



## Introducción a la programación web

Roberto Gómez Cárdenas  
<http://homepage.cem.itesm.mx/rogomez>  
rogomez@itesm.mx

Lámina 1 Dr. Roberto Gómez Cárdenas



## La programación web

- La programación web permite la creación de sitios dinámicos en Internet.
- Objetivo: poner al alcance de un usuario el contenido de una base de datos mediante lenguajes de script como pueden ser PHP, ASP o ASP.NET.
- Dominando la programación web podrá crear sitios dinámicos como periódicos digitales o tiendas virtuales.

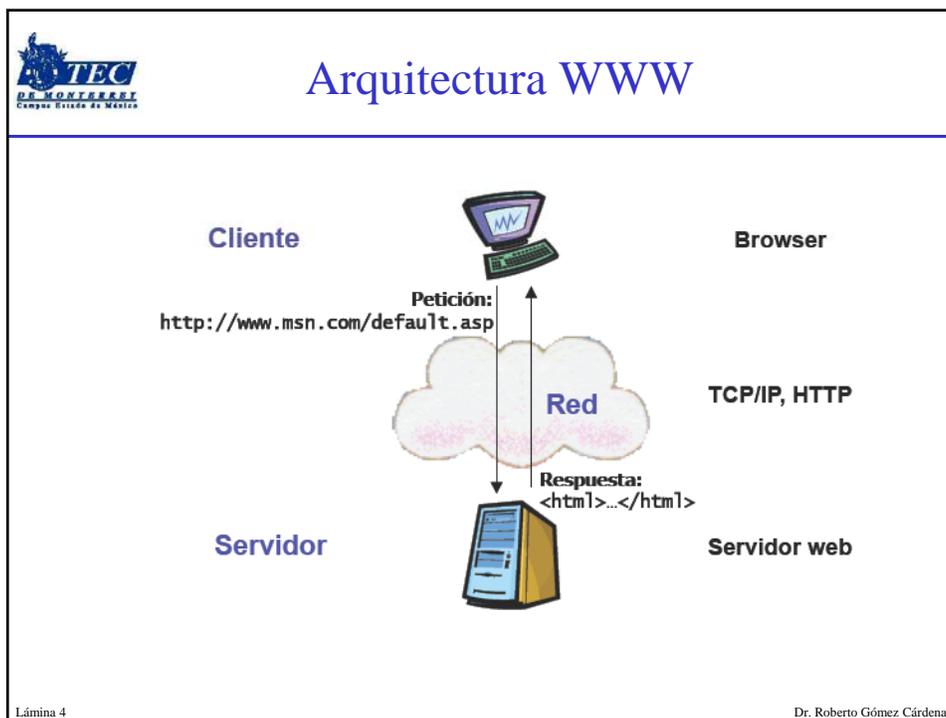


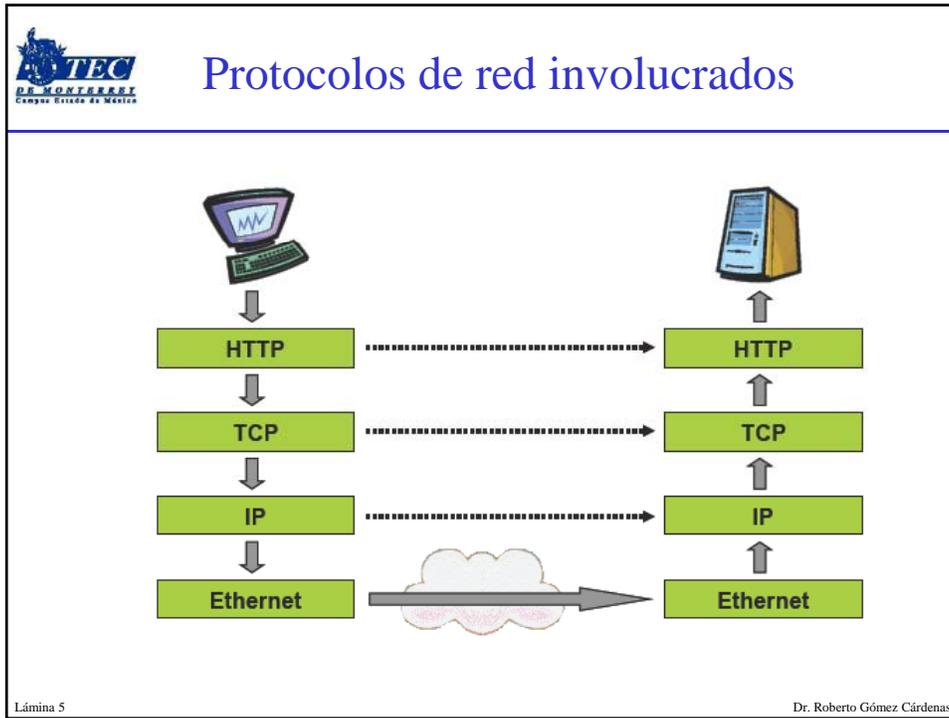
Lámina 2 Dr. Roberto Gómez Cárdenas

 **Principios diseño en la web**

- Universal
- Descentralizada
- Modular
- Extensible
- Escalable
- Accesible
- Versiones compatibles

Lámina 3 Dr. Roberto Gómez Cárdenas





**El protocolo HTTP**

Características principales

Lámina 6 Dr. Roberto Gómez Cárdenas



## Introducción

- Hypertext Transfer Protocol.
- Propuesto por Tim Berners-Lee en 1989
- HTTP es el protocolo utilizado para transferir los archivos (llamados recursos) que forman parte de la World Wide Web. Ya sean estos archivos HTML, imágenes, sonidos, etc...
- Conexión TCP puerto 80 que escucha pasivamente.
- Actualmente versión HTTP/1.1
  - HTTP/1.0 ver RFC 1945 y HTTP/1.1 ver RFC 2616
- Referencia: <http://www.w3.org/protocols>

Lámina 7

Dr. Roberto Gómez Cárdenas



## ¿Qué son los recursos?

- HTTP se utiliza para transferir *recursos*, no solo archivos.
- Un recurso es un trozo de información que puede identificarse a través de un URL.
- La clase más común de recursos son los archivos, pero también pueden ser datos generados dinámicamente.

Lámina 8

Dr. Roberto Gómez Cárdenas



## ¿Qué es un URL?

- Uniform Resource Locator.
- Secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, para su localización.
- Usados por primera vez por Tim Berners-Lee en 1991, para permitir a los autores de documentos establecer hiperenlaces en la World Wide Web.
  - Desde 1994, en los estándares de la Internet, el concepto de URL ha sido incorporado dentro del más general de URI (*uniform resource identifier*).
  - Ver: <http://www.bernzilla.com/item.php?id=100>

Lámina 9

Dr. Roberto Gómez Cárdenas



## Formato URL

- El formato general de un URL es  
esquema://máquina/directorio/archivo
- Es posible añadir otros datos  
esquema://usuario:contraseña@máquina:puerto/directorio/archivo
- Ejemplo  
<http://www.cem.itesm.mx/>
- La especificación se encuentra en el RFC 1738.

Lámina 10

Dr. Roberto Gómez Cárdenas



## Esquema URL

- Un URL se clasifica por su esquema, que generalmente indica el protocolo de red que se usa para recuperar, a través de la red, la información del recurso identificado.
- Un URL comienza con el nombre de su esquema, seguido por dos puntos, y por una parte específica del esquema.
  - http - recursos HTTP
  - https - HTTP sobre SSL
  - ftp - File Transfer Protocol
  - mailto - direcciones E-mail
  - ldap - búsquedas LDAP Lightweight Directory Access Protocol
  - file - recursos disponibles en el sistema local, o en una red local
  - news - grupos de noticias Usenet (newsgroup)

Lámina 11

Dr. Roberto Gómez Cárdenas



## Referencias URI

- URI: Universal Resource Identifier
- El término referencia URI se refiere a un caso particular de un URI, o una porción de éste, tal como es usada en un documento HTML.
- Las referencias URI introducen dos nuevos conceptos: la distinción entre referencias *absolutas* y *relativas*, y el concepto de un identificador de fragmento.

Lámina 12

Dr. Roberto Gómez Cárdenas



## Referencias URI absolutas y relativas

- Un *URL absoluto* es una referencia URI que es parecida a los URL definidos arriba; empieza por un esquema seguido de dos puntos (":") y de una parte específica del esquema.
- Un *URL relativo* es una referencia URI que comprende sólo la parte específica del esquema de un URL, o de algún componente de seguimiento de aquella parte.
- El esquema y componentes principales se infieren del contexto en el cual aparece la referencia URL: el URI base (o URL base) del documento que contiene la referencia.

Lámina 13

Dr. Roberto Gómez Cárdenas



## Ejemplos

- Ejemplos de URL absolutos:
  - `http://acme.org/w/pagina.phtml?title=URL&action=history`
  - `http://acme.org/wiki/URL#Esquemas_en_URL`
- Ejemplos de URL relativos:
  - `//acme.org/contabilidad/Facturas_Agosto`
  - `/toto/URL`
  - `URL#Referencias_URI`

Lámina 14

Dr. Roberto Gómez Cárdenas



## Estructura de las transacciones HTTP

- HTTP utiliza el modelo cliente/servidor.
- Un cliente HTTP abre una conexión hacia un servidor HTTP y envía un mensaje de petición (request message), luego el servidor envía un mensaje de respuesta (response message) el cual contiene el recurso que solicitado.
- Luego de enviar la respuesta al servidor cierra la conexión.
- Por ende el protocolo no mantiene el estado (*stateless*) entre las distintas transacciones de un mismo cliente.

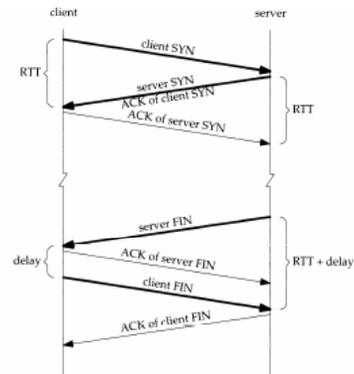


Lámina 15

Dr. Roberto Gómez Cárdenas



## Mensajes HTTP

- Los mensajes HTTP pueden ser:
  - Solicitudes/Peticiones
  - Respuestas
- Tanto las solicitudes como las repuestas utilizan el formato genérico de e-mails (RFC-822).
- Ambos tipos de mensajes consisten de
  - Una línea inicial.
  - Cero o más encabezados (headers).
  - Una línea en blanco.
  - Un cuerpo del mensaje (opcional, ej. Archivo, datos de una consulta).

Lámina 16

Dr. Roberto Gómez Cárdenas



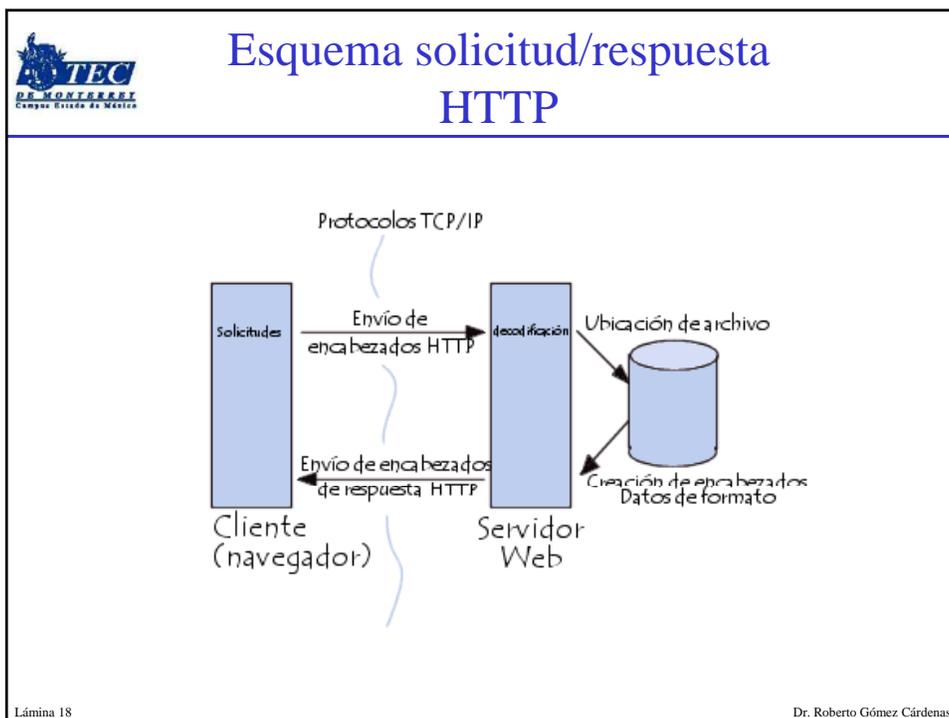
## Formato mensaje HTTP

---

- El formato de un mensaje HTTP es
  - < línea inicial, distinta para solicitudes y respuestas >
  - Encabezado1: valor1
  - Encabezado2: valor2
  - ...<más encabezados>...
  - Encabezado N: valor N
  - < línea en blanco >
  - < cuerpo de mensaje opcional, contenidos de un archivo, de una consulta, datos binarios, etc >

Lámina 17

Dr. Roberto Gómez Cárdenas





## La línea inicial (solicitud)

- La línea inicial de una solicitud tiene tres partes separadas entre sí por un espacio.
  - El método.
  - El identificador del recurso.
  - La versión del protocolo HTTP en uso.

METODO

IDENTIFICADOR RECURSO

VERSION HTTP

- Ejemplos
 

GET /directorio1/directorio2/index.html HTTP/1.0

GET / HTTP/1.1

Lámina 19 Dr. Roberto Gómez Cárdenas



## Métodos solicitudes

Método	Descripción
GET	Solicita el recurso ubicado en la URL especificada. Transmite información a través de la URI agregando parámetros a la URL.
HEAD	Igual que el método GET, salvo que el servidor solo devuelve las cabeceras, no el contenido.
POST	Envía datos para ser procesados por un programa ubicado en la URI especificada. Los datos se incluirán en el cuerpo de la petición.
PUT	Permite guardar el contenido de la petición en el servidor bajo la URI especificada. Si ya existe lo reemplaza, sino lo crea.
DELETE	Borra el recurso ubicado en la URL especificada. No se garantiza al cliente que la operación se lleve a cabo, aunque la respuesta sea satisfactoria.
TRACE	Este método solicita al servidor que envíe de vuelta en un mensaje de respuesta. Se utiliza con fines de comprobación y diagnóstico.
OPTIONS	Devuelve los métodos HTTP que el servidor soporta para un URL específico.
CONNECT	Utilizado para llevar a cabo conexiones. Podría ser usado para utilizar un site como proxy.

Lámina 20 Dr. Roberto Gómez Cárdenas



## La línea inicial (respuestas)

- La línea inicial de una respuesta (llamada línea de estado) tiene tres partes separadas entre sí por un espacio.
  - Versión HTTP.
  - Código de estado.
  - Fase explicativa (legible por humanos).

VERSION HTTP	CODIGO ESTADO	EXPLICACION
--------------	---------------	-------------

- Ejemplos.
 

HTTP/1.0	200	OK
HTTP/1.0	404	Not Found

Lámina 21 Dr. Roberto Gómez Cárdenas



## Códigos de estado

- El código de estado es un entero de tres dígitos
  - 1xx: Informativos
  - 2xx: Éxito
  - 3xx: Redirección
  - 4xx: Error de cliente
  - 5xx: Error de servidor
- Los más comunes:
  - 200 OK
    - Solicitud exitosa, la respuesta se envía en el cuerpo.
  - 404 Not Found
    - El recurso no existe.
  - 303 See Other
    - El recurso se ha movido a otra URL (Dada en el header Location)
  - 500 Server Error
    - Error no esperado en el servidor.

Lámina 22 Dr. Roberto Gómez Cárdenas



## Códigos de estado

Código (s)	Significado
100111	Conexión rechazada
200	OK
201-203	Información no oficial
204	Sin contenido
205	Contenido para recargar
206	Contenido parcial
301	Mudado permanentemente
302	Encontrado
303	Vea otros
304	No modificado
305	Utilice un proxy
307	Redirección temporal

Lámina 23 Dr. Roberto Gómez Cárdenas



## Continuación códigos estados

Código	Descripción
400	Solicitud incorrecta
402	Pago requerido
403	Prohibido
404	No encontrado
409	Conflicto
410	Ya no disponible
412	Falló precondition
500	Error interno
501	No implementado
502	Pasarela incorrecta
503	Servicio nodisponible
504	Tiempo de espera de la pasarela agotado
505	Versión de HTTP no soportada

Lámina 24 Dr. Roberto Gómez Cárdenas



## Líneas de encabezado

- Estas líneas proporcionan información acerca de la solicitud o respuesta.
- Cada línea de encabezado consiste de un nombre de campo seguido por un carácter dos puntos “:” y el valor para ese campo.
- El orden de los campos no es importante.
- Ejemplos:

```
User-Agent: Mozilla/6.0  
From: juan@perez.com  
Content-Type: text/html
```

Lámina 25

Dr. Roberto Gómez Cárdenas



## Líneas de encabezado

- HTTP 1.0 define 16 headers (ninguno es obligatorio).
- HTTP 1.1 define 46 headers (solo Host es obligatorio).
- En las solicitudes suelen incluirse los siguientes:
  - User-Agent: (Identifica al software del cliente y la versión).
  - From: (La dirección de e-mail de quien envía la solicitud).
- En las respuestas algunos encabezados comunes son:
  - Server: (análogo a User-Agent:, ej. Server: Apache/1.3.14).
  - Last-Modified: (fecha de última modificación del recurso, se utiliza para mantener actualizados los cachés, ej. Last-Modified: Fri, 31 Jan 2000 12:12:12 GMT)

Lámina 26

Dr. Roberto Gómez Cárdenas



## Líneas encabezado solicitud

Nombre encabezado	Descripción
Accept	Tipo de contenido aceptado por el navegador (por ejemplo, <i>texto/html</i> )
Accept-Charset	Conjunto de caracteres que el navegador espera
Accept-Encoding	Codificación de datos que el navegador acepta
Accept-Language	Idioma que el navegador espera (de forma predeterminada, inglés)
Authorization	Identificación del navegador en el servidor.
Content-Encoding	Tipo de codificación para el cuerpo de la solicitud
Content-Language	Tipo de idioma en el cuerpo de la solicitud
Content-Type	Tipo de contenido del cuerpo de la solicitud (por ejemplo, <i>texto/html</i> ).
Date	Fecha en que comienza la transferencia de datos

Lámina 27

Dr. Roberto Gómez Cárdenas



## Líneas encabezado solicitud

Nombre encabezado	Descripción
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
From	Permite especificar la dirección de correo electrónico del cliente
From	Permite especificar que debe enviarse el documento si ha sido modificado desde una fecha en particular
Link	Vínculo entre dos direcciones URL
Orig-URL	Dirección URL donde se originó la solicitud
Referer	Dirección URL desde la cual se realizó la solicitud
User-Agent	Cadena con información sobre el cliente, por ejemplo, el nombre y la versión del navegador y el sistema operativo

Lámina 28

Dr. Roberto Gómez Cárdenas



## Líneas encabezado respuesta

Nombre encabezado	Descripción
Content-Encoding	Tipo de codificación para el cuerpo de la respuesta.
Content-Language	Tipo de idioma en el cuerpo de la respuesta
Content-Length	Extensión del cuerpo de la respuesta
Content-Type	Tipo de contenido del cuerpo de la respuesta
Date	Fecha en que comienza la transferencia de datos
Expires	Fecha límite de uso de los datos
Forwarded	Utilizado por equipos intermediarios entre el navegador y el servidor
Location	Re-direccionamiento a una nueva dirección URL asociada con el documento
Server	Características del servidor que envió la respuesta

Lámina 29 Dr. Roberto Gómez Cárdenas



## Contenidos MIME

- MIME (extensiones multipropósito de correo en Internet) es un estándar propuesto por los laboratorios *Bell Communications* en 1991 para ampliar las posibilidades del correo electrónico al incluir la posibilidad de insertar documentos (imágenes, sonidos y texto) en un mensaje.

Lámina 30 Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
application/acad	Archivos de AutoCAD	dwg
application/clariscad	Archivos de ClarisCAD	ccad
application/drafting	Archivos de bosquejo preliminar de MATRA	drw
application/dxf	Archivos de AutoCAD	dxf
application/i-deas	Archivos de SDRC I-deas	unv
application/iges	Formato de intercambio CAO IGES	igs, iges
application/octet-stream	Archivos binarios no interpretados	bin
application/oda	Archivos ODA	oda
application/pdf	Archivos Adobe Acrobat	pdf

Lámina 31

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
application/postscript	Archivos PostScript	ai, eps, ps
application/pro_eng	Archivos de ProEngineer	prt
application/rtf	Formato de texto enriquecido	rtf
application/set	Archivos CAO SET	set
application/sla	Archivos de estereolitografía	stl
application/solids	Archivos solids de MATRA	dwg
application/step	Archivos de datos STEP	step
application/vda	Archivos de superficie	vda
application/x-mif	Archivos de Framemaker	Mif
application/x-csh	Secuencia de comandos C-Shell (UNIX)	dwg
application/x-dvi	Archivos de texto dvi	dvi

Lámina 32

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
application/hdf	Archivos de datos	hdf
application/x-latex	Archivos de LaTeX	latex
application/x-netcdf	Archivos de NetCDF	nc, cdf
application/x-sh	Secuencia de comandos Bourne Shell	sh
application/x-tcl	Secuencia de comandos Tcl	tcl
application/x-tex	Archivos tex	tex
application/x-texinfo	Archivos eMacs	texinfo, texi
application/x-troff	Archivos Troff	t, tr, troff
application/x-troff-man	Archivos Troff/macro man	man
application/x-troff-me	Archivos Troff/macro ME	me

Lámina 33

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
application/x-troff-ms	Archivos Troff/macro MS	ms
application/x-wais-source	Fuente Wais	src
application/x-bcpio	CPIO binario	bcpio
application/x-cpio	CPIO Posix	cpio
application/x-gtar	Tar GNU	gtar
application/x-shar	Archivos Shell	shar
application/x-sv4cpio	CPIO SVR4n	sv4cpio
application/x-sv4crc	CPIO SVR4n con CRC	sv4crc
application/x-tar	Archivos tar comprimidos	tar
application/x-ustar	Archivos tar Posix comprimidos	ustar
application/zip	Archivos ZIP comprimidos	zip

Lámina 34

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
audio/basic	Archivos de audio básicos	au, snd
audio/x-aiff	Archivos de audio AIFF	aif, aiff, aifc
audio/x-wav	Archivos de audio Wave	wav
image/gif	Imágenes Gif	gif
image/ief	Imágenes con formato de intercambio	ief
image/jpeg	Imágenes jpeg	jpg, jpeg, jpe
image/tiff	Imágenes tiff	tiff, tif
image/x-cmu-raster	Ráster cmu	cmu
image/x-portable-anymap	Archivos Anymap PBM	pnm
image/x-portable-bitmap	Archivos de mapa de bits PBM	pbm

Lámina 35

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
image/x-portable-graymap	Archivos Graymap PBM	pgm
image/x-portable-pixmap	Archivos Pixmap PBM	ppm
image/x-rgb	Imágenes RGB	rgb
image/x-xbitmap	Imágenes X Bitmap	xbm
image/x-xpixmap	Imágenes X Pixmap	xpm
image/x-xwindowdump	Imágenes de volcado X Window	man
multipart/x-zip	Archivos Zip almacenados	zip
multipart/x-gzip	Archivos Zip GNU almacenados	gz, gzip
text/html	Archivos HTML	htm, html

Lámina 36

Dr. Roberto Gómez Cárdenas



## Lista de tipos MIME

Tipo MIME	Tipo de archivo	Extensión asociada
text/plain	Archivos de texto sin formato	txt, g, h, c, cc, hh, m, f90
text/richtext	Archivos de texto enriquecido	rtx
text/tab-separated-value	Archivos de texto con separación de valores	tsv
text/x-setext	Archivos de texto struct	etx
video/mpeg	Video MPEG	mpeg, mpg, mpe
video/quicktime	Videos de QuickTime	qt, mov
video/msvideo	Videos de Microsoft Windows	avi
video/x-sgi-movie	Videos de MoviePlayer	movie

Lámina 37 Dr. Roberto Gómez Cárdenas



## Configurando respuestas servidor

- Varias formas de configurar encabezados de respuesta del servidor.
- Una de las más sencillas: Hacer uso de las opciones del propio servidor, por ejemplo en Apache utilizar los `ServerTokens`
  - Si se asigna el `ServerTokens` el valor `Minimal` en `httpd.conf`, el servidor enviará solo `"Apache/x.xx.xx"` como campo `server` en las respuestas HTTP.
- En Apache se puede activar el `mod_headers` y configurar los encabezados.

Lámina 38 Dr. Roberto Gómez Cárdenas



## Cuerpo del mensaje

- Luego de las líneas de encabezado un mensaje HTTP puede contener un cuerpo (body).
- En las respuestas el cuerpo es la sección en donde se envía el recurso solicitado.
- En las solicitudes el cuerpo se utiliza para subir datos que ingresó el usuario o para transferir archivos hacia el servidor.
- Las líneas de encabezado más comunes que definen el cuerpo son:
  - Content-Type: (Da el tipo MIME de los datos del cuerpo, ejemplo: text/html image/gif).
  - Content-Lenght: (Especifica el número de bytes en el cuerpo).

Lámina 39

Dr. Roberto Gómez Cárdenas



## Ejemplo sesión HTTP

- El cliente desea acceder a
  - `http://www.ieee.org/numero/uno.html`
- El cliente establece una conexión TCP al puerto 80 de `www.ieee.org` y envía la solicitud con el método GET.

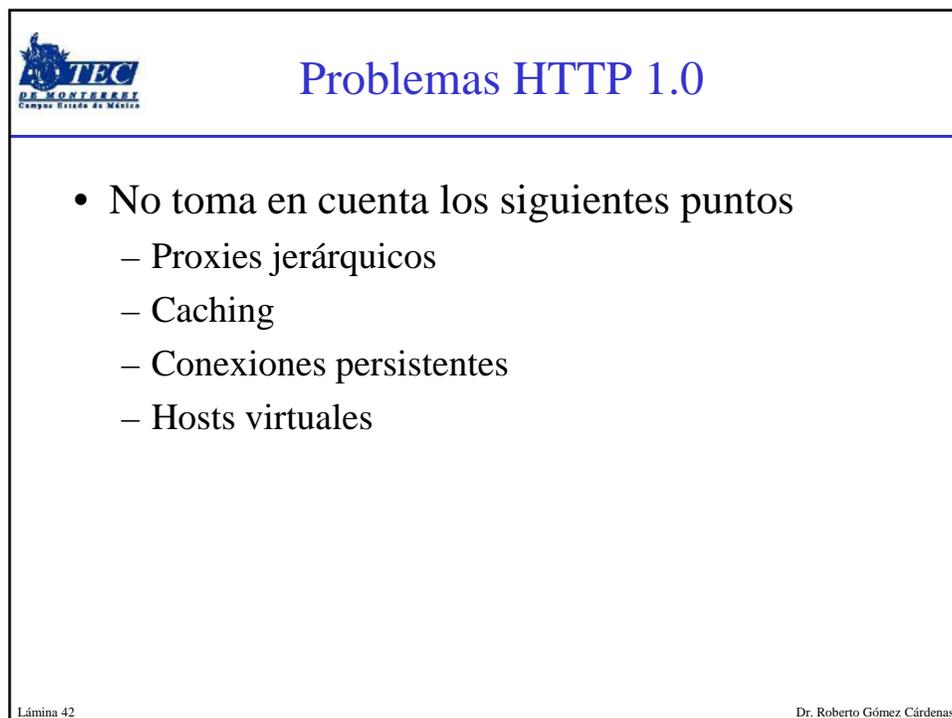
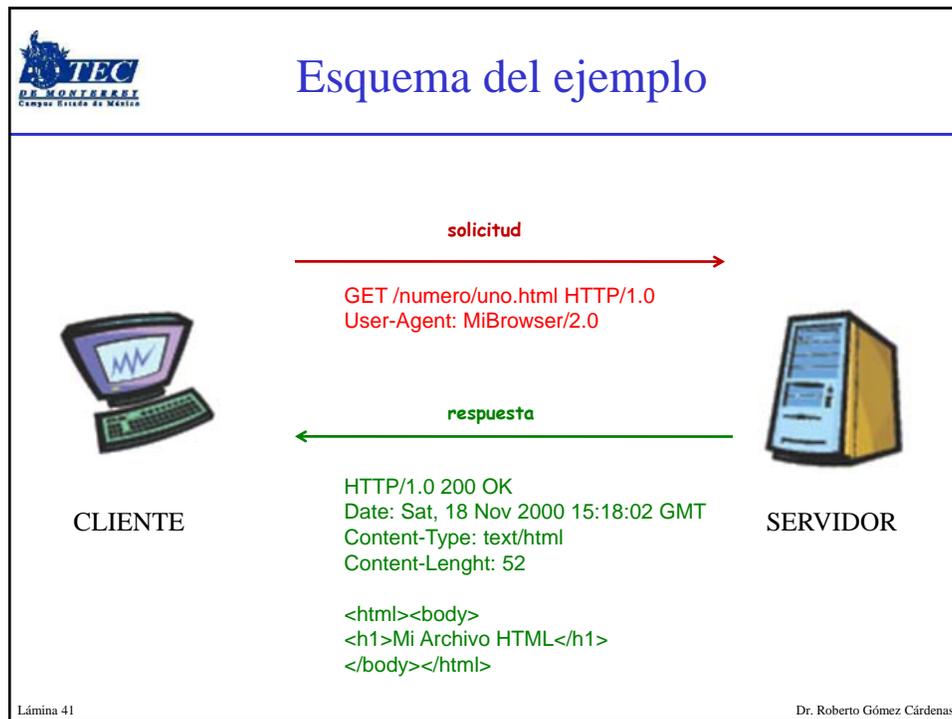
```
GET /numero/uno.html HTTP/1.0
User-Agent: MiBrowser/2.0
[línea en blanco]
```

- El servidor responde por la misma conexión con:

```
HTTP/1.0 200 OK
Date: Sat, 18 Nov 2000 15:18:02 GMT
Content-Type: text/html
Content-Lenght: 52
[línea en blanco]
<html><body>
<h1>Mi Archivo HTML</h1>
</body></html>
```

Lámina 40

Dr. Roberto Gómez Cárdenas





## HTTP 1.1

- Fue definido para atacar nuevas necesidades y solucionar problemas de HTTP 1.0. Las mejoras incluyen:
  - Respuesta más veloz (permite que en una sola conexión se realicen varias transacciones solicitud/respuesta).
  - Ahorro de ancho de banda a través del uso de caché.
  - Respuesta más rápida para páginas generadas automáticamente, permite que una respuesta se envíe aún cuando no se sepa su longitud total (chunked response).
  - Uso eficiente de las direcciones IP (permite servidores virtuales basados en nombres).

Lámina 43 Dr. Roberto Gómez Cárdenas



## HTTP 1.1 Encabezado “Host”

- A partir de HTTP 1.1 un servidor en una dirección IP puede manejar múltiples sitios webs virtuales.
  - *www.toto.com* y *www.cachafas.com* pueden residir en el mismo host
- Para que ello sea posible cada solicitud debe incluir el encabezado “Host”.
- Ejemplo: 

```
GET /directorio/archivo.html HTTP/1.1
Host: www.sitio.com
[línea en blanco]
```
- Host es el único encabezado obligatorio en una solicitud HTTP 1.1.

Lámina 44 Dr. Roberto Gómez Cárdenas



## Conexiones persistentes

- En HTTP 1.0,
  - Conexiones TCP se cierran luego de cada “request – response”.
  - Páginas WEB constan de varios recursos.
  - Abrir y cerrar conexiones consume tiempo y recursos como CPU, memoria y ancho de banda.
- En HTTP 1.1,
  - Se mantienen conexiones persistentes.
  - Se envían varios “requests” en serie sobre la misma conexión y se reciben los distintos recursos en el orden que fueron solicitados.

Lámina 45 Dr. Roberto Gómez Cárdenas



## Encabezado "Connection: close"

- El cliente lo incluye en la solicitud si considera que el recurso es el último.
- Si el servidor lo incluye en la respuesta, el cliente deberá abrir otra conexión para transferir más datos.
- Un cliente que no soporta conexiones persistentes debe incluir siempre el encabezado “Connection: close”.

Lámina 46 Dr. Roberto Gómez Cárdenas



## Encabezados para manejo de cache

- If-Modified-Since:
  - Solamente envía el recurso si ha sido modificado desde la fecha indicada.
- If-Unmodified-Since
  - Lo contrario.
  - Indica que solo se debe enviar el recurso solicitado si éste no ha cambiado luego de la fecha especificada.
- Respuesta

```
HTTP/1.1 304 Not Modified
Date: Fri, 31 Dec 1999 23:59:59 GMT
[blank line here]
```

Lámina 47 Dr. Roberto Gómez Cárdenas



## HTTP 1.1: Chunked Transfer-Encoding

- Si un servidor desea comenzar a enviar la respuesta antes de conocer su longitud total puede hacerlo incluyendo el encabezado Transfer-Encoding: chunked.
- El cuerpo de un mensaje con esta codificación contiene una serie de trozos (chunks) seguidos con una línea con un "0" (cero), seguido de una serie de footers (iguales a los headers).
- Cada trozo contiene dos partes:
  - Una línea con el tamaño de ese trozo (en hexadecimal).
  - Los datos en si mismos (al final se agrega CRLF).

Lámina 48 Dr. Roberto Gómez Cárdenas



## Ejemplo HTTP 1.1: Chunked Transfer-Encoding

```

HTTP/1.1 200 OK
Date: Sat, 18 Nov 2000 13:29:14 GMT
Content-Type: text/plain
Transfer-Encoding: chunked

25
This is the data in the first chunk

1C
and this is the second one

3
con

8
sequence

0
[línea en blanco]
```

"This is the data in the first chunk\r\n" (37 caracteres => hex: 0x25)

"and this is the second one\r\n" (28 caracteres => hex: 0x1C)

"con" (3 caracteres => hex: 0x03)

"sequence" (8 caracteres => hex: 0x08)

Lámina 49 Dr. Roberto Gómez Cárdenas



## Ejemplo HTTP 1.1: Chunked Transfer-Encoding

- El ejemplo anterior equivale a:

```

HTTP/1.1 200 OK
Date: Sat, 18 Nov 2000 13:29:14 GMT
Content-Type: text/plain
Content-Length: 76

This is the data in the first chunk
and this is the second one
consequence
```

Lámina 50 Dr. Roberto Gómez Cárdenas



## HTTP 1.1: El encabezado “Date”

- Para implementar cachés HTTP es necesario registrar las fechas y horas de creación/modificación de los recursos (timestamps).
- Para ello se incluye el encabezado “Date”. Los servidores deben incluir la fecha y hora actual utilizando este encabezado.
- Ejemplo:

**Date: Sun, 19 Nov 2000 19:39:22 GMT**

Lámina 51

Dr. Roberto Gómez Cárdenas



## 100 (Continue Status)

- Permitir a un cliente que envía un mensaje de solicitud con la petición dentro del cuerpo del mensaje, determinar si el servidor está dispuesto a aceptar la solicitud.
- En algunos casos, puede ser inapropiado o altamente ineficiente para el cliente el enviar el cuerpo si el servidor rechaza el mensaje sin mirar al cuerpo de este.

Lámina 52

Dr. Roberto Gómez Cárdenas



## RFC-2616

- De acuerdo al RFC-2616:
  - If a client will wait for a 100 (Continue) response before sending the request body, it MUST send an Expect request-header field (section 14.20) with the "100-continue" expectation.
  - A client MUST NOT send an Expect request-header field (section 14.20) with the "100-continue" expectation if it does not intent to send a request body.

Lámina 53

Dr. Roberto Gómez Cárdenas



## Cientes HTTP 1.1

- Para cumplir con HTTP 1.1 los clientes deben:
  - Incluir el encabezado Host: en cada solicitud.
  - Aceptar respuestas en modo chunk.
  - Soportar conexiones persistentes o incluir el encabezado "Connection: close" en cada solicitud.
  - Ser capaces de manejar la respuesta "100 Continue".

Lámina 54

Dr. Roberto Gómez Cárdenas



## Analizadores protocolo HTTP WebProxy

---

1. Charles Web Debbing Proxy
  - Puede interceptar comunicaciones http/https/ssl raw sobre http (CONNECT), crea sus propios certificados copiando toda la información del certificado original.
  - Posee utilerías de crawling y estadísticos.
  - Tiene opción a recibir comunicaciones en un puerto para todos.
  
2. WebProxy 2.1
  - Sencillo y confiable, por lo general sólo permite la redirección en el localhost.
  - Intercepta comunicaciones HTTP/HTTPS.
  - Posee utilerías de crawling y estadísticos.


Lámina 55
Dr. Roberto Gómez Cárdenas



## Analizadores protocolo HTTP WebProxy

---

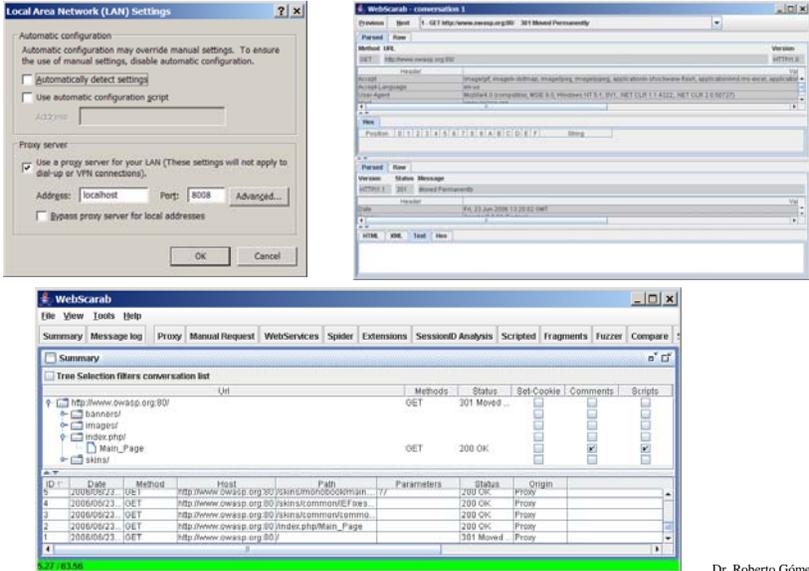
3. Paros Proxy
  - Sin clientes pesados, y de configuración sencilla.
  - Funcionalidad limitada a una interfaz web con pocas opciones de configuración.
  
4. WebScarab de OWASP
  - Tiene buenas características de crawling y análisis de vulnerabilidades a la Nessus.
  
5. Fiddler2
  - Proxy de depuración de aplicaciones web.
  - Inspección de tráfico HTTP(S).
  - Incluye subsistema de scripting con el tráfico de entrada o salida.


Lámina 56
Dr. Roberto Gómez Cárdenas

 **Screenshots WebScarab**



The image shows two screenshots. The top-left screenshot is the 'Local Area Network (LAN) Settings' dialog box. It has sections for 'Automatic configuration' (with checkboxes for 'Automatically detect settings' and 'Use automatic configuration script') and 'Proxy server' (with a checked box 'Use a proxy server for your LAN' and fields for 'Address: localhost' and 'Port: 8008'). The top-right screenshot shows the 'WebScarab - Conversation 1' window, displaying a list of requests and responses with columns for Method, URL, Status, and Cookies. The bottom screenshot shows the 'WebScarab' main interface with a 'Summary' tab selected, displaying a tree view of the selected filters and a table of request details.

Lámina 57 Dr. Roberto Gómez Cárdenas

 **Web Scanners**

- Nikto
  - Busca archivos peligrosos, CGI's y problemas específicos.
  - <http://www.cirt.net/nikto2>
- WebInspect (\$)
  - <http://www.spidynamics.com/products/webinspect/>
- Whisker/libwhisker
  - <http://www.wiretrip.net/rfp/>
- Burpsuite
  - <http://portswigger.net/suite/>





Lámina 58 Dr. Roberto Gómez Cárdenas



## Web Scanners

- Wikto
  - <http://www.sensepost.com/research/wikto/>
- Acunetix WVS (\$)
  - <http://www.acunetix.com/>
- Rational AppScan (S)
  - <http://www.ibm.com/software/awdtools/appscan/>
- N-Stealth(\$)
- Netsparker






Lámina 59

Dr. Roberto Gómez Cárdenas



## Ejemplo configuración y salida nikto-2.1.3

```

40 #####
41 # PROXY STUFF
42 #####
43 #PROXYHOST=10.10.10.221
44 #PROXYPORT=8080
45 #PROXYUSER=proxyuserid
46 #PROXYPASS=proxypassword
47 #####
48 # COOKIE STUFF
49 #####
50 # send a cookie with all requests
51 #STATIC-COOKIE=cookie=cookievalue
52 #####
53 # The below allows you to vary
54 # an HTTP(s) server is running
55 # server do not implement the
56 CHECKMETHODS=HEAD GET
57 #####
58 # If you want to specify the location of any of the files, specify them here
59 EXECDIR=/Users/L00445569/Desktop/nikto-2.1.3
60 #PLUGINDIR=/opt/nikto/plugins
61 # TEMPLATEDIR=/opt/nikto/templates
62 # DOCDIR=/opt/nikto/docs
        
```

```

D:\Users\L00445569\Desktop\nikto-2.1.3>c:\strawberry\perl\bin\perl.exe nikto.pl
-h 1
-46
-Nikto v2.1.3

+ Target IP: 14 46
+ Target Hostname: webt n.mx
+ Target Port: 80
+ Start Time: 2010-10-03 19:46:06

+ Server: Apache/2.2.10 (Linux/SUSE)
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ robots.txt contains 1 entry which should be manually viewed.
+ Etag header found on server. inode: 160817, size: 44, mtime: 0x3e9564c23b600
+ Apache/2.2.10 appears to be outdated (current is at least Apache/2.2.16). Apache
  1.3.42 and 2.0.63 are also current.
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSUDB-077: HTTP TRACE method is active, suggesting the host is vulnerable to X
  ST
+ OSUDB-3092: /manual/: Web server manual found.
+ OSUDB-3258: /icons/: Directory indexing found.
+ OSUDB-3260: /manual/images/: Directory indexing found.
+ OSUDB-3233: /icons/README: Apache default file found.
+ 6417 items checked: 45 error(s) and 9 item(s) reported on remote host
+ End Time: 2010-10-03 20:07:26 (1280 seconds)

+ 1 host(s) tested
        
```

Lámina 60

Dr. Roberto Gómez Cárdenas



## Screenshot Netsparker

---

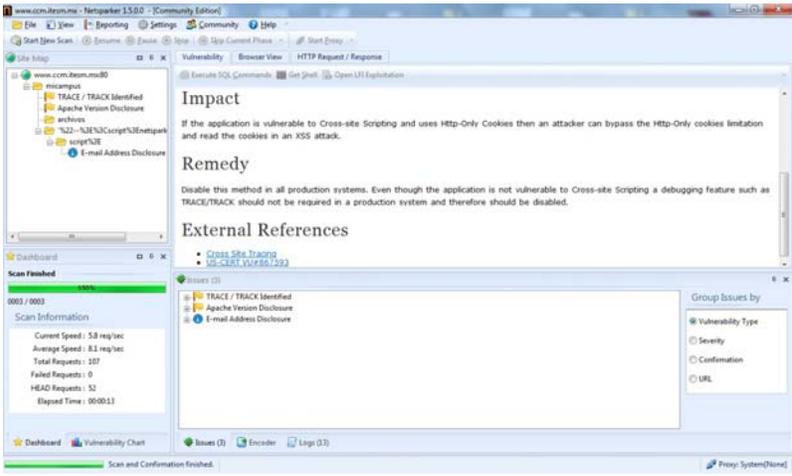


Lámina 61

Dr. Roberto Gómez Cárdenas



# El lenguaje HTML

## características

Lámina 62

Dr. Roberto Gómez Cárdenas



## El lenguaje HTML

- Hyper Text Markup Language
- Creado en 1989 por el físico Tim Berners-Lee
  - Propuso diseñar un sistema de unificación de acceso a los datos generados por los científicos del CERN (Centro Europeo para la Investigación Nuclear).
  - Se desarrolló una plataforma de tipo hipertexto y un protocolo de comunicaciones denominado HTTP, que permitía a los científicos consultar información de cualquier tema, aunque se encontrara diseminada en diferentes computadoras.
  - Se definió un lenguaje de documentos estructurados que se denominó HTML.

Lámina 63

Dr. Roberto Gómez Cárdenas



## HTML

- Lenguaje simple de programación que permite describir hipertexto.
- Hipertexto
  - Texto presentado de forma estructurada y agradable, con enlaces (hyperlinks) que conducen a otros documentos o fuentes de información relacionadas y con inserciones multimedia (gráficos, sonido, ...)
- Archivos con extensión .htm y .html

Lámina 64

Dr. Roberto Gómez Cárdenas



## Características HTML

- Descripción se basa en especificar la estructura lógica del contenido.
  - Títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc.
- También se especifica los diferentes efectos que se desea proporcionar.
  - Especificar los lugares del documento donde se debe poner cursiva, negrita, o una gráfica.
- La presentación final de dicho hipertexto la lleva a cabo una aplicación que interpreta el texto
  - Un navegador o browser

Lámina 65

Dr. Roberto Gómez Cárdenas



## El documento HTML

- Son archivos de texto plano (ASCII) que se pueden crear con cualquier editor de texto
  - notepad, vi, emacs, ...
- También hay disponibles editores HTML de tipo WYSIWYG ( What You See Is What You Get)
  - Hotmetal, Adone Pagemil, MS Frontpage, Dreamwaver.
- Navegadores cuentan con una opción para crear documentos HTML y se puede ver el resultado de forma inmediata.

Lámina 66

Dr. Roberto Gómez Cárdenas



## Arquitectura HTML

- Un documento se encuentra formado por elementos
  - Títulos, párrafos, listas, tablas...
- Para delimitar los elementos se utilizan etiquetas.

**<nombre-etiqueta>** Etiqueta de apertura

Contenido del elemento

**</nombre-etiqueta>** Etiqueta de cierre

Lámina 67
Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

- Ejemplos: <b>, <font>, <tle>, <p> etc.
- Generalmente se usan en pares: abrir(<b>) y cerrar (<\b>)

Efecto	Código	Ejemplo	Resultado
Bold	B	<B> Bold </B>	<b>Bold</b>
Italic	I	<I> Italic </I>	<i>Italic</i>

- Etiquetas simples (no se cierran): <hr>, <br>
- Las etiquetas no son sensibles a las mayúsculas ó minúsculas

Lámina 68
Dr. Roberto Gómez Cárdenas



## Listado de etiquetas

---

• <!>	• <DIR>	• <LI>	• <SMALL>
• <A>	• <DL>	• <LISTING>	• <STRONG>
• <ADDRESS>	• <DT>	• <LIT>	• <SUB>
• <BASE>	• <EM>	• <MARQUEE>	• <SUP>
• <BASEFONT SIZE>	• <EMBED>	• <MENU>	• <TABLE>
• <BGSOUND>	• <FONT>	• <META>	• <TD>
• <BIG>	• <FORM>	• <NEXTID>	• <TEXTAREA>
• <BLINK>	• <H1 ...H6>	• <NOBR>	• <TH>
• <BLOCKQUOTE>	• <HEAD>	• <OL>	• <TITLE>
• <BODY>	• <HR>	• <OPTION>	• <TR>
•  	• <HTML>	• <P>	• <TT>
• <CAPTION>	• <I>	• <P ALIGN>	• <UL>
• <CENTER>	• <IMG>	• <PLAINTEXT>	• <VAR>
• <CITE>	• <INPUT>	• <PRE>	• <WBR>
• <CODE>	• <ISINDEX>	• <S>	• <XMP>
• <DD>	• <ISMAP>	• <SAMP>	
• <DFN>	• <KBD>	• <SELECT>	

Lámina 69 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

---

Apertura	Acción	Atributos	Cierre
< !>	Comentario.	Ninguno	-->
<A>	Hipervínculo.	HREF, NAME, REL, REV, TITLE	</A>
<ADDRESS>	Formato para dirección del autor.	Ninguno	</ADDRESS>
<BASE>	Url del autor; contexto del documento.	HREF	</BASE>
<BASEFONT SIZE>	Tamaño de la fuente base.	Ninguno	NO
<BGSOUND>	Sonido de fondo.	SRC, LOOP.	NO - Internet Explorer
<BIG>	Aumenta el tamaño.	Ninguno	</BIG>

Lámina 70 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<BLINK>	Hace parpadear el texto.	Ninguno	</BLINK> - Netscape
<BLOCKQUOTE>	Da formato con sangría a un párrafo	Ninguno	</BLOCKQUOTE>
<BODY>	Cuerpo del documento.	BGCOLOR, BACKGROUND, TEXT, LINK, VLINK, ALINK	</BODY>
 	Retorno de línea.	CLEAR: Se utiliza en combinación con ALIGN de IMAGE.	NO
<CAPTION>	Posición de la leyenda en una tabla.	ALIGN: TOP/BOTTOM. Internet Explorer: LEFT, RIGHT, CENTER	</CAPTION>

Lámina 71 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<CENTER>	Centrar.	Ninguno	</CENTER>
<CITE>	Formato para citas en itálicas.	Ninguno	</CITE>
<CODE>	Formato en tipo código.	Ninguno	</CODE>
<DD>	Definiciones marcadas, para Lista de Definiciones <DL>.	Ninguno	NO
<DFN>	Formato en itálica.	Internet Explorer	</DFN>
<DIR>	Lista de directorio, con elementos marcados con <LI>.	Ninguno	</DIR>

Lámina 72 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<DL>	Lista de Definiciones, con términos marcados con <DT> y definiciones marcadas con <DD>.	Ninguno	</DL>
<DT>	Términos marcados, para Lista de Definiciones <DL>.	Ninguno	NO
<EM>	Formato enfatizado en itálica.	Ninguno	</EM>
<EMBED>	Sonido de Fondo.	SRC, WIDTH, HEIGHT, AUTOSTART, LOOP.	NO - Netscape
<FONT>	Definición de la fuente.	SIZE, COLOR. Internet Explorer: FACE.	</FONT>
<FORM>	Para ingreso de datos del usuario en un formulario.	ACTION, METHOD	</FORM>

Lámina 73 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<H1 ...H6>	Tamaño de letras del 1 al 6.	HTML 3.0: LEFT, CENTER, RIGHT	</H1 .../H6>
<HEAD>	Encabezamiento del documento.	BASE, TITLE, ISINDEX, NEXTID, META	</HEAD>
<HR>	Línea horizontal.	NOSHADE, SIZE, WIDTH, ALIGN. Internet Explorer: COLOR	NO
<HTML>	Al principio y al fin de todo documento.	HEAD, BODY	</HTML>
<I>	Itálica (Cursiva).	Ninguno	</I>
<IMG>	Cargar imágenes.	ALIGN, SRC, ALT, ISMAP, WIDTH, HEIGHT, VSPACE, HSPACE	NO

Lámina 74 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<INPUT>	Define un objeto de ingreso en un formulario.	TYPE, NAME, VALUE, SIZE, MAXLENGHT, ALIGN, SRC, CHECKED	</INPUT>
<ISINDEX>	Indica que existe un index en el server para el documento.	Netscape: PROMPT	NO
<ISMAP>	Activa la selección de imágenes para el usuario.	Ninguno	NO
<KBD>	Formato monoespaciado.	Ninguno	</KBD>
<LI>	Ítem de lista.	Netscape: VALUE, TYPE	NO
<LISTING>	Listados	Ninguno. Obsoleto.	</LISTING>

Lámina 75 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<LIT>	Literal. Como PRE, pero usa letra proporcional.	Ninguno	</LIT>
<MARQUEE>	Marquesina.	ALIGN, BEHAVIOR, BGCOLOR, DIRECTION, HEIGHT, WIDTH, HSPACE, VSPACE, LOOP, SCROLLAMOUNT, SCROLLDELAY.	</MARQUEE> - Internet Explorer
<MENU>	Lista menú.	Ninguno	</MENU>
<META>	Metainformación ubicada en HEAD.	EQUIV, CONTENT, NAME	NO
<NEXTID>	Es un parámetro que identifica al documento.	NO	NO
<NOBR>	Elimina los saltos de líneas.	Ninguno	NO

Lámina 76 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<OL>	Lista ordenada, con elementos marcados con <LI>.	TYPE, START, VALUE.	</OL>
<OPTION>	Opción de selección dentro de un formulario.	VALUE, SELECTED VALUE	NO
<P>	Retorno de línea, con un espacio.	Ninguno	NO
<P ALIGN>	Alineación de texto.	LEFT, CENTER, RIGHT	</P>
<PLAINTEXT >	Pasaje de texto plano.	Ninguno. Obsoleto.	</PLAINTEXT>
<PRE>	Visualiza el texto en su formato original.	WIDTH	</PRE>
<S>	Texto tachado.	Ninguno	</S>

Lámina 77 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<SAMP>	Formato tipo ejemplo.	Ninguno	</SAMP>
<SELECT>	Para selección de opciones dentro de un formulario.	NAME, SIZE, MULTIPLE	</SELECT>
<SMALL>	Disminuye el tamaño.	Ninguno	</SMALL>
<STRONG>	Formato enfatizado más fuerte que <EM>.	Ninguno	</STRONG>
<SUB>	Subíndice.	Ninguno	</SUB>
<SUP>	Superíndice.	Ninguno	</SUP>
<TABLE>	Tabla.	BORDER, CELLPADDING, CELLSPACING, HEIGHT, WIDTH. Internet Explorer: COLOR	</TABLE>

Lámina 78 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<TD>	Celdas de una fila en una tabla, dentro de <TR>.	ALIGN, VALIGN, NOWRAP, COLSPAN, ROWSPAN, HEIGHT, WIDTH	</TD>
<TEXTAREA>	Área para ingreso de texto dentro de un formulario.	NAME, ROWS, COLS.	</TEXTAREA>
<TH>	Título de Tabla.	ROWSPAN, COLSPAN, ALIGN, VALIGN, NOWRAP, HEIGHT, WIDTH	</TH>
<TITLE>	Título dentro de HEAD.	Ninguno	</TITLE>
<TR>	Fila de una Tabla.	ALIGN, VALIGN	</TR>

Lámina 79 Dr. Roberto Gómez Cárdenas



## Etiquetas HTML

Apertura	Acción	Atributos	Cierre
<TR>	Fila de una Tabla.	ALIGN, VALIGN	</TR>
<TT>	Formato tipo máquina.	Ninguno	</TT>
<UL>	Lista no ordenada, con elementos marcados con <LI> .	COMPACT, TYPE	</UL>
<VAR>	Formato tipo variable.	Ninguno	</VAR>
<WBR>	Se usa con NOBR para una sección que deba ser separada.	Ninguno	NO
<XMP>	Similar a PRE.	Ninguno	</XMP>

Lámina 80 Dr. Roberto Gómez Cárdenas



## Elementos HTML

---

- Los documentos HTML se encuentran definidos por elementos HTML.
- Un elemento HTML es todo lo que se encuentra entre una etiqueta inicial y una etiqueta final.

Etiqueta inicio	Contenido elemento	Etiqueta final
<p>	Esto es un parrafo	</p>
<a href="default.htm" >	Esto es una liga	</a>

Lámina 81
Dr. Roberto Gómez Cárdenas



## Características elementos HTML

---

- Un elemento HTML empieza con una etiqueta de inicio o de apertura.
- Un elemento HTML termina con una etiqueta de termino o de cierre.
- El contenido del elemento es todo lo que se encuentra entre las etiquetas de apertura y cierre.
- Algunos elementos HTML cuentan con contenido vacío.
- Elementos vacíos se cierran en la etiqueta de inicio.
- La mayor parte de los elementos pueden tener atributos.

Lámina 82
Dr. Roberto Gómez Cárdenas



## Atributos HTML

---

- Elementos HTML pueden contar con atributos.
- Los atributos proporcionan información adicional acerca de un elemento.
- Los atributos son especificados en la etiqueta de inicio.
- Los atributos se presentan en pares nombre/valor como nombre="valor"
  - Los valores deben ir entre comillas.
- Ejemplo
 

```
<a href="http://www.google.com">Esto es una liga </a>
```

Lámina 83
Dr. Roberto Gómez Cárdenas



## Atributos más utilizados

---

Atributo	Valor	Descripción
class	Nombre de una clase.	Especifica un nombre de clase para un elemento
id	Identificador	Especifica un identificador único para un elemento
style	Definición de estilo	Especifica un estilo de linea para un elemento
title	Texto	Especifica información extra acerca de un elemento,

Lámina 84
Dr. Roberto Gómez Cárdenas



## Más atributos

Nombre	Etiqueta	Acción	Valor
HREF	Dirección del URL local o remoto.	href="www.hp.com"	
LOOP	<BGSOUND>	Número de veces que se reproduce el archivo de sonido. Para permanente se usa INFINITE o -1.	loop=infinite
SRC		Nombre del archivo de sonido (Internet Explorer)	src="sonido.wav"
LOOP	<EMBED>	Número de veces que se reproduce el archivo de sonido. Para permanente se usa INFINITE o -1.	loop=-1
SRC		Nombre del archivo de sonido (con Crescendo)	src="sonido.mid"

Lámina 85 Dr. Roberto Gómez Cárdenas



## Más atributos

Nombre	Etiqueta	Acción	Valor
SRC	<IMAGE>	Nombre del gráfico	src="imagen.gif"
ALT		Nombre que reemplaza a la imagen cuando ésta no puede ser cargada.	alt="Nombre de Imágen"
BGCOLOR	<BODY>	Color de fondo	bgcolor="#FFFFFF" (blanco)
BACKGROUND		Imágen de fondo	background="foto.gif"
TEXT		Color del texto	text="#000000" (negro)
LINK		Color de vínculo	link="#0000FF" (azul)
VLINK		Color de vínculo visitado	vlink="#FF0000" (rojo)
ALINK		Color de vínculo presionado	alink="#00FF00" (verde)

Lámina 86 Dr. Roberto Gómez Cárdenas



## Más atributos

Nombre	Etiqueta	Acción	Valor
ALIGN	<CAPTION> dentro de <TABLE>	Colocar título arriba (TOP) o debajo (BOTTOM) de la tabla.	align=top
ALIGN	<H1..H6>, <P ALIGN>, <TD>, <TH>, <TR>, <DIV>,	Alinear el texto: LEFT, RIGHT, CENTER	align=center
ALIGN		Alinear el texto: LEFT, RIGHT, CENTER	align=center
SIZE	<HR>	Valor de la altura en pixels o porcentaje	size=3
WIDTH		Valor del ancho en pixels o porcentaje	width=50%

Lámina 87 Dr. Roberto Gómez Cárdenas



## Más atributos

Nombre	Etiqueta	Acción	Valor
SIZE		Tamaño de la letra de 1 a 7	size=6
COLOR	<FONT>	Color de la letra	color="#000000" (negro)
FACE		Fuente del texto (Internet Explorer)	face="helv" (helvética)
HEIGHT	<EMBED>, <IMAGE>,	Valor de la altura en pixels o porcentaje	height=80
WIDTH	<MARQUEE>, <TD>, <TH>	Valor del ancho en pixels o porcentaje	width=50%

Lámina 88 Dr. Roberto Gómez Cárdenas



## Más atributos

Nombre	Etiqueta	Acción	Valor
BORDER	<TABLE>	Borde y ancho del borde en la tabla	border=5
CELLPADDING		Espacio entre el borde y el texto	cellpadding=10
CELLSPACING		Espacio entre las líneas interna y externa del borde	cellspacing=3
HEIGHT		Valor de la altura en pixels o porcentaje	height=80
WIDTH		Valor del ancho en pixels o porcentaje	width=50%
COLSPAN	<TD> dentro de <TABLE>	Expandir una celda varias columnas	colspan=4
ROWSPAN		Expandir una celda varias filas	rowspan=4

Lámina 89 Dr. Roberto Gómez Cárdenas



## Caracteres especiales

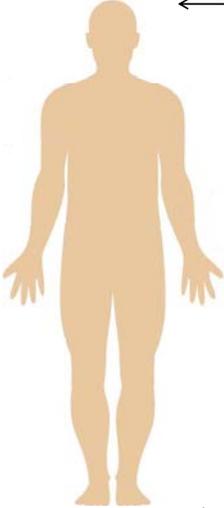
- Algunos caracteres tienen un significado especial en HTML, si se desea que el browser los despliegue hay que remplazarlos en el código HTML.

Carácter	Descripción	Código HTML
	Espacio en blanco	&nbsp;
<	Menor que	&lt;
>	Mayor que	&gt;
&	Ampersand	&amp;
“	Comillas	&quot;
©	Copyright	&copy;

Lámina 90 Dr. Roberto Gómez Cárdenas

 **Estructura básica**

---

 ← `<HTML>`

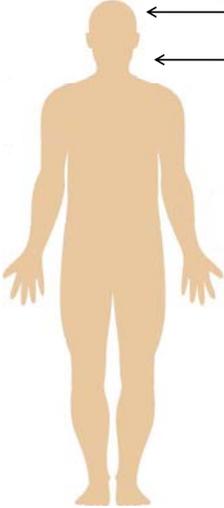
Define el comienzo y el fin del documento.

← `</HTML>`

Lámina 91 Dr. Roberto Gómez Cárdenas

 **Estructura básica**

---

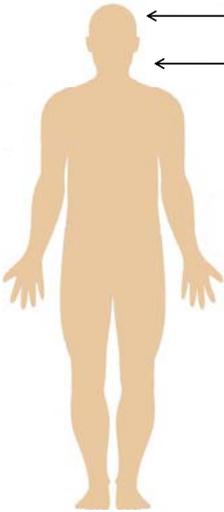
 ← `<HTML>` `<HEAD>`

← `</HEAD>`

`</HTML>`

Lámina 92 Dr. Roberto Gómez Cárdenas

 **Estructura básica**

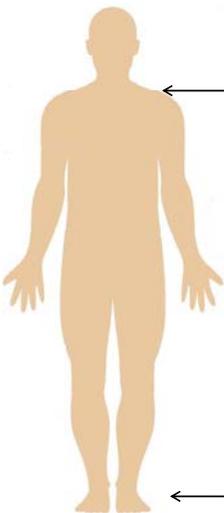


```
<HTML>
<HEAD> <TITLE>
</TITLE>
</HEAD>
</HTML>
```

Define la cabecera del documento, título, elementos invisibles, funciones en otros lenguajes, etc

Lámina 93 Dr. Roberto Gómez Cárdenas

 **Estructura básica**



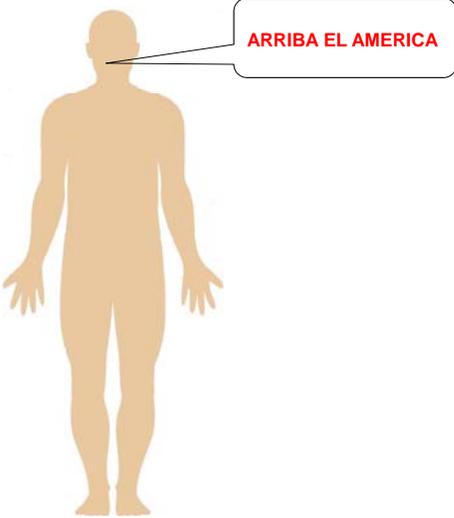
```
<HTML>
<HEAD> <TITLE>
</TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

Define el cuerpo del documento. En su interior se encuentran todas las etiquetas y contenidos necesarios que forman la página.

Lámina 94 Dr. Roberto Gómez Cárdenas



## Estructura básica



```
<HTML>
  <HEAD>
    <TITLE>
      BETO
    </TITLE>
  <BODY>
    ARRIBA EL AMERICA
  </BODY>
</HTML>
```

Humano llamado BETO

Lámina 95 Dr. Roberto Gómez Cárdenas



## Documento simple

```
<html>
  <head>
    <TITLE>Un documento simple</TITLE>
  </head>
  <body>
    Esto es un texto que no tendrá estilos, ni características de
    ningún tipo.
  </body>
</html>
```

Lámina 96 Dr. Roberto Gómez Cárdenas



## Ejemplo archivo HTML

---

```

<html>
<body>
  <b>Texto en negrillas</b>

  <br>
  <strong>
    Texto strong
  </strong>
  <br>

  <i> Texto en italica </i>

</body>
</html>

```

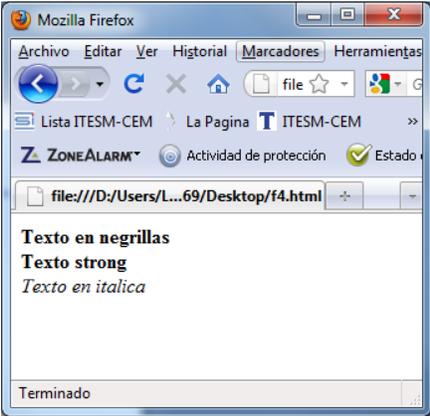


Lámina 97
Dr. Roberto Gómez Cárdenas



## Encabezados

---

- Existen 6 niveles de encabezados

**<H1>Encabezado de nivel 1</H1>**

**<H2>Encabezado de nivel 2</H2>**

**<H3>Encabezado de nivel 3</H3>**

**<H4>Encabezado de nivel 4</H4>**

**<H5>Encabezado de nivel 5</H5>**

**<H6>Encabezado de nivel 6</H6>**

Lámina 98
Dr. Roberto Gómez Cárdenas



## Listas

---

- Lista no numerada

**Código:**

```
<ul>
<li>Pera</li>
<li>Manzana</li>
</ul>
```

**Salida:**

- Pera
- Manzana

- Lista numerada

**Código:**

```
<ol>
<li>Pera</li>
<li>Manzana</li>
</ol>
```

**Salida:**

1. Pera
2. Manzana

Lámina 99
Dr. Roberto Gómez Cárdenas



## Texto párrafo

---

- Texto pre-formateado: `<pre> </pre>`
- Párrafo: `<p> </p>`

```
<html>
<head><TITLE>Texto Preformatado</TITLE></head>
<body>
<pre>
El siguiente texto es preformateado
Ahora si que valen los blancos,
tabuladores y retornos de carro.
</pre>
<p> el no preformateado
suele ser proporcional
</body>
</html>
```

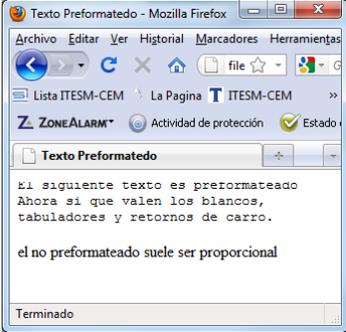


Lámina 100
Dr. Roberto Gómez Cárdenas



## Bloques de texto

- Sintaxis: `<blockquote>` `</blockquote>`

```

<html>
<head><TITLE>Bloques</TITLE></head>
<body>
<p>Esto es un párrafo normal</p>
<blockquote><p align=justify>Esto es un "bloque de texto", párrafo
usado en citas o para resaltar el bloque de texto. Normalmente se
aumentan los márgenes derecho e izquierdo para distinguirlo del
resto del texto</blockquote>
<p align=justify>Aquí otro párrafo estándar con suficiente texto
como
para apreciar las diferencias de los márgenes. Disminuir el tamaño
de
la ventana del navegador también ayuda.
</body>
</html>
                
```

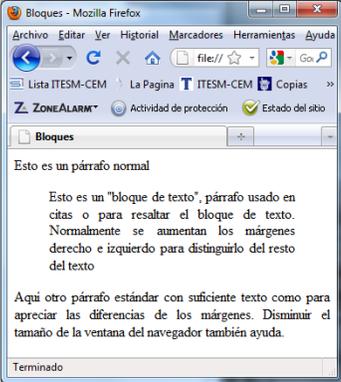


Lámina 101
Dr. Roberto Gómez Cárdenas



## Tablas

```

<table border="1">
<tr>
<th>Encabezado 1</th>
<th>Encabezado 2</th>
</tr>
<tr>
<td>fila 1, columna 1</td>
<td>fila 1, columna 2</td>
</tr>
<tr>
<td>fila 2, columna 1</td>
<td></td>
</tr>
</table>
                
```

Encabezado 1	Encabezado 2
fila 1, columna 1	fila 1, columna 2
fila 2, columna 1	

Lámina 102
Dr. Roberto Gómez Cárdenas



## Ligas

---

- Hipertexto
  - < a href="http://homepage.cem.itesm.mx/rogomez">  
Página del profesor</a>
  - Salida: [Página del profesor](http://homepage.cem.itesm.mx/rogomez)
- Correo
  - < a href="mailto:rogomez@itesm.mx">  
correo</a>
  - Salida: [Correo](mailto:rogomez@itesm.mx)

Lámina 103 Dr. Roberto Gómez Cárdenas



## Insertando Imágenes

---

- Imagen usada como etiqueta
  - <img src> es una etiqueta simple (no se cierra)
  - 
  - Salida 
- Imagen usada como Hyperlink
  - <a href="http://www.cem.itesm.mx/"></a>
  - Salida 

Lámina 104 Dr. Roberto Gómez Cárdenas



TEC  
DE MONTERREY  
Campus Estado de México

## Tamaño de la imagen

```

```



Width

Height

Lámina 105

Dr. Roberto Gómez Cárdenas



TEC  
DE MONTERREY  
Campus Estado de México

## XHTML

- EXtensible HyperText Markup Language
- Caso idéntico HTML 4.0.1
- Versión más limpia y estricta de HTML.
- Definida como una aplicación HTML.
- Es una recomendación W3C.
- Todos los browsers soportan XHTML.

Lámina 106

Dr. Roberto Gómez Cárdenas



## HTML vs XHTML

- Principales diferencias
  - Todos los elementos se anidan propiamente.
  - Todos los nombres de los elementos se escriben con minúsculas.
  - Todos los nombres de atributos deben escribirse con minúsculas.
  - Los valores de los atributos se encierran entre comillas simples o dobles.
  - Las etiquetas vacías se deben escribir con un espacio y un / al final.

Lámina 107

Dr. Roberto Gómez Cárdenas



## Ejemplo de un código XHTML incorrecto

```
<html>
<head>
<title>Ejemplo de un HTML incorrecto</title>
<body>
<h1>HTML incorrecto
<p>Esto es un párrafo
</body>
```



Lámina 108

Dr. Roberto Gómez Cárdenas



## Ejemplos diferencia en anidado

HTML	XHTML
<b>&lt;b&gt;&lt;i&gt;Esto es una prueba&lt;/b&gt;&lt;/i&gt;</b>	<b>&lt;b&gt;&lt;i&gt;Esto es una prueba&lt;/i&gt;&lt;/b&gt;</b>
<pre> &lt;ul&gt; &lt;li&gt;Cafe&lt;/li&gt; &lt;li&gt;Te   &lt;ul&gt;     &lt;li&gt;Te negro&lt;/li&gt;     &lt;li&gt;Te verde&lt;/li&gt;   &lt;/ul&gt; &lt;li&gt;Leche&lt;/li&gt; &lt;/ul&gt; </pre>	<pre> &lt;ul&gt; &lt;li&gt;Cafe&lt;/li&gt; &lt;li&gt;Te   &lt;ul&gt;     &lt;li&gt;Te negro&lt;/li&gt;     &lt;li&gt;Te verde&lt;/li&gt;   &lt;/ul&gt; &lt;/li&gt; &lt;li&gt;Leche&lt;/li&gt; &lt;/ul&gt; </pre>

Lámina 109 Dr. Roberto Gómez Cárdenas



## Ejemplos elementos deben estar cerrados

- Primer ejemplo
  - HTML
 

```
<p> Esto es un párrafo
<p> Esto es otro párrafo
```
  - XHTML
 

```
<p> Esto es un párrafo </p>
<p> Esto es otro párrafo </p>
```
- Segundo ejemplo
  - HTML
 

```
Un break: <br>
Una línea horizontal: <hr>
Una imagen : 
```
  - XHTML
 

```
Un break: <br />
Una línea horizontal: <hr />
Una imagen : 
```

Lámina 110 Dr. Roberto Gómez Cárdenas



## Ejemplo elementos deben estar en minúsculas

- HTML

```
<BODY>
<P>Esto es un párrafo</P>
</BODY>
```
- XHTML

```
<body>
<p>Esto es un párrafo</p>
</body>
```

Lámina 111 Dr. Roberto Gómez Cárdenas



## Nombres y valores atributos

- Nombres de atributos deben estar en minúsculas
  - HTML `<table WIDTH="100%">`
  - XHTML `<table width="100%">`
- Valores de atributos deben ir entre comillas.
  - HTML `<table width=100%>`
  - XHTML `<table width="100%">`

Lámina 112 Dr. Roberto Gómez Cárdenas



## Se prohíbe la minimización de atributos

---

- Se prohíbe la minimización de atributos

HTML	XHTML
<input checked>	<input checked="checked" />
<input readonly>	<input readonly="readonly" />
<input disabled>	<input disabled="disabled" />
<option selected>	<option selected="selected" />
<frame noresize>	<frame noresize="noresize" />

Lámina 113
Dr. Roberto Gómez Cárdenas



## Equivalencias

---

HTML	XHTML
compact	compact="compact"
checked	checked="checked"
declare	declare="declare"
readonly	readonly="readonly"
disabled	disabled="disabled"
selected	selected="selected"
defer	defer="defer"
ismap	ismap="ismap"
nohref	nohref="nohref"
noshade	noshade="noshade"
nowrap	nowrap="nowrap"
multiple	multiple="multiple"
noresize	noresize="noresize"

Lámina 114
Dr. Roberto Gómez Cárdenas



## El atributo lang

---

- Puede ser usado en casi cualquier elemento XHTML.
- Especifica el lenguaje del contenido dentro de un elemento.
- Si se usa este atributo, se debe añadir el atributo xml:lang.
- Ejemplo:

```
<div lang="it" xml:lang="it">Ciao bella!</div>
```

Lámina 115
Dr. Roberto Gómez Cárdenas



## Elementos XHTML mandatorios

---

- Un documento XHTML debe contar con una declaración DOCTYPE.
- Los elementos html, head, title y body deben estar presentes.
- Ejemplo:

```

<!DOCTYPE Doctype goes here>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Aquí va el título</title>
</head>

<body>
</body>

</html>
```

Declaración DOCTYPE  
no es parte de un  
documento XHTML en sí.  
No es un elemento XHTML.

Atributo xmlns especifica un  
Espacio de nombres xml  
Para el documento.  
Si no se especifica, se asume  
el valor que se presenta en  
el ejemplo..

Lámina 116
Dr. Roberto Gómez Cárdenas



## DTD

- Document Type Definitions.
- Especifica la sintaxis de una página web.
- Permiten especificar reglas para documentos de un tipo en particular, incluyendo un conjunto de elementos y declaración de entidades.
- Existen tres DTDs de XHTML
  - STRICT
  - TRANSITIONAL
  - FRAMESET

Lámina 117 Dr. Roberto Gómez Cárdenas



## Estructura documentos XHTML

- Un documentos XHTML consiste de tres partes
  - La declaración DOCTYPE
  - La sección <head>
  - La sección <body>
- La estructura básica del documento es

```
<!DOCTYPE ...>
<html>
<head>
<title>... </title>
</head>
<body> ... </body>
</html>
```

Lámina 118 Dr. Roberto Gómez Cárdenas



## DTD Strict

- El DTD sigue al pie de la letra las convenciones marcadas separando completamente las etiquetas de formato del contenido.
- Todos los aspectos de presentación se especifican en hojas de estilo “Cascading Style Sheets” (CSS).
- Declaración.

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

Lámina 119

Dr. Roberto Gómez Cárdenas



## DTD Transitional

- El DTD “transitional” se utiliza en los documentos xhtml que requieren compatibilidad con browsers antiguos que no aceptan CSS.
- Permite la utilización de algunas etiquetas de formato dentro del código de las páginas.
  - Mucho de la presentación de HTML puede ser utilizado.
- Declaración.

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

Lámina 120

Dr. Roberto Gómez Cárdenas



## DTD Frameset

- El DTD “Frameset” es similar en funcionalidad al “Transitional”.
- Se utiliza cuando se divide la pantalla en varios marcos o frames.
- Declaración

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

Lámina 121 Dr. Roberto Gómez Cárdenas



## Validación

- Un documento XHTML se valida contra un Document Type Definition (DTD).
- Para poder ser validado, se debe incluir el DTD correspondiente en la primera línea del documento XHTML.

Lámina 122 Dr. Roberto Gómez Cárdenas



## Estándares

---

- El Consorcio World Wide Web (W3C) es un consorcio internacional donde las organizaciones miembro, personal a tiempo completo y el público en general, trabajan conjuntamente para desarrollar estándares Web.
- Entre los servicios que ofrece se destacan:
  - Los tutoriales <http://www.w3schools.com/>
  - El servicio de validación de documentos <http://validator.w3.org/>



Lámina 123

Dr. Roberto Gómez Cárdenas



## El validador

---

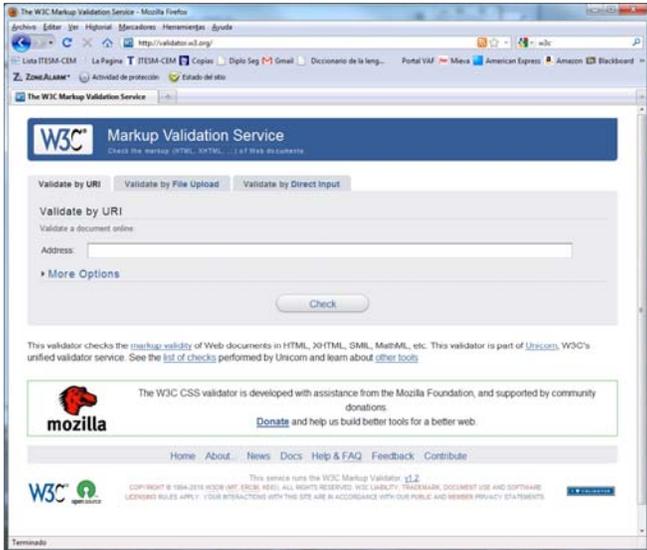


Lámina 124

Dr. Roberto Gómez Cárdenas



## Formas

- Una forma es un área que puede contener elementos.
- Se utilizan para
  - Proporcionar información
  - Proveer interactividad
- Contiene y organiza otros objetos
  - Cajas de texto
  - Cajas de chequeo
  - Botones

Lámina 125 Dr. Roberto Gómez Cárdenas



## La etiqueta form

- Sintaxis: `<form>`
  - Contiene a los elementos de la forma.
- Atributos requeridos
  - method
    - Puede ser get o post.
    - Es la manera como se comunica la forma con el servidor.
  - action
    - URL del script que va a procesar la forma.
  - name
    - Nombre el elemento de tal forma que puedan referirse a este, desde una hoja de estilo o un script.
  - id
    - Atributo name es mantenido para compatibilidad hacia atrás; se debe usar el atributo id.
- Ejemplo

```
<form id="forma" name="forma" method="post"
action="/codigo/validar.php">
</form>
```

Lámina 126 Dr. Roberto Gómez Cárdenas



## Formas

- La etiqueta *FORM* actúa como una especie de contenedor para almacenar elementos que permiten al usuario seleccionar o introducir datos.
- Todos los datos se enviarán a la dirección URL indicada en el atributo *ACTION* de la etiqueta *FORM*, por el método indicado en el atributo *METHOD*.
- Se puede insertar cualquier elemento HTML en una etiqueta *FORM* (como texto, botones, tablas y enlaces), pero los elementos interactivos son los más interesantes.

Lámina 127

Dr. Roberto Gómez Cárdenas



## Controles

- Los usuarios interactúan con formas a través de controles.
- Cada control cuenta con un valor inicial y un valor final.
- Ambos valores son cadenas de caracteres.
- Cada control define sus valores iniciales y restricciones en los posibles valores.
- Valor inicial definido a través del atributo *value*.
- El valor actual se inicializa con el valor inicial.
- Después el valor puede modificarse a través de la interacción con el usuario y scripts.

Lámina 128

Dr. Roberto Gómez Cárdenas



## La etiqueta <input />

---

- La etiqueta <input /> es usada para configurar los elementos de la forma, a través de elementos “interactivos”.
- Sintaxis
 

```
<INPUT type="Nombre de campo"
      value="Valor predeterminado"
      name="Nombre del elemento" >
```
- Donde
  - value: valor inicial
  - name: permite a un script reconocer qué campo esta asociado con un par nombre/valor
    - El nombre del campo estará seguido de un signo (“=”) seguido de un valor que el usuario introdujo, o si el usuario no introdujo ningún valor, por el valor predeterminado.
  - type: especifica que tipo de elemento se representa con la etiqueta INPUT.

Lámina 129
Dr. Roberto Gómez Cárdenas



## Atributos type de la etiqueta <input />

---

Atributo	Significado
checkbox	Las casillas de elección pueden adoptar uno de dos estados: checked (seleccionado) o unchecked (no seleccionado). Cuando la casilla es seleccionada, el par nombre/valor se envía al CGI.
hidden	Este campo, que el navegador no muestra, es para definir una configuración única que se enviará al CGI como par nombre/valor.
file	Un campo que permite al usuario especificar una ruta de archivo que lleva al archivo que se enviará con el formulario
image	Un <i>botón de envío personalizado</i> que aparece cuando se ubica una imagen en la ubicación definida por el atributo SRC.
password	Una <i>casilla de texto</i> donde los caracteres escritos aparecen como asteriscos para camuflar el texto de entrada.
radio	Un <i>botón</i> que permite al usuario elegir entre varias opciones. Cada uno de estos botones debe tener el mismo atributo <i>name</i> . El par nombre/valor del botón radio seleccionado se enviará al CGI

Lámina 130
Dr. Roberto Gómez Cárdenas



## Atributos type de la etiqueta <input />

Atributo	Significado
reset	Un <i>botón de restauración</i> para quitar todos los elementos en el formulario y restablecer sus valores predeterminados.
submit	Un <i>botón de envío</i> para enviar el formulario. El texto en el botón puede definirse usando el atributo <i>value</i> .
text	Una <i>casilla de texto</i> para escribir una línea de texto. El tamaño de la casilla puede definirse usando el atributo <i>size</i> y la extensión máxima del texto con el atributo <i>maxlength</i> .

Lámina 131
Dr. Roberto Gómez Cárdenas



## <input> Text box

- Permite recibir información escrita.
- Atributos
  - **type** = “text”
  - **name**: “variable” donde se almacena la información capturada
  - **size**: longitud del cuadro de captura
  - **maxlength**: longitud máxima del texto a introducir en el cuadro de captura.
  - **value**: valor predefinido a aparecer en la ventana

■ Atributo opcional
■ Atributo obligatorio

Lámina 132
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<form method="get" action="http://10.17.40.23/servlet/eco">
  Formato preferido?
  <input type="text" name="formato" size="20"></input>
  <br />
  Nombre
  <input type="text" name="nombre" size="20"></input>
  <br /><br />
  <input type="submit" value="enviar"></input>
</form>

```

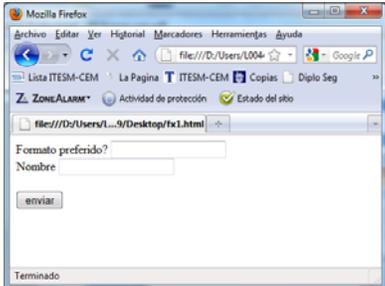


Lámina 133
Dr. Roberto Gómez Cárdenas



## <input> Password

---

- La información se oculta.
- Atributos
  - **type** = “password”
  - **name**: “variable” donde se almacena la información capturada
  - **size**: longitud del cuadro de captura
  - **maxlength**: longitud máxima del texto a introducir en el cuadro de captura.
  - **value**: valor predefinido a aparecer en la ventana

■ Atributo opcional
■ Atributo obligatorio

Lámina 134
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<form method="get" action="http://10.17.40.23/servlet/eco">
  Usuario
  <input type="password" name="usuario" size="20"></input>
  <br />
  Contraseña
  <input type="password" name="password" size="20"></input>
  <br /><br />
  <input type="submit" value="enviar"></input>
</form>

```

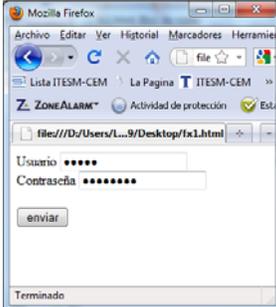


Lámina 135
Dr. Roberto Gómez Cárdenas



## <input> Check box

---

- El usuario puede seleccionar una o más opciones de un grupo determinado.
- Atributos
  - **type**="checkbox"
  - **name**: "variable" donde se almacena la información capturada
  - **value**: lo que el browser va a enviar si la opción es elegida.
  - **checked**: la opción esta seleccionada por default.

■ Atributo opcional
■ Atributo obligatorio

Lámina 136
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<h3>Elija todas las que apliquen</h3>
<form method="get" action="http://10.17.40.23/servlet/eco">
Linux
<input type="checkbox" name="linux"
      value="linux" />
<br /><br />
Solaris
<input type="checkbox" name="solaris"
      value="solaris" />
<br /><br />
<input type="submit" value="enviar" />
</form>

```

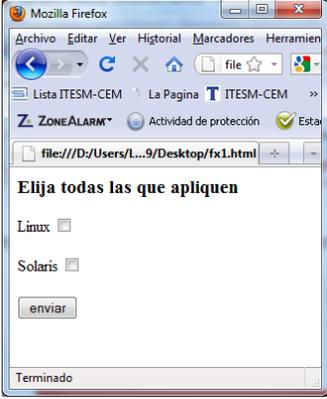


Lámina 137
Dr. Roberto Gómez Cárdenas



## <input> Radio Button

---

- El usuario solo puede seleccionar una opción por cada grupo de ellas.
- Cada botón de un grupo tiene el mismo nombre pero un valor distinto.
- Atributos
  - **type**="radio"
  - **name**: "variable" donde se almacena la información capturada
  - **value**: lo que el browser va a enviar si la opción es elegida.
  - **checked**: la opción esta seleccionada por default

■ Atributo opcional
■ Atributo obligatorio

Lámina 138
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<h3>Como te consideras de acuerdo a tu actividad
física?</h3>
<form method="get"
action="http://10.17.40.23/servlet/eco">
  Deportista
  <input name="deportista" value="si" type="radio" />
  <br /><br />
  No deportista
  <input name="deportista" value="no" type="radio" />
  <br /><br />
  <input value="enviar" type="submit" />
</form>

```

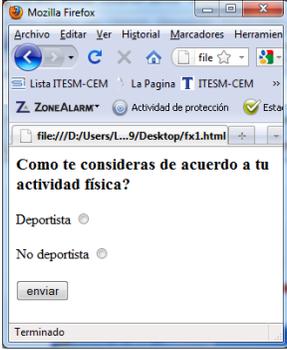


Lámina 139
Dr. Roberto Gómez Cárdenas



## <textarea>

---

- Crea un recuadro donde el usuario puede introducir texto.
- Permite obtener un “text box” con “scroll”
- Atributos
  - **name**: cuando la forma sea enviada, esta es la información que se enviará como encabezado.
  - **cols** longitud del cuadro
  - **rows** altura del cuadro

■ Atributo opcional
■ Atributo obligatorio

Lámina 140
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

**<h3>Curso de XHTML</h3>**  
**<textarea name="comments" cols="40" rows="10">**  
**Introduzca sus comentarios.**  
**</textarea>**

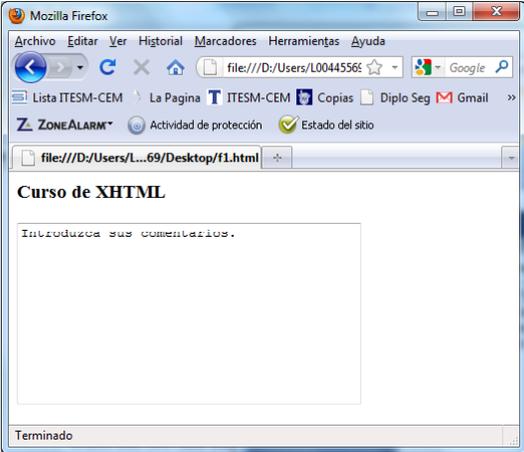


Lámina 141
Dr. Roberto Gómez Cárdenas



## <select>

---

- Se usa para agrupar una lista de opciones <option> formando así una lista de selecciones.
- También se le conoce como “Drop-Down List”
- El usuario puede escoger una o mas opciones de entre la lista predeterminada.
- Atributos:
  - name
  - size
  - multiple

Lámina 142
Dr. Roberto Gómez Cárdenas



## <option>

---

- Configura las opciones de una lista “Select”
- Atributos:
  - value
  - selected

Lámina 143
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<h2>Seleccione las tecnologías que utiliza:</h2>
<form method="post" action="http://10.17.40.23/servlet/eco">
<select name="Tecnologias" multiple="multiple" size="3">
  <option>ASP.NET</option>
  <option>C#</option>
  <option>VB.NET</option>
  <option>XPath</option>
  <option>XSL</option>
  <option>XQuery</option>
</select><br /><br />
<input type="submit" value="enviar">
</form>

```



Lámina 144
Dr. Roberto Gómez Cárdenas



## <input> Submit Button

- Se usa para enviar la forma para ser procesada
- Dispara la acción registrada en la forma <form> provocando que el browser envíe los datos es decir pares (name , value) de cada elemento de la forma hacia el servidor de web.
- Atributos:
  - type=“submit”
  - name
  - value

Lámina 145 Dr. Roberto Gómez Cárdenas



## <input> Reset Button

- Se usa para limpiar todos los campos de la forma o asignarles sus valores por default
- Atributos:
  - type=“reset”
  - name
  - value

Lámina 146 Dr. Roberto Gómez Cárdenas



## <input> Button

---

- No hay una acción por default cuando el botón se oprime.
- Usualmente, presionando el botón se invoca una función en JavaScript
- Atributos:
  - type="button"
  - name
  - value

Lámina 147
Dr. Roberto Gómez Cárdenas



## Ejemplo

---

```

<h2>Proporciona los siguientes datos</h2>
<form method="get" action="http://10.17.40.23/servlet/eco">
Nombre
<input type="text" name="nombre" />
<br /><br />
Mayor de edad?
<input type="checkbox" name="mayor" />
<br /><br />
<input type="reset" value="borrar" />
<input type="submit" value="enviar" />
</form>

```



Lámina 148
Dr. Roberto Gómez Cárdenas



## <label>

- Es una etiqueta asociada a un control.
- Se puede operar el control pulsando el texto

```
<label for="email">Email: </label>  
<input type="text" name="email" id="email" />
```

Lámina 149 Dr. Roberto Gómez Cárdenas



## <fieldset>

- Se usa para agrupar visualmente en la pantalla un conjunto de elementos de una forma
  - La etiqueta **<legend>**
  - Se usa para crear una etiqueta de texto dentro del fieldset

Lámina 150 Dr. Roberto Gómez Cárdenas



## Ejemplo

```

<h2>Proporciona los siguientes datos</h2>
<form method="get" action="http://10.17.40.23/servlet/eco">
<label for="nombre">Nombre: </label>
<input type="text" name="nombre" id="nombre" />
<br /><br />
<label for="mayor">Mayor de edad </label>
<input type="checkbox" name="mayor"
      id="mayor"/>
<br /><br />
<input type="reset" value="borrar" />
<input type="submit" value="enviar" />
</form>

```



Lámina 151
Dr. Roberto Gómez Cárdenas



## Introducción a la programación web

Roberto Gómez Cárdenas  
<http://homepage.cem.itesm.mx/rogomez>  
 rogomez@itesm.mx

Lámina 152
Dr. Roberto Gómez Cárdenas



<fieldset>

- Agrupa controles en un formulario. Esto permite a los autores agrupar temáticamente los controles de un formulario, haciéndolo más fácil de entender y rellenar para el usuario.

```
<fieldset>
<legend>Health information:</legend>
<form action="">
Height <input type="text" size="3" />
Weight <input type="text" size="3" />
</form>
</fieldset>
```

Lámina 154

Dr. Roberto Gómez Cárdenas



## Password Field

- Sintaxis: `<input type="password" />`
- Define un campo de contraseña
  - Lo que el usuario introduzca no se ve
- Ejemplo

```
<form>
Password: <input type="password" name="pwd" />
</form>
```
- Salida

Password:

Lámina 155 Dr. Roberto Gómez Cárdenas



## Radio Buttons

- Sintaxis: `<input type="radio"/>`
- Define un botón de opción
  - Permite que el usuario seleccione solo una de un número limitado de opciones.
- Ejemplo

```
<form>
<input type="radio" name="sexo" value="Masculino" /> Masculino<br />
<input type="radio" name="sexo" value="Femenino" /> Femenino
</form>
```
- Salida ejemplo

Masculino

Femenino

Lámina 156 Dr. Roberto Gómez Cárdenas



## Checkboxes

---

- Sintaxis: `<input type="radio"/>`
- Define un botón de opción
- Código
 

Rojo  
 Naranja  
 Azul  
 Verde  
 Amarillo
- Salida

```

<form>
<input type="checkbox" name="color" value="Rojo" /> Rojo<br />
<input type="checkbox" name="color" value="Naranja" /> Naranja<br />
<input type="checkbox" name="color" value="Azul" /> Azul<br />
<input type="checkbox" name="color" value="Verde" /> Verde<br />
<input type="checkbox" name="color" value="Amarillo" /> Amarillo
</form>

```

Lámina 157
Dr. Roberto Gómez Cárdenas



## Submit button

---

- Código
- Salida

Usuario:

```

<form name="input" action="html_form_action.asp" method="get">
Usuario: <input type="text" name="user" />
<input type="submit" value="Submit" />
</form>

```

Lámina 158
Dr. Roberto Gómez Cárdenas