

# Cuestionario básico sobre “Programación en Internet”

Sergio Luján Mora  
Jaume Aragonés Ferrero



# Índice general

Índice general	III
Índice de figuras	VII
Índice de cuadros	IX
Índice de acrónimos	XI
<b>1. Introducción</b>	<b>1</b>
1.1. Introducción . . . . .	3
1.2. Descripción de la asignatura “Programación en Internet” . . . . .	3
1.3. Temario de la asignatura . . . . .	4
1.4. Estructura del libro . . . . .	5
1.5. Convenciones tipográficas . . . . .	6
<b>2. Exámenes sin solución</b>	<b>9</b>
2.1. Febrero de 2002 . . . . .	11
2.2. Septiembre de 2002 . . . . .	17
2.3. Diciembre de 2002 . . . . .	23
2.4. Examen extra . . . . .	29
<b>3. Exámenes con solución</b>	<b>35</b>
3.1. Febrero de 2002 . . . . .	37
3.2. Septiembre de 2002 . . . . .	43
3.3. Diciembre de 2002 . . . . .	49
3.4. Examen extra . . . . .	55
<b>4. Explicaciones</b>	<b>61</b>
4.1. Internet . . . . .	63
4.1.1. Explicación 1 . . . . .	63
4.1.2. Explicación 2 . . . . .	63

---

4.1.3.	Explicación 3 . . . . .	64
4.1.4.	Explicación 4 . . . . .	65
4.1.5.	Explicación 5 . . . . .	65
4.1.6.	Explicación 6 . . . . .	66
4.1.7.	Explicación 7 . . . . .	67
4.1.8.	Explicación 8 . . . . .	68
4.2.	Arquitecturas cliente/servidor . . . . .	69
4.2.1.	Explicación 9 . . . . .	69
4.3.	Aplicaciones web . . . . .	69
4.3.1.	Explicación 10 . . . . .	69
4.3.2.	Explicación 11 . . . . .	70
4.3.3.	Explicación 12 . . . . .	71
4.3.4.	Explicación 13 . . . . .	71
4.4.	HTML . . . . .	72
4.4.1.	Explicación 14 . . . . .	72
4.4.2.	Explicación 15 . . . . .	73
4.4.3.	Explicación 16 . . . . .	73
4.4.4.	Explicación 17 . . . . .	73
4.4.5.	Explicación 18 . . . . .	74
4.4.6.	Explicación 19 . . . . .	74
4.4.7.	Explicación 20 . . . . .	74
4.4.8.	Explicación 21 . . . . .	75
4.4.9.	Explicación 22 . . . . .	75
4.4.10.	Explicación 23 . . . . .	76
4.4.11.	Explicación 24 . . . . .	76
4.4.12.	Explicación 25 . . . . .	77
4.4.13.	Explicación 26 . . . . .	77
4.4.14.	Explicación 27 . . . . .	78
4.4.15.	Explicación 28 . . . . .	79
4.4.16.	Explicación 29 . . . . .	81
4.5.	JavaScript . . . . .	83
4.5.1.	Explicación 30 . . . . .	83
4.5.2.	Explicación 31 . . . . .	84
4.5.3.	Explicación 32 . . . . .	85
4.5.4.	Explicación 33 . . . . .	85
4.5.5.	Explicación 34 . . . . .	85
4.5.6.	Explicación 35 . . . . .	86
4.5.7.	Explicación 36 . . . . .	87
4.5.8.	Explicación 37 . . . . .	87
4.6.	VBScript . . . . .	89
4.6.1.	Explicación 38 . . . . .	89
4.6.2.	Explicación 39 . . . . .	89

---

4.7. CGI . . . . .	90
4.7.1. Explicación 40 . . . . .	90
4.7.2. Explicación 41 . . . . .	90
4.7.3. Explicación 42 . . . . .	90
4.7.4. Explicación 43 . . . . .	91
4.8. SSI . . . . .	93
4.8.1. Explicación 44 . . . . .	93
4.8.2. Explicación 45 . . . . .	94
4.8.3. Explicación 46 . . . . .	95
4.9. ASP . . . . .	96
4.9.1. Explicación 47 . . . . .	96
4.9.2. Explicación 48 . . . . .	97
4.9.3. Explicación 49 . . . . .	97
4.9.4. Explicación 50 . . . . .	98
4.9.5. Explicación 51 . . . . .	99
4.9.6. Explicación 52 . . . . .	99
4.9.7. Explicación 53 . . . . .	101
4.9.8. Explicación 54 . . . . .	101
4.9.9. Explicación 55 . . . . .	102
4.9.10. Explicación 56 . . . . .	102
4.9.11. Explicación 57 . . . . .	103
4.9.12. Explicación 58 . . . . .	103
4.9.13. Explicación 59 . . . . .	104
4.9.14. Explicación 60 . . . . .	104
4.9.15. Explicación 61 . . . . .	104
4.10. Java . . . . .	105
4.10.1. Explicación 62 . . . . .	105
4.10.2. Explicación 63 . . . . .	105
4.11. JSP . . . . .	106
4.11.1. Explicación 64 . . . . .	106
4.11.2. Explicación 65 . . . . .	108
4.11.3. Explicación 66 . . . . .	109
4.11.4. Explicación 67 . . . . .	109
4.11.5. Explicación 68 . . . . .	110
4.11.6. Explicación 69 . . . . .	110
<b>Bibliografía recomendada</b>	<b>113</b>
<b>Índice alfabético</b>	<b>115</b>



# Índice de figuras

4.1. Página principal del sitio web de FidoNet . . . . .	67
4.2. Modelo de referencia TCP/IP . . . . .	68
4.3. Tecnologías empleadas en el cliente y el servidor web . . . . .	70
4.4. Ejemplo de tabla con color de fondo . . . . .	77
4.5. Ejemplo de lista ordenada y no ordenada . . . . .	78
4.6. Alineamiento del contenido de una tabla . . . . .	81
4.7. Áreas de texto de distinto tamaño . . . . .	83
4.8. Ventana de alerta . . . . .	84
4.9. Ventana de confirmación . . . . .	84
4.10. Ventana de solicitud de datos . . . . .	85
4.11. Configuración de parámetros de ASP en Microsoft Internet Information Server . . . . .	98
4.12. Página de error al solicitar el fichero Global.asa . . . . .	100
4.13. Ciclo de vida de una página JSP y su correspondiente servlet . . . . .	107





# Índice de cuadros

4.1. Opciones de envío con el protocolo mailto: . . . . .	75
4.2. Caracteres especiales . . . . .	86
4.3. Colecciones, eventos, métodos y propiedades de los objetos Application, Server y Session . . . . .	102



# Índice de acrónimos

## **API** *Application Program Interface*

Interfaz de programación de aplicaciones. Conjunto de constantes, funciones y protocolos que permiten programar aplicaciones. Una buena **API** facilita la tarea de desarrollar aplicaciones, ya que facilita todas las piezas y el programador sólo tiene que unirlas para lograr el fin que desea.

## **ARPA** *Advanced Research Projects Agency*

Agencia de Proyectos de Investigación Avanzados. Agencia creada por el Departamento de Defensa de los Estados Unidos de Norteamérica en 1958. También conocida como **DARPA**. A lo largo de los años ha cambiado su nombre varias veces: en 1971 **DARPA**, en 1993 **ARPA** y en 1996 **DARPA** otra vez. El proyecto más conocido de los desarrollados por esta agencia es ARPANET (o ARPAnet), semilla de la actual Internet.

## **ASCII** *American Standard Code for Information Interchange*

Código binario utilizado para representar letras, números, símbolos, etc. A cada carácter se le asigna un número del 0 al 127 (7 bits). Por ejemplo, el código **ASCII** para la A mayúscula es 65. Existen códigos **ASCII** extendidos de 256 caracteres (8 bits), que permiten representar caracteres no ingleses como las vocales acentuadas o la ñe. Los caracteres de la parte superior (128 a 255) de estos códigos **ASCII** extendidos varían de uno a otro. Por ejemplo, uno de los más extendidos es ISO Latin-1 (oficialmente ISO-8859-1).

## **ASP** *Active Server Pages*

Tecnología propietaria de MICROSOFT que permite crear páginas web dinámicas en el servidor. Desarrollada con el objetivo de sustituir a la tecnología **CGI**, ofrece una serie de características que facilitan la programación de aplicaciones web. Las páginas **ASP** suelen estar programadas en *VBScript*, aunque también se pueden programar en otros lenguajes, como *JScript*.

## **BBS** *Bulletin Board System*

Una **BBS** es un centro de intercambio de mensajes electrónicos. El funcionamiento básico de una **BBS** es muy sencillo: el usuario se conecta a través de

un módem, puede revisar los mensajes dejados por otros usuarios o puede dejar sus propios mensajes.

**BMP** *Bit-map*

Formato gráfico de mapa de bits estándar en los sistemas operativos Microsoft Windows. Almacena las imágenes en un formato llamado “mapa de bits independiente del dispositivo”, que significa que el color de cada punto (*pixel*) se almacena de un modo independiente del método empleado por un dispositivo para representar el color. Existen diversos formatos: 1 bit (blanco y negro), 4 bits (16 colores), 8 bits (256 colores) y 24 bits (16 777 216 colores).

**CERN** *Conseil Européenne pour le Recherche Nucléaire*

Organización Europea para la Investigación Nuclear. Es el mayor centro científico a nivel mundial dedicado a la física de partículas. Su sede central se encuentra en Ginebra, Suiza. Fundado en 1954 por 12 países, actualmente está formado por 20 países, entre ellos España. Tim Berners-Lee, mientras trabajaba en él a principios de 1990, inventó la **WWW**.

**CFML** *ColdFusion Markup Language*

Lenguaje de etiquetas empleado para programar las páginas web dinámicas en el servidor basadas en la tecnología ColdFusion.

**CGI** *Common Gateway Interface*

Estándar que permite el intercambio de información entre un servidor y un programa externo al servidor. Un programa **CGI** es un programa preparado para recibir y enviar datos desde y hacia un servidor web según este estándar. Normalmente se programan en *C* o en *Perl*, aunque se puede usar cualquier lenguaje de propósito general.

**CSP** *Caché Server Pages*

Tecnología propietaria de INTERSYSTEMS que permite crear páginas web dinámicas en el servidor. Se diferencia de otras tecnologías similares como **ASP** y **JSP** en que la lógica de negocio reside junto con la lógica de datos en el sistema gestor de bases de datos.

**CSS** *Cascading Style Sheets*

Tecnología empleada en la creación de páginas web, que permite un mayor control sobre el lenguaje **HTML**. Permite crear hojas de estilo que definen como cada elemento, como por ejemplo los encabezados o los enlaces, se tiene que mostrar. El término “en cascada” indica que diferentes hojas de estilo se pueden aplicar sobre la misma página. **CSS** ha sido desarrollada por **W3C**.

**DARPA** *Defense Advanced Research Projects Agency*

Ver **ARPA**.

**DHTML** *Dynamic HTML*

Conjunto de extensiones a **HTML** que permiten modificar el contenido de una página web en el cliente sin necesidad de establecer una nueva comunicación con el servidor. Se basa en el uso de **DOM** para acceder al contenido de la página.

**DNS** *Domain Name System*

También conocido como *Domain Name Service*, es un servicio de Internet que traduce los nombres de dominio (por ejemplo, `http://www.ua.es`) a direcciones IP (193.145.233.8). Como los nombres de dominio son palabras, son más fáciles de recordar que las direcciones IP. Sin embargo, Internet se basa en las direcciones IP. Cada vez que se emplea un nombre de dominio, el servicio de **DNS** traduce un nombre de dominio en su correspondiente dirección IP. Si un servidor de **DNS** no sabe cómo traducir un nombre de dominio, traslada la pregunta a otro servidor de **DNS** y así sucesivamente hasta que se obtenga la dirección IP correspondiente o una respuesta de error.

**DOM** *Document Object Model*

Especificación que define como se puede acceder a los objetos de un documento **HTML** (ventanas, imágenes, formularios) a través de un lenguaje de *script*. Básicamente define una jerarquía de objetos. **DOM** se encuentra en proceso de estandarización por **W3C**. **DHTML** depende de **DOM** para cambiar dinámicamente el contenido de una página web. Desgraciadamente, los dos navegadores mayoritarios poseen distintos modelos de objetos que en algunas partes son incompatibles entre sí.

**GIF** *Graphics Interchange Format*

Formato gráfico de mapa de bits desarrollado por COMPUSERVE para su servicio de información. Sus principales características son: compresión de datos sin pérdidas (**LZW**), soporte de transparencias y de animaciones. Existen dos versiones de este estándar gráfico: 87A y 89A. Es el formato más adecuado para imágenes con pocos colores, dibujos sencillos o textos.

**GNU** *GNU is Not Unix*

**GNU** es un acrónimo recursivo de “*GNU is Not Unix*”. El proyecto **GNU** nació en 1984 de la mano de Richard Stallman en el **MIT**, con el fin de desarrollar un sistema operativo del estilo de Unix, pero totalmente gratuito. El objetivo de **GNU** es la creación de software no propietario. Este software se puede descargar, usar y modificar libremente, siempre que se cumpla la **GPL**. La principal limitación que impone esta licencia es que no se puede limitar la distribución y uso de un software basado en **GNU**.

**GPL** *GNU General Public License*

**GPL** es la licencia que se emplea con algunos tipos de software de código abierto (*open source software*), que detalla bajo que condiciones el software y su correspondiente código fuente se puede copiar, distribuir y modificar.

**HTML** *HyperText Markup Language*

Lenguaje compuesto de una serie de etiquetas o marcas que permiten definir el contenido y la apariencia de las páginas web. Aunque se basa en el estándar **SGML**, no se puede considerar que sea un subconjunto de él. Existen cientos de etiquetas con diferentes atributos. **W3C** se encarga de su estandarización. El futuro sustituto de **HTML** es **XHTML**.

**HTTP** *HyperText Transfer Protocol*

Es el protocolo que emplea la **WWW**. Define como se tienen que crear y enviar los mensajes y que acciones debe tomar el servidor y el navegador en respuesta a un comando. Es un protocolo *stateless* (sin estado), porque cada comando se ejecuta independientemente de los anteriores o de los posteriores. Actualmente, la mayoría de los servidores soportan **HTTP 1.1** (**RFC 2616** de junio de 1999). Una de las principales ventajas de esta versión es que soporta conexiones persistentes: una vez que el navegador se conecta al servidor, puede recibir múltiples ficheros a través de la misma conexión, lo que aumenta el rendimiento de la transmisión hasta en un 20 %.

**IAB** *Internet Architecture Board*

Comité del **IETF**. En el **RFC 2850** “Charter of the Internet Architecture Board (IAB)” se establece su organización y su misión. La dirección de su sitio web en Internet es la siguiente: <http://www.iab.org/>.

**IANA** *Internet Assigned Numbers Authority*

Comité del **IETF** encargado de estandarizar los distintos parámetros que se emplean en Internet, como son las direcciones **IP**, los nombres de dominio, los puertos empleados por los protocolos y otros tipos de nombres e identificadores. La dirección de su sitio web en Internet es <http://www.iana.org/>.

**IDC** *Internet Database Connector*

Conector de bases de datos de Internet. Tecnología propietaria de **MICROSOFT** que permite generar páginas web dinámicas a partir de la información almacenada en una base de datos. Es el precursor de **ASP**.

**IETF** *Internet Engineering Task Force*

Organización internacional formada por diseñadores de red, operadores de Internet y fabricantes, que vigila por la correcta evolución y funcionamiento de Internet. En el **RFC 3160** “The Tao of IETF - A Novice’s Guide to the Internet Engineering Task Force” se puede consultar su composición, funcionamiento y objetivos. La dirección de su sitio web en Internet es <http://www.ietf.org/>.

**IP** *Internet Protocol*

El protocolo **IP** especifica el formato de los paquetes (datagramas) y el esquema de direccionamiento en Internet. La versión actual de este protocolo es IPv4,

pero en la actualidad se está comenzando a implantar la nueva versión IPv6, también llamada IPng (*IP next generation*).

**ISO** *International Organization for Standards*

Organización fundada en 1946, cuyos miembros son las organizaciones nacionales de normalización (estandarización) correspondientes a los países miembros. Entre sus miembros se incluyen ANSI (Estados Unidos), BSI (Gran Bretaña), AFNOR (Francia), DIN (Alemania) y UNE (España).

**ISOC** *Internet Society*

La **ISOC** es una sociedad profesional establecida en 1992, que agrupa más de 150 organizaciones y 16.000 miembros individuales en más de 180 países. Esta organización agrupa otras organizaciones responsables de desarrollar los estándares de Internet, como la **IETF**, la **IANA** y la **IAB**. La dirección de su sitio web en Internet es <http://www.isoc.org/>.

**JPEG** *Joint Photographic Experts Group*

Nombre del comité de expertos que desarrolló el formato gráfico con el mismo nombre. Se trata de un formato gráfico de mapa de bits que incorpora compresión de datos con pérdidas y permite trabajar con 24 bits de color (color real o verdadero). El nivel de compresión es variable, por lo que se puede elegir entre mejor calidad y menor compresión o peor calidad y mayor compresión. Este formato se suele emplear con imágenes fotográficas o complejas, pero no es el adecuado para imágenes sencillas, dibujos o textos.

**JPG** *Joint Photographic Experts Group*

Ver **JPEG**.

**JSP** *Java Server Pages*

Tecnología de SUN MICROSYSTEMS que permite crear páginas web dinámicas en el servidor. Equivale a la tecnología **ASP** de MICROSOFT. Se programan en Java.

**LZW** *Lempel Ziv Welch*

Esquema de compresión sin pérdidas empleado en el formato gráfico **GIF** de COMPUSERVE. Desarrollado por J. Ziv and A. Lempel en 1977, y posteriormente mejorado por T. Welch. La patente de **LZW** la ostenta UNISYS CORPORATION. Durante muchos años, UNISYS CORPORATION permitió el uso de **LZW** sin cobrar un canon (la mayoría de la gente no sabía que había sido patentado en 1983). Sin embargo, a partir de 1995 decidió cobrar una tasa y se desató una gran controversia ya que se había extendido ampliamente su uso.

**MIME** *Multipurpose Internet Mail Extensions*

Estándar que permite la transmisión de cualquier tipo de fichero mediante correo electrónico. Además, los navegadores también emplean los tipos **MIME**

para identificar y visualizar distintos tipos de recursos que no están en formato **HTML**.

**MIT** *Massachusetts Institute of Technology*

Instituto Tecnológico de Massachusetts. Centro de investigación avanzado situado en los Estados Unidos. Famoso por su “Media Lab”, en el cual trabajan investigadores de la talla de Nicholas Negroponte o Marvin Minsky.

**MPEG** *Moving Picture Experts Group*

Nombre del comité de expertos que desarrolló el formato de vídeo digital con el mismo nombre. En realidad, se trata de un conjunto de formatos de compresión de vídeo con pérdidas (MPEG-1, MPEG-2 and MPEG-4) con diferentes resoluciones (352 x 240, 720 x 480, etc.) y velocidades de refresco.

**MPG** *Moving Picture Experts Group*

Ver **MPEG**.

**MVJ** *Máquina Virtual Java*

Entorno de ejecución independiente de la plataforma que convierte los *byte-codes* del lenguaje *Java* al lenguaje máquina de la plataforma (arquitectura de microprocesador y sistema operativo) donde se ejecuta.

**NCSA** *National Center for Supercomputing Applications*

Centro Nacional para Aplicaciones de Supercomputación. Centro creado en la Universidad de Illinois en enero de 1986. Famoso porque uno de los primeros navegadores web gratuitos, *NCSA Mosaic*, se creó en sus instalaciones.

**ODBC** *Open DataBase Connectivity*

Conectividad abierta de bases de datos. **ODBC** es un estándar *de facto* para el acceso a bases de datos en entornos cliente/servidor. El objetivo de **ODBC** es facilitar el acceso a cualquier dato desde cualquier aplicación, independientemente del sistema gestor de bases de datos empleado. Para ello, en **ODBC** se inserta una capa intermedia, llamada controlador (*driver*) de la base de datos, entre la aplicación y el sistema gestor de bases de datos. El propósito de esta capa es traducir las consultas que genera la aplicación en comandos que entienda el sistema gestor de bases de datos. Por tanto, mediante **ODBC**, se puede cambiar la parte servidor (la base de datos) sin tener que cambiar el cliente, siempre que todas las partes sean compatibles con **ODBC**.

**OSI** *Open System Interconnection*

También conocido como el Modelo de Referencia **OSI** o el Modelo **OSI**. Se trata de un estándar de **ISO** que define un marco para implementar los protocolos de red en siete capas. Los siete niveles, desde el más inferior (1) al superior (7) son: físico, enlace, red, transporte, sesión, presentación y aplicación.



**PNG** *Portable Network Graphics*

Formato gráfico de mapa de bits similar a **GIF**. **W3C** ha desarrollado este formato gráfico con la idea de sustituir **GIF** por **PNG** debido a que el primero emplea un algoritmo que está patentado, mientras que **PNG** es totalmente gratuito. No permite crear animaciones, pero sí que permite definir distintos niveles de transparencia. Al igual que **GIF**, emplea un esquema de compresión sin pérdidas que logra tasas de compresión mayores que **GIF**. Tanto Microsoft Internet Explorer como Netscape Navigator aceptan este formato, aunque no todas sus características.

**RFC** *Request for Comments*

Medio de publicar propuestas sobre Internet. Cada **RFC** recibe un número. Algunos se convierten en un estándar de Internet.

**RGB** *Red Green Blue*

Notación de los colores en la que cada color se representa como una combinación de los tres colores básicos (primarios) rojo (*red*), verde (*green*) y azul (*blue*). Se trata de un modelo aditivo (se parte del negro). Mediante la combinación adecuada de los tres colores básicos se consigue todo el espectro de colores. Además de **RGB** existen otras formas de representar los colores. Otra de las más corrientes es CMYK (*cyan, magenta, yellow, black*), que se trata de un modelo sustractivo.

**RTB** *Red de Telefonía Básica*

También llamada Red de Telefonía Conmutada. Es el servicio que los operadores de telefonía dan a los usuarios a través de plataformas terrestres y cableadas hasta el punto final.

**SGML** *Standard Generalized Markup Language*

Lenguaje que permite organizar y etiquetar los distintos elementos que componen un documento. Se emplea para manejar grandes documentos que sufren constantes revisiones y se imprimen en distintos formatos e idiomas. Desarrollado y estandarizado por **ISO** en 1986.

**SSI** *Server Side Include*

Directivas de inclusión del servidor. Comandos que se incluyen en una página **HTML** y que son ejecutados por el servidor web antes de transmitir la página al cliente. Permite generar páginas web dinámicas.

**TCP/IP** *Transmission Control Protocol/Internet Protocol*

Familia de protocolos que se emplean en las comunicaciones de Internet.

**TIFF** *Tagged Image File Format*

Formato gráfico de mapa de bits muy popular tanto en entorno Microsoft Windows como Apple Macintosh. Permite almacenar imágenes de cualquier resolución, en blanco y negro, escala de grises o color.

**URL** *Universal Resource Locator*

También conocido como *Uniform Resource Locator*. Sistema de direccionamiento de máquinas y recursos en Internet. Es decir, se trata de una dirección que permite localizar cualquier máquina o documento que se encuentre accesible a través de Internet.

**VRML** *Virtual Reality Modeling Language*

Lenguaje de Modelado de Realidad Virtual. Lenguaje para crear objetos en tres dimensiones en la Web. Los ficheros creados con este lenguaje poseen la extensión `.wrl` (de *world*) y para visualizarlos es necesario emplear un visor adecuado o que el navegador web disponga del correspondiente *plug-in*.

**W3C** *World Wide Web Consortium*

Consortio internacional de compañías y organizaciones involucradas en el desarrollo de Internet y en especial de la **WWW**. Su propósito es desarrollar estándares y “poner orden” en Internet.

**WWW** *World Wide Web*

También conocida como “la Web” o “la Red”. Sistema mundial de servidores web conectados a Internet (no todos los ordenadores conectados a Internet forman parte de la **WWW**). Su protocolo de comunicación es **HTTP**, su lenguaje de creación de documentos **HTML** y su sistema de direccionamiento de los recursos **URL**. Los navegadores web (*browsers*) permiten navegar por la web.

**XHTML** *eXtensible HyperText Markup Language*

**HTML** escrito según las normas que marca **XML**. Por tanto, se trata de una aplicación concreta de **XML** y no tienen que confundirse entre sí.

**XML** *Extensible Markup Language*

Metalenguaje de etiquetado basado en **SGML**. Diseñado específicamente para la **WWW** por **W3C**. Permite que un usuario diseñe sus propias etiquetas, con sus atributos y las reglas de construcción de documentos (sintaxis).

# Capítulo 1

## Introducción

En este capítulo se realiza una introducción del libro y se presenta el contenido de cada uno de los capítulos. Además, también se comentan las convenciones tipográficas empleadas para distinguir los acrónimos, nombres de programas, etc.



## 1.1. Introducción

Desde el curso 2001-2002, se imparte la asignatura “Programación en Internet” en la Universidad de Alicante. Esta asignatura pertenece al plan de estudios 2001 de las titulaciones de Ingeniería en Informática, Ingeniería Técnica en Informática de Sistemas e Ingeniería Técnica en Informática de Gestión de la Universidad de Alicante. En la Universidad de Alicante, esta asignatura la imparte el Departamento de Lenguajes y Sistemas Informáticos, adscrito a la Escuela Politécnica Superior de dicha Universidad.

Este libro trata de satisfacer el deseo de los alumnos de “Programación en Internet” de consultar las pruebas de evaluación de cursos anteriores, que emplean como medio de preparación a las pruebas de evaluación de conocimientos a las que se tendrán que enfrentar al final de la asignatura. Para ello, este libro contiene los tres exámenes de tipo test que se plantearon a lo largo del curso 2001-2002 (convocatorias de febrero, septiembre y diciembre). Además, se incluye un examen extra formado por preguntas de los otros tres exámenes. Cada uno de los exámenes incluye su correspondiente solución, que además incluye una explicación de las preguntas y respuestas.

## 1.2. Descripción de la asignatura “Programación en Internet”

La asignatura “Programación en Internet” es optativa y de duración cuatrimestral, con una carga docente de 6 créditos, repartidos entre 3 de teoría y 3 de prácticas. La descripción oficial de la asignatura, publicada en el Boletín Oficial del Estado número 230 de 25 de septiembre de 2001 es:

- Desarrollo y programación de sistemas de acceso a bases de datos de Internet.
- Planificación, diseño y administración de sitios Web.
- Migración de aplicaciones a entornos en Internet.
- Herramientas de desarrollo.
- Diseño y programación de elementos multimedia en Internet.

La asignatura “Programación en Internet” no posee prerequisites, pero sí las siguientes recomendaciones oficiales (no es necesario haber aprobado las siguientes asignaturas, pero sí recomendable haberlas cursado previamente):

- Fundamentos de Programación I (1er curso).
- Fundamentos de Programación II (1er curso).

- Bases de Datos I (2º curso).

Los objetivos principales de “Programación en Internet” son:

- Que el alumno conozca las características principales de las tecnologías empleadas en el desarrollo de aplicaciones web.
- Que el alumno conozca la estructura y funcionamiento de una aplicación web.
- Que el alumno adquiera los conocimientos y habilidades necesarios para programar aplicaciones destinadas a ser usadas en entornos Internet.
- Que el alumno conozca los recursos específicos (hardware y software) necesarios para poner en producción aplicaciones web.

Además, no se trata sólo de aprender habilidades técnicas, sino de dotar de conocimientos de fondo para formar profesionales flexibles capaces de trabajar con cualquier tecnología con una curva de aprendizaje mínima. Las tecnologías que se emplean en Internet están en continua evolución, por lo que no tiene sentido especializarse en una única tecnología, sino ofrecer una visión más amplia del estado actual de las tecnologías de desarrollo en Internet.

### 1.3. Temario de la asignatura

La asignatura está dividida en tres módulos principales que tratan aspectos generales de la programación en Internet, la programación de la parte cliente y la programación de la parte servidor:

- Módulo I: Introducción a la programación en Internet
  - Presentación y objetivos de la asignatura
  - Historia de Internet
  - Arquitecturas cliente/servidor
  - ¿Qué es una aplicación web?
  - Estructura de un sitio web
- Módulo II: Programación de clientes web
  - HTML
  - Guía de estilo
  - Lenguajes de script
  - JavaScript

- Modelo de objetos de documento
- Módulo III: Programación de servidores web
  - CGI
  - SSI
  - IDC
  - Conceptos comunes de las tecnologías de programación de servidor web: ASP, ColdFusion, JSP y PHP
  - Active Server Pages (ASP)
  - Java Server Pages (JSP)
  - Otras tecnologías: PHP, ColdFusion y Caché

## 1.4. Estructura del libro

Este libro se compone de 4 capítulos, un apéndice con bibliografía recomendada y varios índices (figuras, acrónimos, etc.) que facilitan la búsqueda de información.

En el Capítulo 2 (**Exámenes sin solución**), se presentan los exámenes sin solución empleados durante el curso 2001-2002, más un examen extra formado por preguntas de los otros tres.

En el Capítulo 3 (**Exámenes con solución**), aparecen los mismos exámenes presentados en el Capítulo 2, pero con la respuesta correcta indicada. Además, aparece una referencia a la explicación de la pregunta que aparece en el Capítulo 4.

En el Capítulo 4 (**Explicaciones**), se explica la solución de cada una de las preguntas de los exámenes presentados. Aquellas preguntas que tratan un tema similar aparecen relacionadas entre sí. Además, las explicaciones aparecen ordenadas según el tema que tratan. Las preguntas planteadas para cada tema son:

- Internet (11): 6, 10, 26, 29, 35, 51, 52, 53, 77, 78, 95.
- Arquitecturas cliente/servidor (1): 1.
- Aplicaciones web (7): 5, 9, 14, 42, 54, 55, 80, 91.
- HTML (24): 2, 19, 20, 28, 30, 32, 36, 38, 39, 41, 43, 45, 48, 56, 57, 58, 59, 60, 82, 84, 86, 89, 93, 96.
- JavaScript (10): 15, 21, 33, 49, 61, 62, 63, 64, 88, 98.
- VBScript (2): 24, 67.
- CGI (5): 12, 23, 40, 65, 90.

- SSI (5): 17, 44, 66, 81, 100.
- ASP (22): 3, 7, 13, 16, 18, 22, 25, 31, 37, 47, 50, 68, 69, 70, 71, 72, 76, 79, 83, 87, 92, 97.
- Java (3): 4, 34, 75.
- JSP (9): 8, 11, 27, 46, 73, 74, 85, 94, 99.

## 1.5. Convenciones tipográficas

Con el fin de mejorar la legibilidad del texto, distintas convenciones tipográficas se han empleado a lo largo de todo el libro.

Los ejemplos, que normalmente están completos y por tanto se pueden escribir y probar, aparecen destacados de la siguiente forma (el texto de los ejemplos emplea un tipo de letra de paso fijo como Courier):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
Cabecera de la página
</HEAD>
<BODY>
Cuerpo de la página
</BODY>
</HTML>
```

Los estilos empleados a lo largo del texto son:

- Los acrónimos y siglas que figuran en el índice de acrónimos aparecen siempre destacados en **negrita**. Ejemplo: **ASP**, **HTML**, **URL**, etc.
- Cuando un acrónimo aparece por primera vez, se muestra el nombre completo en *cursiva* y entre paréntesis y en **negrita** el acrónimo. Ejemplo: *Graphics Interchange Format (GIF)*, *World Wide Web (WWW)*, etc.
- Las palabras no escritas en castellano aparecen destacadas en *cursiva*. Ejemplo: *bookmarks*, *browser*, *plug-in*, etc.
- Cuando se hace referencia a un programa, el texto de los interfaces que se ven en pantalla aparece en **negrita**. Además, se emplea una flecha para indicar una secuencia de acciones o pulsaciones en un programa. Ejemplo: **Aceptar**, **Inicio** → **Programas** → **Accesorios**.



- 
- Los nombres de las compañías se muestran con un tipo de letra de MAYÚSCULAS PEQUEÑAS. Ejemplo: MICROSOFT, NETSCAPE, etc.
  - Los nombres de los programas se muestran con un tipo de letra sin palo (sans serif). Ejemplo: Microsoft Paint, Netscape Navigator, Opera, etc.
  - Los lenguajes informáticos se muestran con un tipo de letra *inclinada*. Ejemplo: *C*, *Java*, *Perl*, etc.
  - Las extensiones de los ficheros, las palabras clave de los lenguajes de programación y el código incluido dentro del texto se muestra con un tipo de letra de paso fijo como Courier. Ejemplo: `.html`, `<IMG>`, `var ciudad = "Elche"`, etc.



## Capítulo 2

# Exámenes sin solución

En este capítulo se presenta una serie de exámenes sin solución realizados en la asignatura “Programación en Internet” de la Universidad de Alicante durante el curso académico 2001-2002. El último examen no se ha empleado en ningún examen real y se basa en preguntas de los otros tres exámenes.

Cada examen se compone de 25 preguntas. Las preguntas aparecen tal como aparecieron en los exámenes y no poseen ningún orden, ya que así es como se presentaron a los alumnos.

Cada pregunta posee dos numeraciones:

- La primera indica el número de orden de la pregunta dentro del examen. Para cada examen, esta numeración empieza desde 1.
- La segunda numeración es única para todos los exámenes e indica la posición de la pregunta en el conjunto de todas las preguntas.

Todas las preguntas poseen cuatro respuestas, de las cuales únicamente una es la correcta.



## 2.1. Febrero de 2002

1. (1) En las arquitecturas cliente/servidor:
  - a) La parte cliente se conoce como front-end y la servidor como back-end.
  - b) La parte cliente se conoce como back-end y la servidor como front-end.
  - c) La parte cliente se conoce como display-end y la servidor como process-end.
  - d) Las anteriores respuestas no son correctas.
2. (2) Respecto a XHTML, señala cuál de las siguientes afirmaciones es falsa:
  - a) Las etiquetas y atributos tienen que escribirse siempre en minúsculas.
  - b) El valor de los atributos tiene que ir entre comillas.
  - c) No se admiten atributos sin valor.
  - d) No existen etiquetas vacías.
3. (3) En la siguiente petición, “pag.asp?id=123”, ¿cómo obtenemos el valor de “id” en la página ASP “pag.asp”?:
  - a) `Request.Form("id")`.
  - b) `Request.QueryString("id")`.
  - c) `Request("id")`.
  - d) La b) y la c) son correctas.
4. (4) De las siguientes afirmaciones sobre Java, cuál no es cierta:
  - a) El propósito inicial fue el desarrollo de aplicaciones en Internet.
  - b) Es un lenguaje multiplataforma.
  - c) Es un lenguaje fuertemente tipado.
  - d) Incorpora un recolector de basura.
5. (5) De las siguientes tecnologías, ¿cuál se emplea para programar un cliente web?:
  - a) CGI.
  - b) ASP.
  - c) HTML.
  - d) ColdFusion.
6. (6) El sistema que se emplea en Internet para traducir un nombre de dominio en dirección IP se llama:

- a) Name Resolution System.
  - b) Direct Name System.
  - c) Domain System Name.
  - d) Domain Name System.
7. (7) Respecto a la petición “`file:///localhost/p.asp`” ejecutada en un ordenador que posee el servidor Microsoft Personal Web Server:
- a) El código ASP de la página se ejecuta si el servidor web está iniciado.
  - b) El código ASP de la página no se ejecuta.
  - c) El código ASP de la página no se ejecuta si el servidor web no está iniciado.
  - d) La a) y la c) son correctas.
8. (8) Cuando un servicio web recibe una petición sobre una página JSP:
- a) Se interpreta el código de la página JSP y se devuelve una respuesta.
  - b) Se ejecuta directamente el servlet correspondiente a la página JSP.
  - c) Se busca la página JSP en la caché y se interpreta su código.
  - d) Las anteriores respuestas no son correctas.
9. (9) De las siguientes tecnologías, ¿cuál no se emplea para programar un servidor web?:
- a) Servlets.
  - b) Applets.
  - c) SSI.
  - d) ColdFusion.
10. (10) El protocolo HTTP fue inventado por:
- a) Tim Berners-Lee.
  - b) Marc Andresseen.
  - c) Linus Torvalds.
  - d) Steve Jobs.
11. (11) Respecto a la programación con JSP, señala cuál no es cierta:
- a) Las expresiones son simples volcados de datos a la página.
  - b) Las acciones únicamente tienen sintaxis basada en XML.
  - c) Es necesario hacer una operación de conversión (cast) cuando en una expresión se devuelvan datos que no sean de tipo **String**.

- 
- d) Las directivas tienen como propósito enviar instrucciones al motor de JSP.
12. (12) Un programa CGI:
- a) Sólo puede recibir información por la entrada estándar.
  - b) Se tiene que programar en un lenguaje compilado.
  - c) Mantiene automáticamente el estado (sesión) entre una conexión y otra.
  - d) Las anteriores respuestas no son correctas.
13. (13) El tiempo de vida de una sesión en ASP se puede configurar con:
- a) `Session.Timeout`.
  - b) `Session.SetTimeout`.
  - c) `Session.ScriptTimeout`.
  - d) Las anteriores respuestas no son correctas.
14. (14) Una Intranet es:
- a) Una red global que conecta millones de ordenadores por todo el mundo.
  - b) Una red privada basada en TCP/IP.
  - c) Una red pública basada en TCP/IP.
  - d) Las anteriores respuestas no son correctas.
15. (15) JavaScript es un lenguaje de programación:
- a) Orientado a objetos.
  - b) Basado en objetos.
  - c) Sin objetos.
  - d) Las anteriores respuestas no son correctas.
16. (16) Desde una página ASP, para enviar información al cliente se emplea el objeto:
- a) `Request`.
  - b) `Response`.
  - c) `Session`.
  - d) `Sender`.
17. (17) En SSI, para insertar en una página la fecha de la última modificación de un archivo se emplea:
- a) `<!-- #flastmod file="archivo" -->`.

- b) `<!-- #flastmod virtual="archivo" -->`.
- c) `<!-- #flastmod src="archivo" -->`.
- d) La a) y la b) son correctas.
18. (18) Respecto al fichero `Global.asa`, señala cuál de las siguientes afirmaciones es falsa:
- a) En él se declaran los eventos de inicio y finalización de los objetos `Application` y `Session`.
- b) Es un fichero obligatorio en cualquier aplicación web realizada con ASP.
- c) No es accesible desde el cliente.
- d) El código incluido en él no puede escribir datos en la página devuelta.
19. (19) El esqueleto básico de una página HTML es:
- a) `<HTML>`, `<HEAD>`, `<BODY>`, `</BODY>`, `</HTML>`.
- b) `<HTML>`, `<HEAD>`, `<BODY>`, `</HEAD>`, `</BODY>`.
- c) `<HTML>`, `<HEAD>`, `<BODY>`, `</BODY>`, `</HEAD>`.
- d) Las anteriores respuestas no son correctas.
20. (20) La etiqueta `<P> . . . </P>` se emplea para:
- a) Marcar párrafos de texto.
- b) Destacar el texto con un formato especial.
- c) Aumentar la sangría del texto.
- d) Las anteriores respuestas no son correctas.
21. (21) En JavaScript, si queremos mostrar un mensaje de alerta, usaremos:
- a) `window.alert("mensaje")`.
- b) `window.alert = "mensaje"`.
- c) `document.alert("mensaje")`.
- d) `windows.alert("mensaje")`.
22. (22) Desde una página ASP se puede acceder:
- a) Al sistema de archivos en el servidor.
- b) Al sistema de archivos en el cliente.
- c) A ambos sistemas de archivos.
- d) Las anteriores respuestas no son correctas.



- 
23. (23) Para obtener la información que un cliente web envía con una URL al servidor web, un programa CGI:
- a) Consulta la variable de entorno `QUERY_STRING`.
  - b) Consulta la variable de entorno `CONTENT`.
  - c) Consulta la variable de entorno `PATH_INFO`.
  - d) La a) y la c) son correctas.
24. (24) En VBScript, a la hora de concatenar cadenas, el mejor operador es:
- a) `“+”`.
  - b) `“&”`.
  - c) `“.”`.
  - d) No existe un operador, se tiene que usar una función.
25. (25) Cuando se programa una página ASP:
- a) VBScript es el único lenguaje que se puede emplear.
  - b) JavaScript es el único lenguaje que se puede emplear.
  - c) En la instalación por defecto (estándar) se pueden emplear tanto VBScript como JavaScript.
  - d) Las anteriores respuestas no son correctas.