 230 250
Thistle	D8 BF D8	216 191 216
Plum	DD A0 DD	221 160 221
Violet	EE 82 EE	238 130 238
Orchid	DA 70 D6	218 112 214
Fuchsia	FC 0F	252 15

Linen	FA F0 E6	250 240 230
LavenderBlush	FF F0 F5	255 240 245
MistyRose	FF E4 E1	255 228 225

Colores Grises

Gainsboro	DC DC DC	220 220 220
LightGrey	D3 D3 D3	211 211 211
Silver	C0 C0 C0	192 192 192
DarkGray	A9 A9 A9	169 169 169
Gray	80 80 80	128 128 128
DimGray	69 69 69	105 105 105
LightSlateGray	77 88 99	119 136 153
SlateGray	70 80 90	112 128 144
DarkSlateGray	2F 4F 4F	47 79 79
Black	00 00 00	0 0 0

nombre HTML	código hex R G B	código decimal R G B
-------------	---------------------	-------------------------

Colores Verdes

GreenYellow	AD FF 2F	173 255 47
Chartreuse	7F FF 00	127 255 0
LawnGreen	7C FC 00	124 252 0
Lime	00 FF 00	0 255 0
LimeGreen	32 CD 32	50 205 50
PaleGreen	98 FB 98	152 251 152
LightGreen	90 EE 90	144 238 144
MediumSpringGreen	00 FA 9A	0 250 154
SpringGreen	00 FF 7F	0 255 127
MediumSeaGreen	3C B3 71	60 179 113
SeaGreen	2E 8B 57	46 139 87
ForestGreen	22 8B 22	34 139 34
Green	00 80 00	0 128 0
DarkGreen	00 64 00	0 100 0
YellowGreen	9A CD 32	154 205 50
OliveDrab	6B 8E 23	107 142 35
Olive	80 80 00	128 128 0
DarkOliveGreen	55 6B 2F	85 107 47
MediumAquamarine	66 CD AA	102 205 170
DarkSeaGreen	8F BC 8F	143 188 143
LightSeaGreen	20 B2 AA	32 178 170
DarkCyan	00 8B 8B	0 139 139

	C0	192
Magenta	FF 00 FF	255 0 255
MediumOrchid	BA 55 D3	186 85 211
MediumPurple	93 70 DB	147 112 219
BlueViolet	8A 2B E2	138 43 226
DarkViolet	94 00 D3	148 0 211
DarkOrchid	99 32 CC	153 50 204
DarkMagenta	8B 00 8B	139 0 139
Purple	80 00 80	128 0 128
Indigo	4B 00 82	75 0 130
SlateBlue	6A 5A CD	106 90 205
DarkSlateBlue	48 3D 8B	72 61 139

Teal	00 80 80	0 128 128
Colores Azules		
Aqua/Cyan	00 FF FF	0 255 255
LightCyan	E0 FF FF	224 255 255
PaleTurquoise	AF EE EE	175 238 238
Aquamarine	7F FF D4	127 255 212
Turquoise	40 E0 D0	64 224 208
MediumTurquoise	48 D1 CC	72 209 204
DarkTurquoise	00 CE D1	0 206 209
CadetBlue	5F 9E A0	95 158 160
SteelBlue	46 82 B4	70 130 180
LightSteelBlue	B0 C4 DE	176 196 222
PowderBlue	B0 E0 E6	176 224 230
LightBlue	AD D8 E6	173 216 230
SkyBlue	87 CE EB	135 206 235
LightSkyBlue	87 CE FA	135 206 250
DeepSkyBlue	00 BF FF	0 191 255
DodgerBlue	1E 90 FF	30 144 255
CornflowerBlue	64 95 ED	100 149 237
MediumSlateBlue	7B 68 EE	123 104 238
RoyalBlue	41 69 E1	65 105 225
Blue	00 00 FF	0 0 255
MediumBlue	00 00 CD	0 0 205
DarkBlue	00 00 8B	0 0 139
Navy	00 00 80	0 0 128
MidnightBlue	19 19 70	25 25 112



Anexo IV

Sitio Web más Destacados

Anexo IV

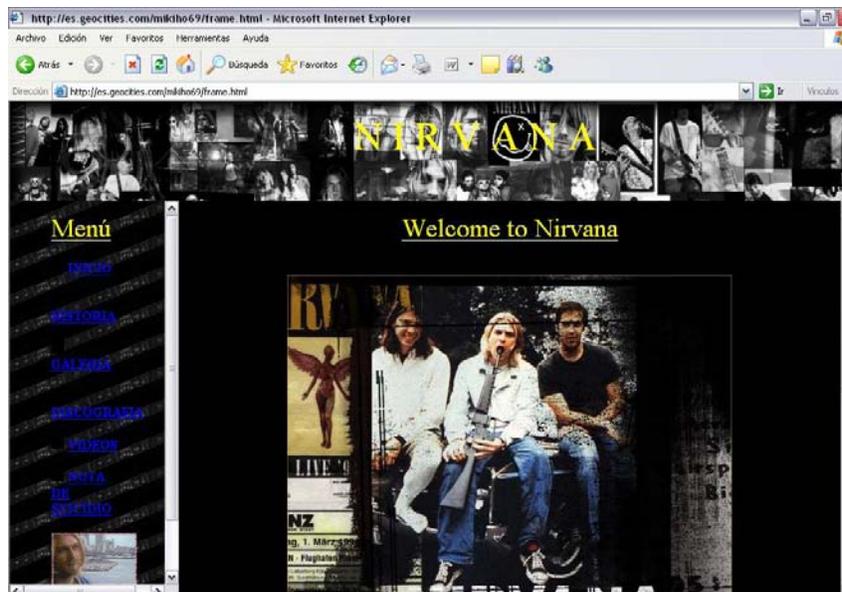
Sitio Web más Destacados.

Carlos Lucas López Figura AIV.1



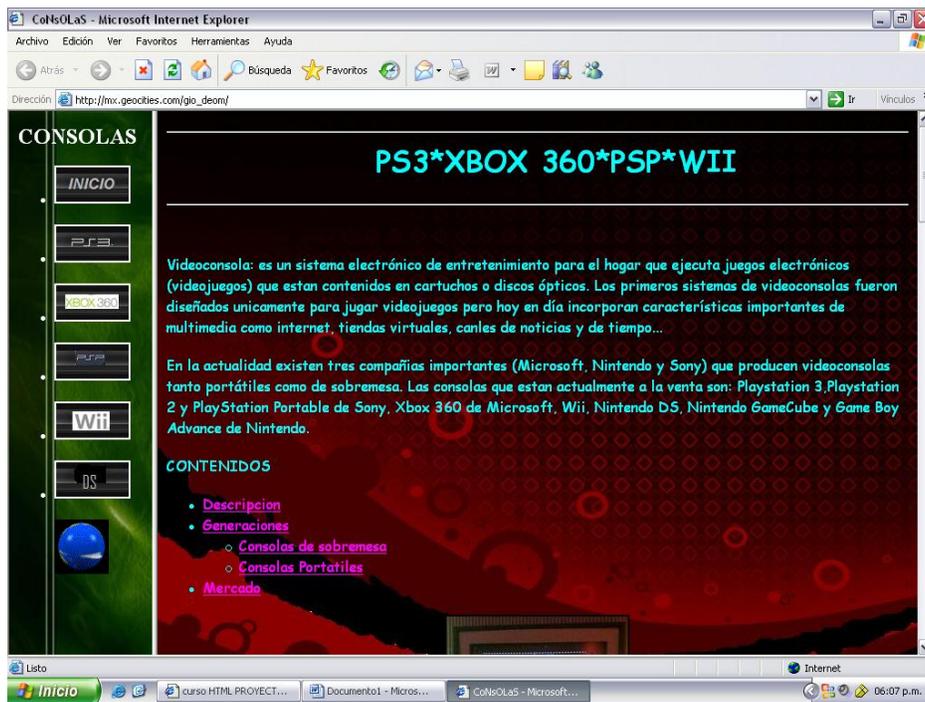
AIV.1 Sitio de Música

Miguel Ángel Medina Morales Figura IV.2



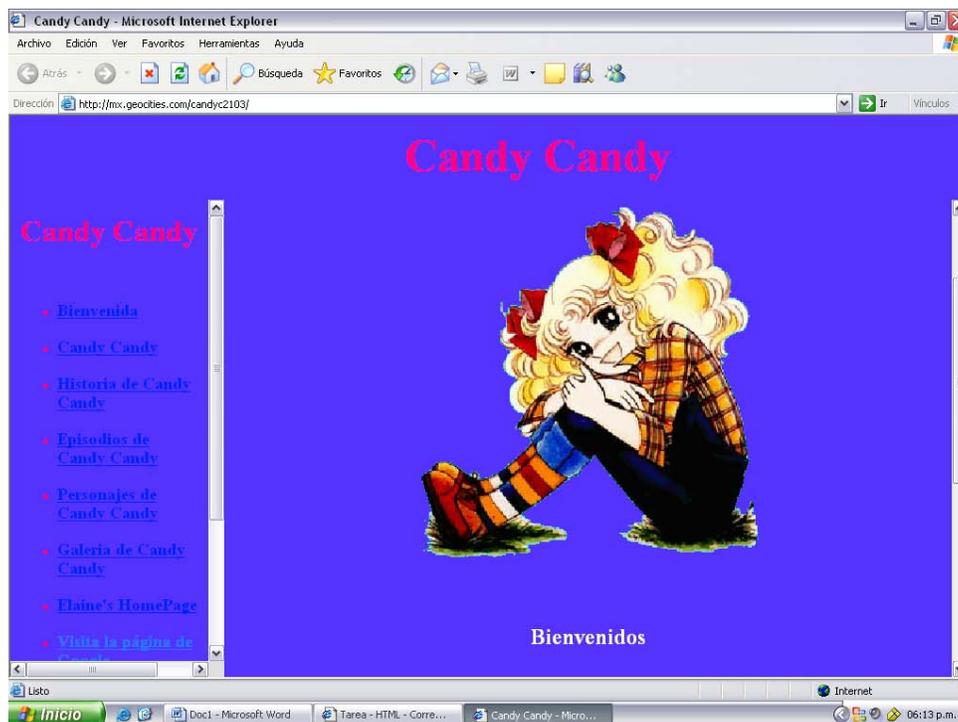
IV.2 Sitio de Grupo Musical Nirvana

✚ Giovanni Vázquez Domínguez Figura AIV.3



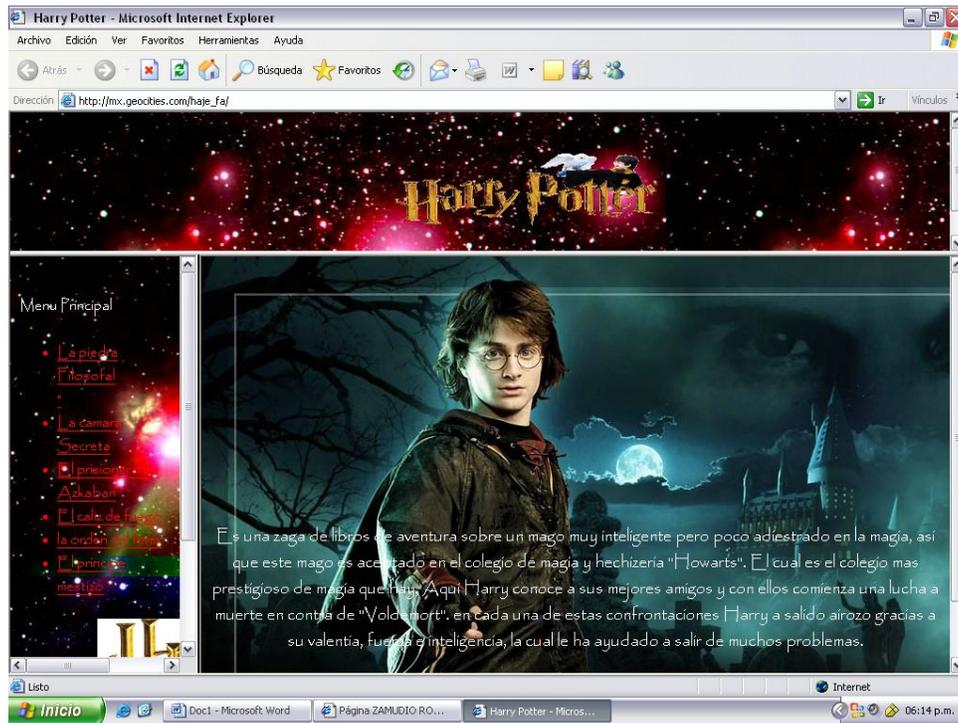
AIV.3 Sitio de Consolas de Juegos de Video

✚ Gabriela Hidai Huerta García Figura AIV.4



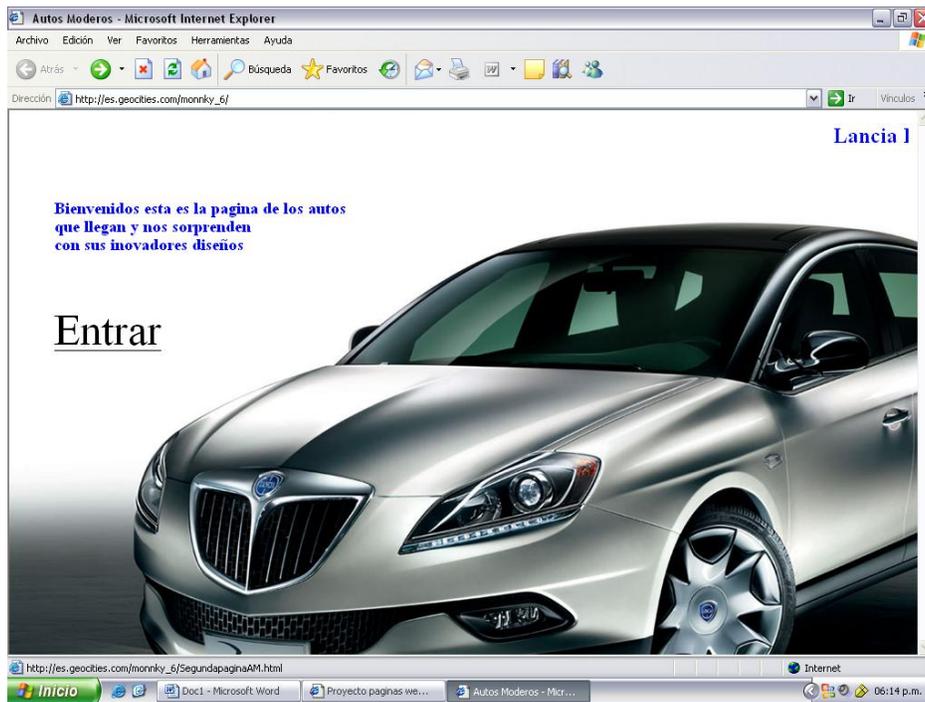
AIV.4 Sitio de Anime Candy Candy

✚ Jezabel Rojas Zamudio Figura AIV.5



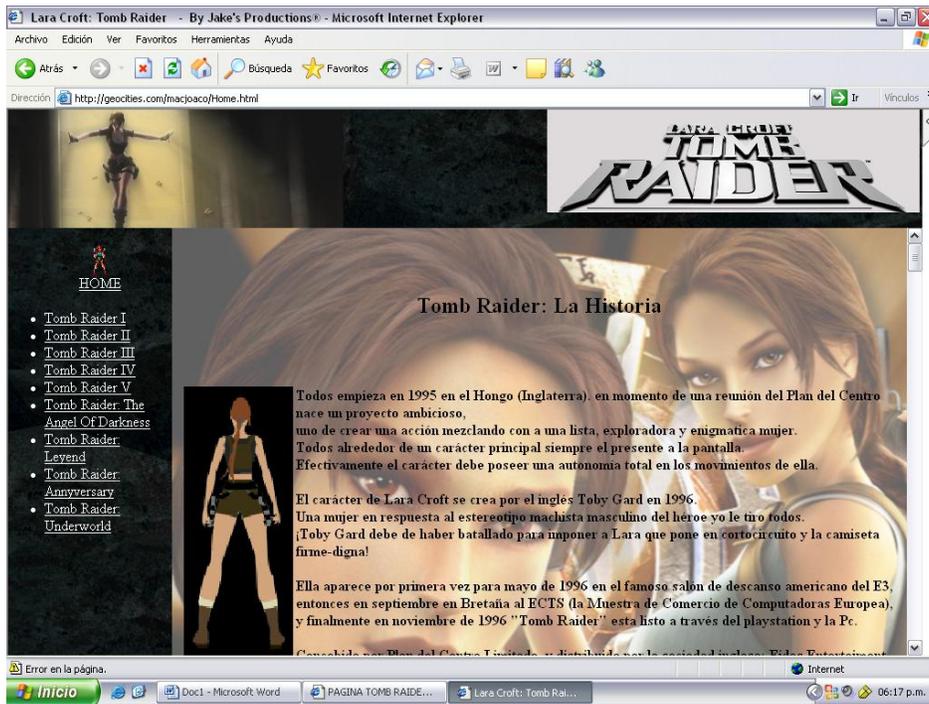
AIV.5 Sitio de Harry Potter

✚ Sebastián Román Acosta Figura AIV.6.



AIV.6 Sitio de Automóviles

✚ Joaquín Antonio Vallejo Ortiz Figura AIV.7.



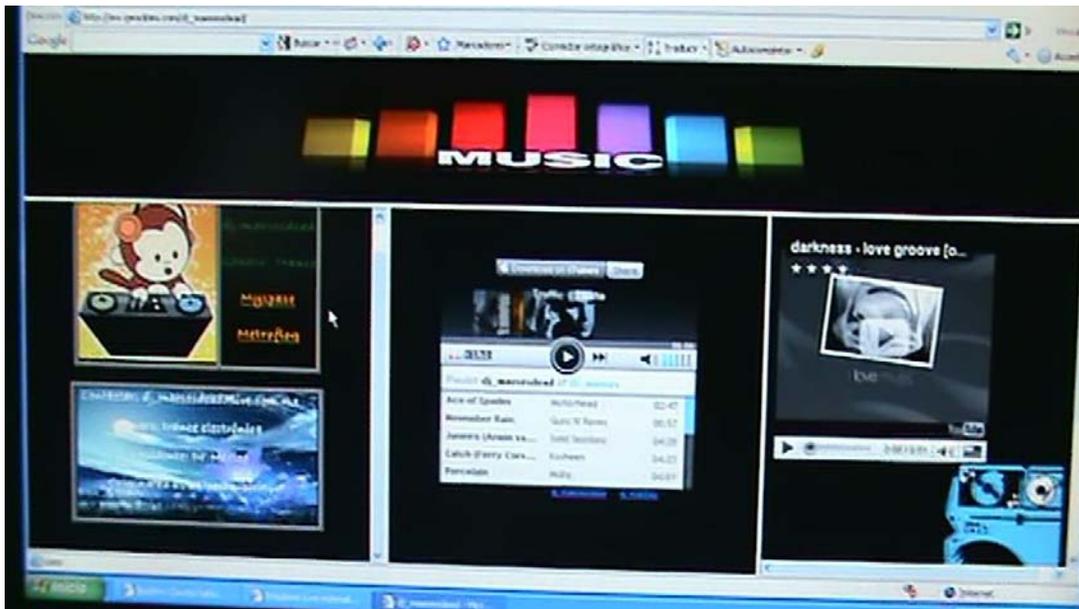
AIV.7 Sitio de Tome Raider

✚ Roberto Gómez Chavarría Figura AIV.8.



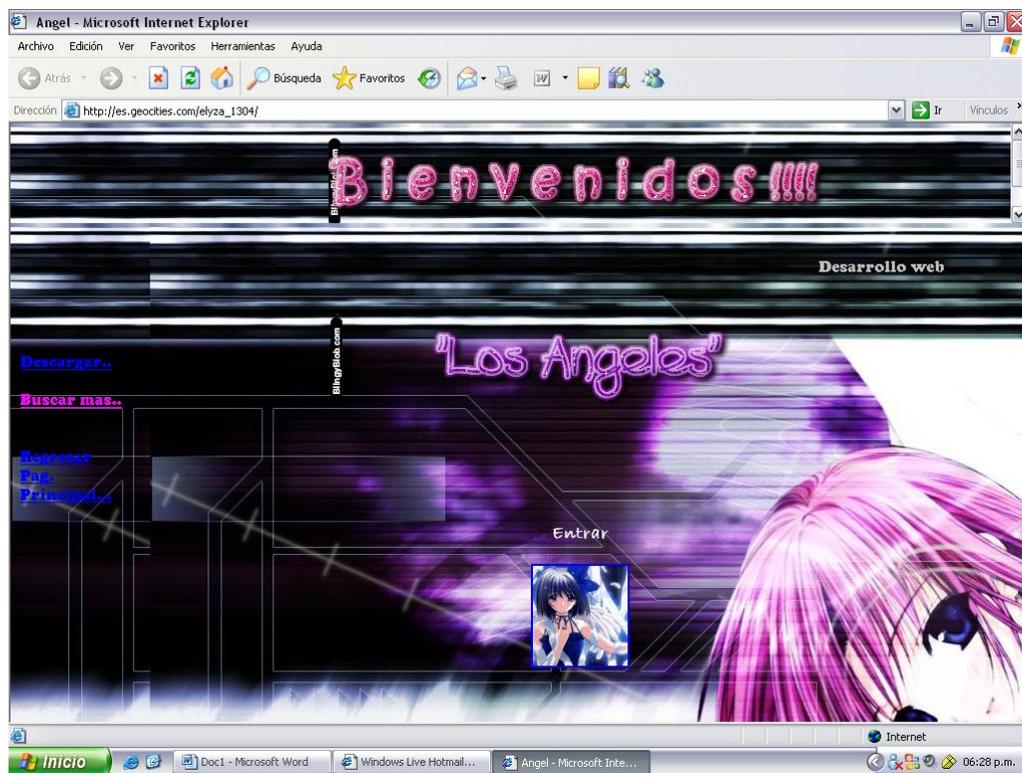
AIV.8 Sitio de One Piece

✚ Luis Miguel Martínez Figura AIV.9.



AIV.9 Sitio de Música DJ.

✚ Elizabeth Díaz Melgarejo Figura AIV.10



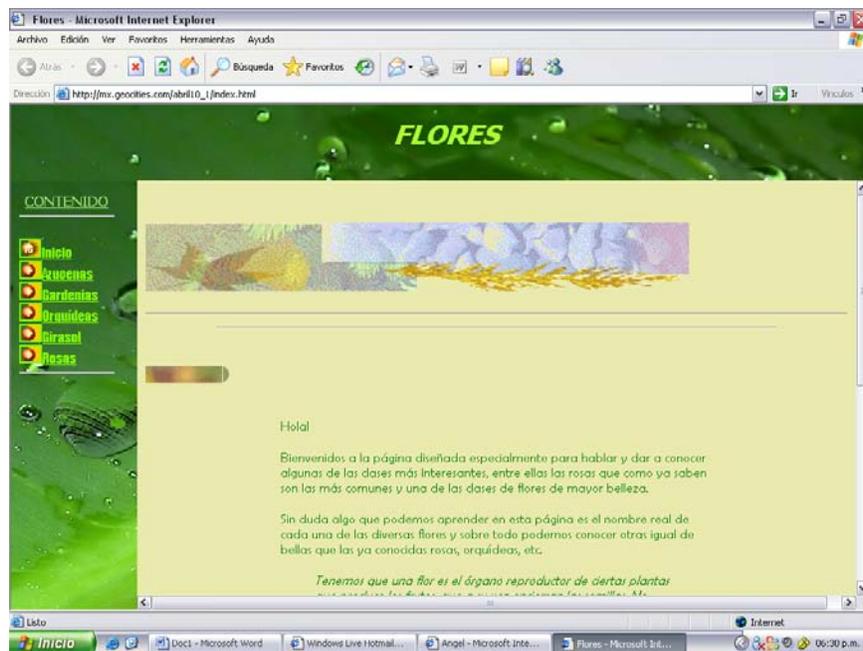
AIV.10 Sitio de Ángeles

Geovanni Alonso Ruiz Díaz Figura AIV.11



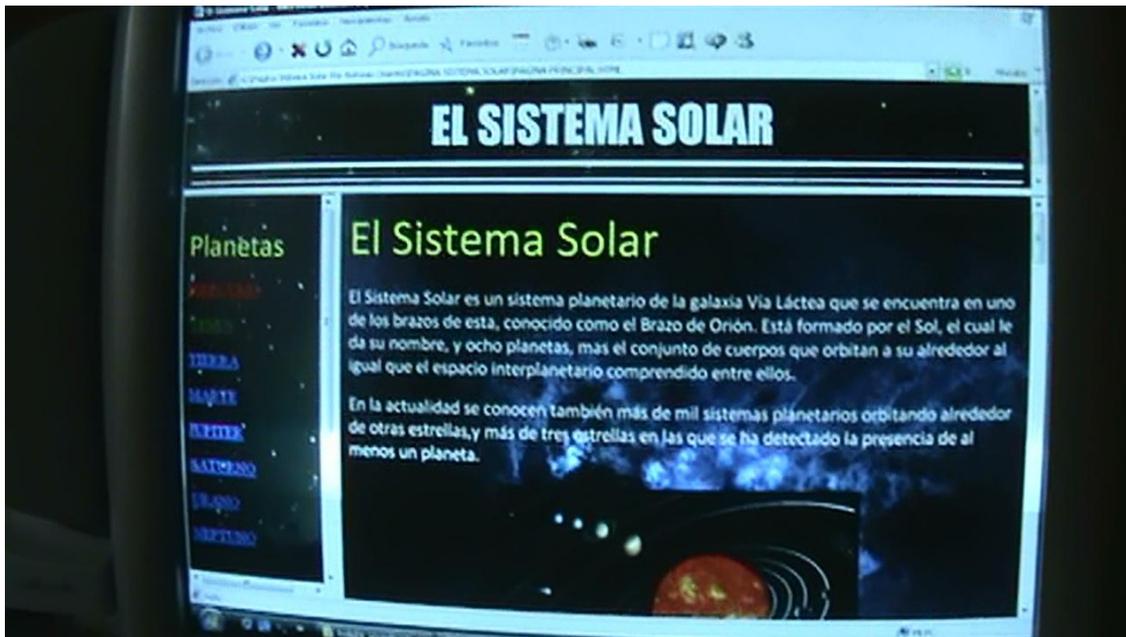
AIV.11 Sitio de Vacaciones en el Cervantino

Teresa Reyes Hernández Figura AIV.12



AIV.12 Sitio de Flores

Antonio Orienta Maya Figura AIV.13



AIV.13 Sitio del Sistema Solar.

Desarrollo Web Figura AIV.14



AIV.14 Sitio Oficial del Curso de Desarrollo Web.

Bibliografía

- ✚ Günter Born, Compendium HTML, Ediciones Marcombo, Páginas 557-622

Webgrafía.

- ✚ Página Oficial del Consorcio de W3, Normas de HTML, Disponible en: <http://www.w3.org/TR/REC-html40>, Fecha de Consulta: 12/05/2009
- ✚ Página Especializada en Manuales de Programación, Disponible en: <http://www.programacion.net/html/>, Fecha de Consulta: 8/06/2009
- ✚ Página Oficial del Consorcio de W3, Normas de CSS1, Disponible en: <http://www.w3.org/TR/REC-CSS1>, Fecha de Consulta: 15/06/2009
- ✚ Página Oficial del Consorcio de W3, Normas de CSS2, Disponible en: <http://www.w3.org/TR/REC-CSS2>, Fecha de Consulta: 18/06/2009
- ✚ Página Oficial del Consorcio de W3, Normas de Estilo, Disponible en: <http://www.w3.org/pub/WWW/Style>, Fecha de Consulta: 20/06/2009
- ✚ Página de Normas Especializadas en Web, Disponice en: <http://www.ecma.ch/stand/ecma-262.htm>, Fecha de Consulta: 22/06/2009
- ✚ Página de Manuales de JavaScript, Disponible en: <http://developer.netscape.com/tech/javascript/index.html>, Fecha de Consulta: 29/06/2009
- ✚ Manual de DHTML con ejercicios practicos, Disponible en: <http://developer.netscape.com/tech/dynhtml/index.html>, Fecha de Consulta: 2/07/2009
- ✚ Sitio Oficial de Reglas y Manulaes de PHP, Disponible en: <http://www.php.net/>, Fecha de Consulta: 15/07/2009

Enlaces Sugeridos.

- ✚ Sitio Oficial de Manuales de Web, Disponibles en: <http://developer.netscape.com/docs/manuals/communicator/jsguide4/index.htm>, Fecha de Consulta: 10/06/2009
- ✚ Sitio de Descargas de Manuales de HTML, Disponible en: <http://wsabstract.com/>, Fecha de Consulta: 18/06/2009
- ✚ Sitio de Reglas para Construir Páginas de HTML, con las normas Exactas, Disponible en: <http://www.w3.org/TR/REC-html40>, Fecha de Consulta: 20/06/2009
- ✚ Sitio de Reglas para Construir Páginas de STYLE, con las normas Exactas, Disponible en: <http://www.w3.org/pub/WWW/Style>, Fecha de Consulta: 22/06/2009

- ✚ Sitio de Reglas para Construir Páginas con PHP, con las normas Exactas, Disponible en: <http://www.php.net/>, Fecha de Consulta: 24/06/2009
- ✚ Sitio donde se pueden descargar manuales de JavaScript, Disponible en: <http://developer.netscape.com/tech/javascript/index.html>, Fecha de Consulta: 24/06/2009

Sitios de Alumnos Sobresalientes.

- ✚ Sitio de Giovanni Adrian Vázquez Domínguez, Disponible en: http://mx.geocities.com/gio_deom/ Fecha de Consulta: 25/06/2009
- ✚ Sitio de Lucas López Carlos, Disponible en: http://www.gratisweb.com/psy_ohm_trance , Fecha de Consulta: 25/06/2009
- ✚ Sitio de Gabriela Hida Huerta García, Disponible en: <http://mx.geocities.com/candyc2103>, Fecha de Consulta 25/06/2009
- ✚ Sitio de Jezabel Zamudio Rojas, Disponible en: http://mx.geocities.com/haje_fa/, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Román Acosta Sebastián, Disponible en: http://es.geocities.com/monnky_6/, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Joaquín Antonio Vallejo Ortiz, Disponible en: <http://geocities.com/macjoaco>, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Roberto Gómez Chavarría, Disponible en: http://es.geocities.com/monkeyd_09/, Fecha de Consulta 25/06/2009
- ✚ Sitio de Geovanni Alonso Ruíz Díaz, Disponible en: http://mx.geocities.com/jovas_2102/, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Elizabeth Díaz Melgarejo, Disponible en: http://es.geocities.com/elyza_1304, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Teresa Reyes Hernández, Disponible en: http://mx.geocities.com/abril10_1/index.html, Fecha de Consulta: 25/06/2009
- ✚ Sitio de Desarrollo Web, Disponible en: http://mx.geocities.com/wizard_pag/, Fecha de Consulta: 25/06/2009