

Aplicaciones Web

Javier Zofío Jiménez





Unidad 1 - Evolución de las aplicaciones web	6
1 >> Aplicaciones web	7
2 >> Desarrollo de las aplicaciones web	8
2.1 > ¿Qué es la Web 2.0?	9
2.2 > Hacia la Web 3.0	10
3 >> Servicios web	11
3.1 > Esquema de funcionamiento de un servicio web	11
3.2 > Instalación de un servidor web, un servidor de bases de datos y un intérprete de PHP	12
 Unidad 2 - Lenguajes de marcas, hojas de estilos y scripts	 18
1 >> Lenguajes de marcas, HTML5	19
1.1 > Definiciones básicas	19
1.2 > Estándares y versiones	21
1.3 > Normas fundamentales	22
1.4 > Estructuración básica	23
1.5 > Colores	26
1.6 > Espaciados, saltos de línea y párrafos	26
1.7 > Cabeceras	28
1.8 > Semántica en textos	29
1.9 > Listas de elementos	30
1.10 > Imágenes	31
1.11 > Vínculos o hiperenlaces	32
1.12 > Tablas	34
1.13 > Formularios	35
2 >> Hojas de estilos CSS	40
2.1 > Introducción	40
2.2 > Hoja de estilos	40
2.3 > Formas de aplicar estilo	41
2.4 > Conceptos fundamentales	42
2.5 > Criterios de selección, selectores	44
2.6 > Unión y combinación de selectores	46
2.7 > Elementos HTML <div> y 	46
2.8 > Colores, fondos, textos y fuentes	49
2.9 > Bloques, capas o cajas	50
2.10 > Tamaño y posicionamiento	51



3 >> Lenguajes de script de navegador	53
3.1 > Introducción a JavaScript	53
3.2 > Sintaxis y características de JavaScript	54
3.3 > Variables, operadores y estructuras básicas de control	55
3.4 > Ventanas emergentes	57

Unidad 3 - Herramientas de comunicación y colaboración **62**

1 >> Herramientas de comunicación y colaboración	63
1.1 > Sindicación web	63
1.2 > Marcadores sociales	64
1.3 > Blogs	65
1.4 > Wikis	68
1.5 > Foros	69
2 >> Comparativa de blogs, wikis y foros	73
2.1 > Wikis y blogs	73
2.2 > Wikis y foros	74
3 >> Redes sociales	75

Unidad 4 - Servicios de gestión de archivos en la nube **82**

1 >> Gestión de archivos en la nube	83
2 >> Servicios de almacenamiento en la nube	85
2.1 > Sincronización de ficheros en la nube	85
2.2 > Almacena y escucha música en la nube	90
2.3 > Gestión de fotos en la nube	92
2.4 > Gestión de vídeos en la nube	95
3 >> Integración de servicios en la nube	96

Unidad 5 - Aplicaciones ofimáticas en la nube **104**

1 >> Aplicaciones web ofimáticas	105
2 >> Gestión de documentos con Google Drive	106
2.1 > Entorno de trabajo Google Drive	106
2.2 > Creando archivos en Google Drive	107
2.3 > Subir archivos en Google Drive	111
2.4 > Formularios	112
2.5 > Dibujos	119



3 >> Gestión de documentos con Microsoft SkyDrive	121
3.1 > Empezar a trabajar con Microsoft SkyDrive	121
3.2 > Crear archivos en Microsoft SkyDrive	124
3.3 > Compartir un documento en Microsoft SkyDrive	135
3.4 > Cargar un archivo en Microsoft SkyDrive	139
4 >> Escritorio en la nube: EyeOS	141
Unidad 6 - Sistemas gestores de contenidos	146
1 >> Introducción	147
1.1 > Características	147
1.2 > Tipos y licencias	148
2 >> Instalación del CMS Joomla!	149
2.1 > Requisitos de instalación	149
2.2 > Versiones Joomla!	149
3 >> Interfaz del CMS Joomla!	153
3.1 > Interfaz frontal, <i>frontend</i>	153
3.2 > Interfaz de administración, <i>backend</i>	153
3.3 > Configuración en la interfaz de administrador	156
4 >> Gestión de usuarios en Joomla!	162
4.1 > Niveles de acceso	162
4.2 > Grupos de usuarios	163
4.3 > Usuarios Joomla!	165
4.4 > Registro de usuarios	166
5 >> Gestión de contenidos en Joomla!	170
5.1 > Categorías	170
5.2 > Artículos	171
6 >> Gestión de menús en Joomla!	172
7 >> Extensiones Joomla!	175
7.1 > <i>Gestor de Extensiones</i>	175
7.2 > Módulos	177
7.3 > <i>Plugins</i>	179
7.4 > Plantillas	185
7.5 > Idiomas	187
8 >> Componentes Joomla!	190
8.1 > Actualizar Joomla!	190
8.2 > <i>Banners</i>	191
8.3 > Buscar	192

8.4 > Búsqueda Inteligente	192
8.5 > Contactos	192
8.6 > Enlaces	192
8.7 > Mensajería	193
8.8 > Noticias Externas	193
8.9 > Redireccionar	193
9 >> Copias de seguridad Joomla!	194
9.1 > Copiar la carpeta principal	194
9.2 > Exportar con phpMyadmin	194
9.3 > Componente Akeeba Backup	195

Unidad 7 - Sistemas gestores de aprendizaje electrónico **204**

1 >> Introducción	205
2 >> Sistemas de gestión de contenidos de aprendizaje	206
3 >> Instalación del LCMS Moodle	208
3.1 > Requisitos de instalación y versiones	208
4 >> Estructura, entorno y configuración	212
4.1 > Estructura del sitio y jerarquía de directorios	212
4.2 > Personalizar el entorno	212
4.3 > Ajustes de configuración	213
4.4 > Navegación y edición	215
4.5 > Agregar bloques	215
5 >> Elementos lógicos: comunicación, materiales y actividades	217
6 >> Modos de registro	218
7 >> Gestión de usuarios	219
7.1 > Creación de usuarios	219
7.2 > Tipo de usuario: rol	220
8 >> Gestión de cursos	221
8.1 > Categorías	221
8.2 > Cursos	222
8.3 > Matriculación	225
8.4 > Grupos vs Cohortes	227
8.5 > <i>Copia de seguridad / Restaurar cursos</i>	228
8.6 > Funcionalidades	229

Apéndice: Siglas y Enlaces web de interés **236**





u n i d a d 1

Evolución de las aplicaciones web

SUMARIO

- Aplicaciones web
- Desarrollo de las aplicaciones web
- Esquema de funcionamiento de un servicio web
- Instalación de un servidor web

OBJETIVOS

- Conocer la evolución de las aplicaciones web.
- Distinguir los elementos necesarios para utilizar aplicaciones web.
- Instalar software que permita implantar aplicaciones web.
- Entender el esquema básico de funcionamiento de una aplicación web.

1 >> Aplicaciones web

Se denomina aplicación web al software que reside en un ordenador, denominado servidor web, que los usuarios pueden utilizar a través de Internet o de una intranet, con un navegador web, para obtener los servicios que ofrezca.

Existen multitud de aplicaciones web, de muy diversos tipos, tales como gestores de correo, web mails, wikis, blogs, tiendas en línea, etc.

Según el tipo de acceso, las aplicaciones web pueden ser:

- Públicas: como las tiendas virtuales, diarios digitales, portales de Internet, etc.
- Restringidas: como las **intranets**, que ofrecen servicios para mejorar las gestiones internas de una empresa, tales como el control de horas de su personal, gestión de proyectos y tareas, gestores documentales, etc. También suele estar restringido el acceso a aplicaciones web en las **extranets**, cuyo objetivo es aumentar y mejorar el servicio con distribuidores, clientes, proveedores, comerciales o colaboradores externos.

La popularidad de las aplicaciones web se basa en:

- La facilidad de acceso, ya que solo es necesario un navegador web.
- La independencia del sistema operativo.
- La facilidad de actualización y mantenimiento, sin tener que redistribuir y reinstalar el software a miles de usuarios potenciales.

La **World Wide Web** (WWW) es un conjunto de protocolos que permite la consulta remota de archivos de hipertexto. Utiliza Internet como medio de transmisión. Existen muchos otros servicios y protocolos en Internet: el envío de correo electrónico (SMTP), la transmisión de archivos (FTP y P2P), las conversaciones en línea (IRC), la mensajería instantánea y presencial, la transmisión de contenido y comunicación multimedia, los boletines electrónicos (NNTP), el acceso remoto a otros dispositivos (SSH y Telnet), etc.

Una página web estática es aquella que muestra información al navegante, de forma que este se limita a obtener dicha información, sin que pueda interactuar con la página web visitada.

Las webs estáticas están construidas principalmente con hipervínculos o enlaces (*links*) a otras páginas web; este tipo de webs estáticas son incapaces de soportar interactividad con el usuario, como distinguir unos usuarios de otros, recordar sus gustos, preferencias, etc. Algunos ejemplos pueden ser: foros, consultas *online*, redes sociales, etc.

Una página web dinámica es aquella que contiene elementos que permiten una comunicación activa entre el usuario y la aplicación, y cuyo contenido se genera a partir de lo que un usuario introduce en ella.

Estas páginas web dinámicas se pueden considerar una aplicación web, dado que permiten que el usuario acceda a los datos de modo interactivo. La página web responderá a cada una de sus acciones, como, por ejemplo, consultar el correo, rellenar y enviar formularios, participar en juegos o acceder a gestores de base de datos de todo tipo.

Vocabulario

Internet: conjunto descentralizado de redes de comunicación interconectadas que utilizan la familia de protocolos TCP/IP, garantizando que las redes físicas heterogéneas que la componen funcionen como una red lógica única, de alcance mundial.

Intranet: red de ordenadores particulares que utiliza tecnología Internet para compartir dentro de una organización parte de sus sistemas de información y sistemas operacionales. El término intranet se utiliza en oposición a Internet para resaltar que se trata de una red establecida en el ámbito de una organización.

Extranet: parte de la intranet de una organización que se extiende a usuarios fuera de ella, normalmente utilizando Internet. La extranet suele tener un acceso semiprivado; para acceder a la extranet de una empresa, el usuario no necesariamente ha de ser trabajador de la empresa, pero debe tener un vínculo con la entidad. Por ello, una extranet requiere cierto grado de seguridad, para que no pueda acceder cualquier persona.

Navegador web: aplicación que interpreta la información contenida en una página web y la visualiza.



Arquitectura cliente-servidor

Se trata de una aplicación distribuida en el que las tareas se reparten entre los proveedores de recursos o servicios, llamados servidores, y los demandantes, llamados clientes. Un cliente realiza peticiones a otro programa, el servidor, que le da respuesta. Esto también se puede aplicar a programas que se ejecutan sobre un solo ordenador, aunque es más eficaz en un sistema operativo multiusuario distribuido a través de una red de ordenadores.

CGI (*Common Gateway Interface*)

Estándar que permitía transferir datos entre los clientes y el programa servidor. Añadió interactividad a las páginas web, pero sus deficiencias en el desarrollo de aplicaciones y funcionamiento de las mismas condujo al desarrollo de interfaces de programación de aplicaciones (API), específicas de servidor, como ASP o PHP, más eficientes que su predecesor CGI.

2 >> Desarrollo de las aplicaciones web

Con la creación de Internet surgieron infinitas posibilidades de acceso a la información desde casi cualquier sitio. Esto representó un desafío a los desarrolladores de aplicaciones, ya que los avances en tecnología permitieron desarrollar aplicaciones más rápidas, ligeras y robustas para utilizar la red. Actualmente, hay nuevas tecnologías que permiten, por ejemplo, que el acceso a una base de datos desde una página web sea un mero trámite. El único problema es escoger la aplicación correcta para cada situación.

Al principio, con la arquitectura **cliente-servidor**, cada aplicación tenía su propio programa cliente, que servía como interfaz de usuario y tenía que ser instalado y actualizado. Un ejemplo de lo que esto suponía lo podemos ver con Facebook; cuando se cambia su funcionalidad a través de la Web, el usuario no tiene que realizar ningún cambio, pero cuando la quiere cambiar para su aplicación móvil, tiene que instalarse la actualización, lo que supone un inconveniente para los usuarios que tienen que instalarse a menudo actualizaciones de los programas, cosa que no ocurre con las aplicaciones web. El cliente realizaba peticiones a otro programa, el servidor, que le daba respuesta. El antiguo estándar CGI permitía transferir datos entre los clientes y el programa servidor.

En cambio, las aplicaciones web generan dinámicamente una serie de páginas web en un formato estándar, como HTML o XHTML, soportados por los navegadores web comunes. Se utilizan lenguajes interpretados en el lado del cliente, directamente o a través de *plugins*, tales como JavaScript, *applets* de Java, Flash, etc., para añadir elementos dinámicos en la interfaz del usuario. Generalmente, cada página web en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador web y, por lo tanto, el ordenador del cliente interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación web.

Gracias al potencial de estas tecnologías y a la búsqueda de un servidor más extensible y portable, la empresa Sun Microsystems (comprada por Oracle en 2010) desarrolló una tecnología llamada *servlet*, programa que se ejecuta en un servidor; los *servlets* Java fueron eficientes, debido al esquema de hilos de ejecución (*threads*) en el que se basan y al uso de una arquitectura estándar como la máquina virtual de Java (JVM, *Java Virtual Machine*). También se extendió la funcionalidad de los servidores web, a través de las denominadas *Java Server Pages*, JSP. Las JSP permiten juntar HTML, aplicaciones Java, y componentes, como las *Java Beans*, creando una página web especial que el servidor web compila dinámicamente en un *servlet* la primera vez que es llamada.

Otra tecnología de éxito y una de las más utilizadas es el lenguaje **PHP**. Se trata de un lenguaje interpretado en el lado del servidor que permite la incrustación de HTML en los programas, con una sintaxis derivada de los lenguajes C y Perl. El hecho de ser sencillo y potente ha contribuido a hacer de PHP una herramienta muy apropiada para determinados desarrollos web.

-

Copyright © 2013. Macmillan Iberia, S.A.. All rights reserved.

Vocabulario

Ontología: en informática, se refiere al intento de formular un exhaustivo y riguroso esquema conceptual dentro de un dominio dado, para facilitar la comunicación y la distribución de la información entre diferentes sistemas. Un uso común tecnológico se encuentra en la inteligencia artificial y en la representación del conocimiento.

Inteligencia Artificial (IA): agrega a este set de tecnologías la parte de procesamiento computacional para deducir nuevo conocimiento a partir del que se brinda por las distintas fuentes de conocimiento. Básicamente, la IA toma como base las reglas y relaciones entre los conceptos y objetos, y deduce nuevas verdades haciendo crecer la base de conocimiento.

2.2 > Hacia la Web 3.0

Actualmente, se está desarrollando la **Web 3.0 o Web “semántica”**, donde las actividades de los usuarios se analizan en detalle. Así, las conductas de comportamiento, el historial de compras, frecuencia y tamaño de pedidos, gustos, preferencias, etc., son datos que quedan registrados al navegar a través de la web.

La Web 3.0 será más **“inteligente”**, los usuarios podrán hacer búsquedas más cercanas al lenguaje natural. Por ejemplo, si una persona tiene que buscar vuelos para mañana por la mañana, el servidor tendrá que entender si “mañana” se refiere al día siguiente o al periodo previo a las 12:00. Es decir, el servidor tiene que entender nuestro lenguaje.

Además, la información tendrá contenido semántico asociado y la Web podrá relacionar conceptos de múltiples fuentes y deducir información a través de reglas asociadas al significado del contenido, consiguiendo así búsquedas mucho más eficientes. Por ejemplo, SIRI o Google Search, son nuevos sistemas de búsqueda pensados para entender al usuario. Así, se podrán preguntar cosas como “¿necesitaré paraguas mañana?”, y el sistema se conectará a servicios web de predicción meteorológica para saber el tiempo que hará mañana en nuestra ubicación y así contestar si necesitaremos paraguas o no.

Las tecnologías y conceptos que permiten desarrollar la Web 3.0 son:

- **RDF** (*Resource Description Framework*): convierte las descripciones de los recursos en expresiones con la forma sujeto (aquello que se está describiendo) – predicado (relación establecida acerca del recurso) – objeto (el otro recurso con el que se establece la relación).
- **RDF Schema:** lenguaje de ontologías que proporciona los elementos básicos para la descripción de vocabulario.
- **OWL** (*Web Ontology Language*): un lenguaje de marcado para publicar y compartir datos usando ontologías en la WWW, que tiene como objetivo facilitar un modelo de marcado construido sobre RDF y codificado en XML.
- **XML** (*eXtensible Markup Language*): lenguaje estándar para el intercambio de información entre diferentes plataformas.

La combinación de RDF con otras herramientas como RDF Schema y OWL, permite añadir significado a las páginas, y es una de las tecnologías esenciales de la Web semántica.

Actividades propuestas

- 1•• Elabora una tabla resumen que defina cada una de las diferentes tecnologías utilizadas en la creación de las aplicaciones web, desde las más antiguas hasta las más modernas.
- 2•• Explica las semejanzas y diferencias entre la Web 1.0 y la Web 2.0.
- 3•• ¿Cuál es tu opinión sobre las capacidades de la Web 3.0 respecto a la privacidad de los usuarios?
- 4•• Enumera los diferentes tipos de servicios que ofrecen las aplicaciones web.

3 >> Servicios web

Los servicios web engloban una serie de tecnologías, protocolos y estándares que permiten el diálogo entre sistemas informáticos. Independientemente de su plataforma, utilizan los propios recursos de comunicación que ofrece Internet; por ejemplo, los protocolos http y https que usan los navegadores web como clientes, para intercambiar datos con los servidores web.

3.1 > Esquema de funcionamiento de un servicio web

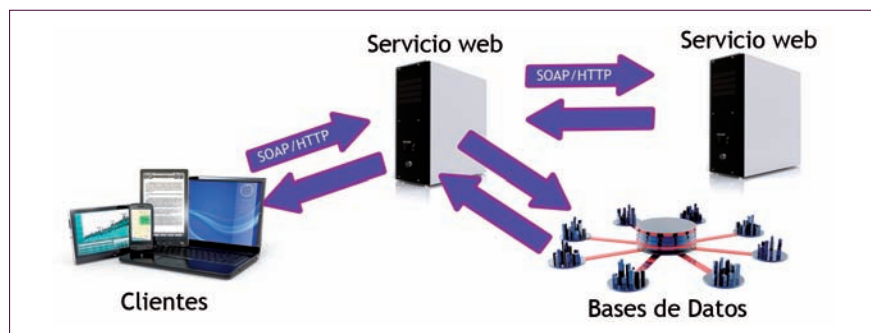
Los servicios web se basan en la arquitectura cliente-servidor, con una ventaja principal, y es que garantizan la independencia del lenguaje y del sistema que los alberga. Por ejemplo, puedo conectar servicios web hechos en Java sobre una máquina con Sistema Operativo MacOs con otro servicio web hecho en C sobre Windows. Lo importante es la funcionalidad que ofrece y no el sistema o el lenguaje sobre el que está implementado.

Para describir el esquema de funcionamiento básico de un servicio web, se tomará como ejemplo el funcionamiento de un servidor de páginas web. Este es un programa que sirve para atender y responder a las diferentes peticiones de los navegadores (clientes), proporcionando los recursos que soliciten, en este caso los contenidos de las páginas web, usando el protocolo http o el protocolo https (versión segura).

Un servidor web básico tiene un esquema de funcionamiento muy simple, basado en ejecutar infinitamente el siguiente bucle:

- Espera peticiones en el puerto TCP indicado (el estándar por defecto para HTTP es el 80).
- Recibe una petición.
- Busca el recurso.
- Envía el recurso utilizando la misma conexión por la que recibió petición o devuelve el error 404 si no lo encuentra.
- Vuelve al primer punto.

Estos servicios web (*Web Services*) constituyen uno de los pilares de la Web 2.0 y permiten crear arquitecturas orientadas a servicios para conseguir que, no solo los datos, sino también muchos tipos de operaciones y servicios puedan procesarse de forma deslocalizada, promoviendo así una mayor participación y transparencia en la creación de contenidos por parte de usuarios y desarrolladores.



1.2. Esquema de funcionamiento de un servicio web.

Algunas de las tecnologías implicadas en los servicios web son:

WSDL (*Web Services Description Language*)

Lenguaje estándar que describe la interfaz pública de los servicios web y permite a un cliente saber qué funciones ofrece el servidor.

SOAP (*Simple Object Access Protocol*)

Protocolo estándar que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML.

XML Schema

Lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML.

3.2 > Instalación de un servidor web, un servidor de bases de datos y un intérprete de PHP

Para poder gestionar la diversidad de aplicaciones y servicios web que permiten publicar diferentes tipos de contenidos (blogs, wikis, foros, etc.) en las páginas web, los servidores web necesitan comunicarse con algún servidor de base de datos, encargado de almacenar toda la información necesaria para el mantenimiento de esos servicios, y con un intérprete de alguno de los lenguajes de programación de servidores (PHP, Perl, Python, etc.).

Para ello, instalaremos XAMPP (la X porque es multisistema, A de Apache, M de MySQL, P de PHP y la otra P de Perl). Hay productos como LAMP para Linux, MAMP para MacOS o WAMP para Windows.

XAMPP es una aplicación que consiste en el servidor web **Apache**, la base de datos **MySQL** y un intérprete del lenguaje de programación **PHP**. Posteriormente se pueden instalar, configurar y mantener otras aplicaciones web sobre estos servidores fundamentales.



Casos prácticos

1

Instalación de XAMPP en Windows y Linux

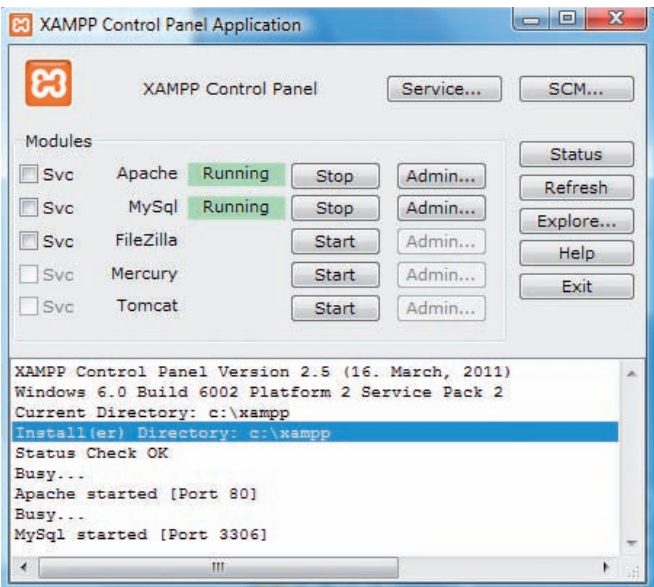
•• Realiza la instalación de XAMPP, tanto en un sistema operativo Linux, como en un sistema Windows.

Solución •• Para realizar la instalación de XAMPP en ambos sistemas operativos, primero descarga del CD *Recursos* las versiones utilizadas en este caso práctico (1.7.7), o bien puedes acceder a la página oficial y descargar las versiones más actuales de la página del propio programa XAMPP.

- XAMPP Windows 1.7.7
- XAMPP Linux 1.7.7

Para instalar en un sistema **Windows**, seguiremos los siguientes pasos:

1. Ejecuta el fichero instalador recién descargado; automáticamente comenzará la instalación. Selecciona el idioma inglés, salvo que prefieras el alemán; luego podrás administrar en castellano.
2. Haz clic en *Next*, en el mensaje de bienvenida de XAMPP. Confirma *c:\xampp* como carpeta de destino de la instalación. Mantén las opciones por defecto y pulsa *Install*, para continuar, y después *Finish*, para terminar con la instalación.
3. Pulsa *Si* para comenzar a usar el panel de control de XAMPP. Pulsa el botón *Start* en Apache y MySQL para arrancar el servidor web y el servidor de bases de datos.



4. Abre un navegador e indica la dirección `http://localhost`, para conectarte al servidor Web y comprobar la instalación.
5. Finalmente, elije el idioma y verás la ventana de bienvenida de administración del servidor XAMPP.



Para instalar en un sistema Linux, seguiremos los siguientes pasos:

1. Descomprime el fichero descargado, XAMPP-Linux-1.7.7.tar.gz; para ello, en una shell de Linux, inicia sesión como administrador del sistema (`su`) para tener los privilegios necesarios de creación de carpetas. Extrae el archivo descargado a `/opt` con este comando: `tar xvfz xampp-linux-1.7.7.tar.gz -C /opt`
2. Comprueba que XAMPP se instala bajo el directorio `/opt/lampp`. Desde esta carpeta escribe la siguiente orden para arrancar con XAMPP el servidor web y el servidor de bases de datos: `./lampp start`. Y para detener los servicios escribe: `./lampp stop`

```
root@administrador:/opt/lampp# ./lampp start
Starting XAMPP for Linux 1.7.7...
XAMPP: Starting Apache with SSL (and PHP5)...
XAMPP: Starting MySQL...
XAMPP: Starting ProFTPD...
XAMPP for Linux started.
```

3. Abre un navegador e indica la dirección `http://localhost`, para conectarte al servidor web y comprobar la instalación. Elige el idioma y verás la ventana de bienvenida de administración del servidor XAMPP.

Actividades propuestas

5. Busca información en Internet sobre los protocolos http y https; explica las diferencias entre ellos.
6. Realiza la instalación de XAMPP en una máquina virtual de Windows y en otra de Linux.

Actividades finales

.: CONSOLIDACIÓN .:

- 1.. Según el tipo de acceso, ¿cómo pueden ser las aplicaciones web?
- 2.. ¿Cuáles son las principales características que han hecho populares las aplicaciones web?
- 3.. ¿Qué diferencias hay entre una página web dinámica y otra estática?
- 4.. ¿En qué se fundamenta la arquitectura cliente-servidor?
- 5.. ¿Podrías explicar las ideas principales en las que se basa la Web 3.0?
- 6.. ¿Cuáles son las tecnologías y conceptos que permiten desarrollar la Web 3.0?
- 7.. ¿Para qué sirven los servicios web de manera general?
- 8.. ¿Qué necesitan los servidores web para poder almacenar y publicar blogs, wikis o foros?



.: APLICACIÓN .:

- 1.. Busca en Internet información sobre las aplicaciones Web 2.0 y desarrolla cada una de las características mencionadas en la unidad.
- 2.. Averigua la diferencia entre lenguajes interpretados en el lado del cliente y lenguajes interpretados en el lado del servidor.
- 3.. Investiga en Internet y haz una lista con los lenguajes interpretados en el lado del cliente, y otra con los lenguajes interpretados en el lado del servidor.
- 4.. Investiga y desarrolla los conceptos en los que se basa la Web 3.0, que la diferencian de la Web 2.0.
- 5.. Busca por Internet aplicaciones similares a XAMPP, que permitan integrar la instalación, la configuración y el mantenimiento de aplicaciones web de manera centralizada.
- 6.. Busca información en foros de Internet e indica las diferencias que hay entre XAMPP y cualquier otra aplicación de instalación integrada similar a XAMPP, explicando las ventajas e inconvenientes de cada aplicación.

Caso final

2

Interfaz de administración de XAMPP, creación de una base de datos

•• Una vez que hemos instalado la aplicación XAMPP, ahora vamos a conocer las opciones de administración que XAMPP nos ofrece a través de su interfaz web y crearemos una base de datos.

Solución •• Teniendo arrancados los servidores Apache y MySQL, haz clic en los enlaces de la izquierda de la web de XAMPP para explorar las siguientes opciones de administración que ofrece.

Estado: permite ver qué componentes de XAMPP ya han sido iniciados y funcionan correctamente.

Chequeo de seguridad: informa sobre los posibles agujeros de seguridad en la instalación del sistema.

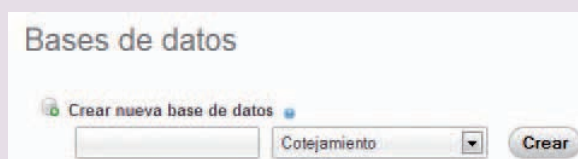
Documentación: ofrece documentación de los paquetes más importantes de XAMPP.

phpinfo(), perlinfo() y Estado (dentro de JEE): información sobre los intérpretes de los lenguajes de programación PHP, Perl y el contenedor de *servlets* de Java(J2EE), Tomcat.

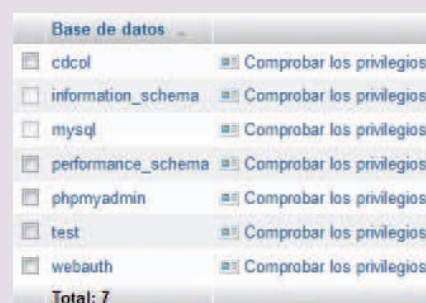
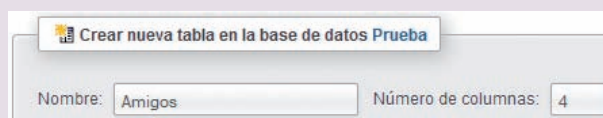
Ejemplos de programación PHP y Perl (Administración de CD, Biorritmo, Instant Art y Agenda de teléfonos, Libro de invitados): se pone a disposición el código fuente de varios ejemplos de programas en PHP y Perl; ejemplos de conexiones a una base de datos MySQL, y generación de un fichero, pdf; de php con la librería de gráficos GD (*Graphics Draw*); de manejo del motor de tipos de letra FreeType; de php con el sistema gestor de bases de datos SQLite; un libro de visitas en Perl, etc.

phpMyAdmin: es la herramienta de administración de las bases de datos MySQL.

1. Vamos a crear una nueva base de datos, pincha sobre la pestaña *Bases de Datos*, escribe el nombre que le asignes (*Prueba*) y pulsa sobre el botón *Crear*.



2. Observa que se ha creado la base de datos denominada *Prueba*.
3. Selecciona *Prueba*, añade la tabla *Amigos* a la base de datos y pulsa *Continuar*.



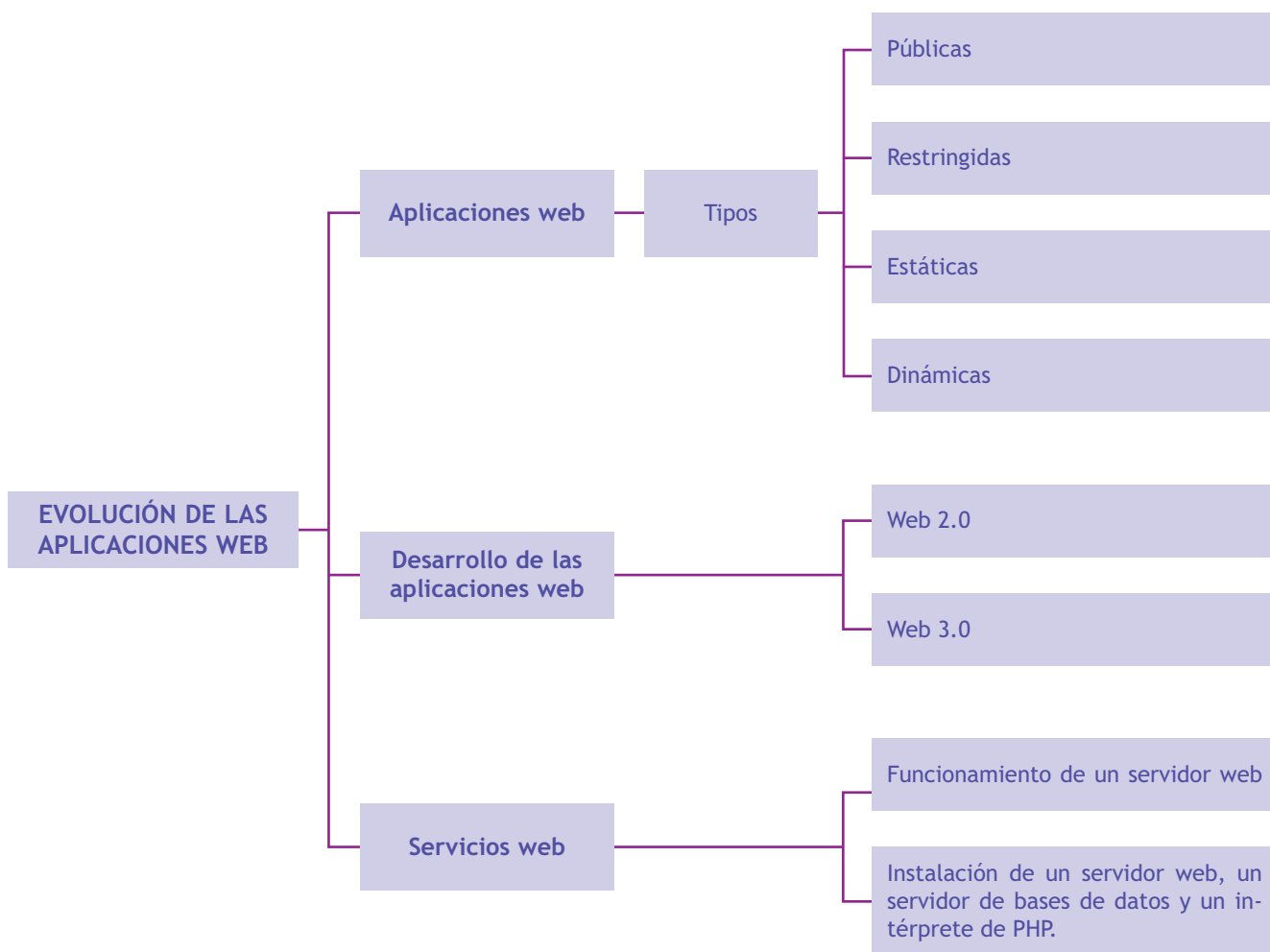
4. Introduce los campos *Nombre*, *Apellidos*, *Cumpleaños* y *Móvil*.
5. Baja el cursor y pulsa en la opción *Guardar* para crear la tabla *Amigos*.



6. Comprueba que la tabla *Amigos* se ha creado satisfactoriamente; ahora puedes insertar los datos de tus amigos; también podrás modificar la estructura de la tabla.



Ideas clave



¿Hacia la Web 4.0?

Este término motiva a pensar, ¿qué será ese tipo de Web? Actualmente, algunos la definen como el resultado de sumar Web 3.0 (Web semántica) + Web 3D + Inteligencia Artificial + Voz como vehículo de intercomunicación = Web 4.0 (WebOS); es decir, que una vez se establezca la Web semántica (entre el año 2010 y el 2020) será el turno de avanzar hacia la Web 4.0, en la que el sistema operativo establecido en la web cobre protagonismo, hacia una Web ubicua, donde el objetivo primordial será el de unir las inteligencias, donde tanto las personas como las cosas se comuniquen entre sí para generar la toma de decisiones.



Para el 2020 se espera que haya agentes en la Web que conozcan, aprendan y razonen como lo hacemos las personas.

Fuente: <http://www.dipity.com>

Actividades

- 1.. ¿Cómo imaginas la Web 4.0 en el futuro?
- 2.. Debate con tus compañeros sobre los nuevos servicios que podría proporcionar la Web del futuro.

Lenguajes de marcas, hojas de estilos y *scripts*

SUMARIO

- Lenguaje de marcas HTML5
- Estándares y versiones
- Normas y elementos de HTML5
- Hoja de estilos CSS3: reglas de estilo
- Identificadores, clases, pseudo-clases y pseudo-elementos
- JavaScript: características, sentencias de control y funciones

OBJETIVOS

- Conocer las características del lenguaje de marcas HTML.
- Distinguir los elementos de la estructura de una página web escrita en HTML.
- Crear páginas web utilizando el lenguaje HTML.
- Conocer la utilidad de CSS y sus diferentes formas de uso.
- Conocer la estructura de las reglas de estilo y crear con ellas hojas de estilos.
- Crear páginas web con estilos CSS.
- Crear *scripts* básicos con JavaScript.

1 >> Lenguajes de marcas, HTML5

El lenguaje de marcas HTML te permitirá realizar tus páginas web; solo necesitas un editor de textos ASCII para incluir el contenido que quieres mostrar en tus páginas (Bloc de notas, Word, etc.), y un navegador web para visualizar la página. Los ficheros que contienen documentos HTML deben tener la extensión .html o .htm.

Existen otros editores HTML más específicos de tipo WYSIWYG (*What You See Is What You Get*), acrónimo en inglés que se traduce por “lo que ves es lo que obtienes”, con los que es posible escribir documentos HTML viendo de manera simultánea cómo sería el resultado final de la página web, tal y como se visualizaría cuando estuviera publicada en Internet.

Así, con herramientas de edición como *Dreamweaver*, *KompoZer*, *Aptana*, *Amaya*, etc., se pueden realizar páginas web de una manera más profesional. Dichos editores permiten, por medio de menús e iconos, incluir directivas (etiquetas) del lenguaje de marcas HTML sin necesidad de teclearlas, reglas de estilo CSS, así como otras muchas funciones orientadas a la creación y el mantenimiento de páginas web.

Una vez tengas tus páginas listas para ser publicadas en Internet, necesitarás un servidor de páginas web donde alojarlas. Un servidor web es un software que reside en un ordenador que está permanentemente conectado a Internet. Al colocar páginas web en el servidor, estas se hacen accesibles para todos los usuarios de su misma red.

Hay proveedores de servicios de Internet que ofrecen a sus clientes espacios web gratuitos para publicar páginas personales o corporativas, lo que evita la necesidad de instalar un servidor web propio.

1.1 > Definiciones básicas

En primer lugar conviene aclarar el significado de algunas siglas que representan los lenguajes de marcas básicos.

¿Qué es el HTML?

El HTML (*Hyper Text Markup Language*), lenguaje de marcado de hipertexto, es un lenguaje de marcas (utiliza etiquetas, como marcas para delimitar elementos del lenguaje), que sirve para describir el contenido y la estructura de las páginas web, que pueden ser interpretadas y visualizadas a través de los navegadores de Internet (clientes web: Firefox, Internet Explorer, Chrome, etc.).

HTML permite publicar documentos en línea que contengan encabezados, textos, tablas, listas, fotos, etc., así como recuperar nueva información a través de otros enlaces de hipertexto, con el clic de un botón del ratón. También sirve para diseñar interfaces con formularios, en los cuales los visitantes podrán rellenar los datos, por ejemplo para la búsqueda de una información concreta, hacer reservas de algún evento, viaje o similar, pedir productos, etc. Pero HTML no puede procesar los datos introducidos en los formularios; para ello se necesita un lenguaje de programación web del lado del servidor, como PHP, JSP, etc.

Navegadores y HTML5

Los actuales navegadores ya están diseñados para manejar la mayoría de las funcionalidades recogidas en HTML5, pero no todas. Para saber si un navegador soporta todas las características de HTML5 puede utilizarse la aplicación de la página <http://www.html5test.com>.

También se puede comprobar si el código HTML de una página web valida con las reglas de HTML5 en:

<http://validator.w3.org/>



W3C

El consorcio W3C (*World Wide Web Consortium*) es un organismo internacional que trabaja para desarrollar reglas y convenciones (estándares) para la red informática mundial. Está liderado por el creador de la Web, Tim Berners-Lee y su misión es guiar la Web hacia su máximo potencial.

XML (*eXtensible Markup Language*)

Como su nombre indica es un lenguaje de marcas extensible, un metalenguaje definido también por el W3C y derivado del SGML, con el fin de ser más adecuado para uso web. Está basado en un sencillo formato de texto para representar la información estructurada de documentos, datos, configuración, libros, transacciones, facturas, etc.

Los nombres de las etiquetas de elementos y de los atributos tienen que estar en minúscula, los valores de los atributos tienen que estar entre comillas dobles (") o simples ('), todos los elementos tienen que estar cerrados, ya tengan contenido (<p> ... </p>) o no (
, los elementos deben de estar correctamente anidados, el elemento raíz (<html>) del documento debe incluir el atributo xmlns, y también se debe incluir una definición (DTD) de tipo de documento.

¿Qué es SGML?

El SGML (*Standard Generalized Markup Language*) es una tecnología estándar (ISO 8879), que sirve para definir lenguajes de marcado generalizado que permitan organizar con etiquetas los documentos. A partir de SGML se creó HTML como una versión simplificada.

El marcado generalizado se basa en dos postulados:

- El marcado debe ser declarativo, es decir, debe describir la estructura de un documento y otros atributos.
- El marcado debe ser riguroso, para que otros programas, bases de datos, etc., puedan procesar estos documentos.

Para la definición de los documentos se utiliza una DTD (*Document Type Definition*), que permite al navegador saber la definición del tipo de documento que debe interpretar.

Una DTD es un documento SGML que contiene las reglas sintácticas que definen un tipo de documento específico. Contiene los elementos permitidos y sus atributos, así como otras reglas sobre la anidación de los elementos y los valores de los atributos. Si se contrasta un documento con su correspondiente DTD, se puede comprobar si este es válido o no, es decir, si la cumple.

Según el W3C, el propósito de una DTD es definir los componentes básicos válidos de un documento; para ello, en una DTD se define la estructura de los documentos con una lista de elementos y atributos que se pueden utilizar en su generación.



¿Qué es el XHTML?

XHTML (*eXtensible HyperText Markup Language*) es un lenguaje considerado una variante, más restrictiva y ordenada, del HTML, que utiliza la sintaxis propia del lenguaje XML (*eXtensible Markup Language*). Se consideró inicialmente como sustituto del HTML, pero en la práctica no fue así, dado que diferencias con los fabricantes de los navegadores hizo que HTML siguiera usándose manteniendo una continua evolución.

XHTML tiene prácticamente los mismos elementos que HTML, pero su sintaxis se orienta de manera diferente para exigir una estructuración de los elementos más coherente que en HTML. Debido a que XHTML es una aplicación del XML, es decir, está escrito de forma que cumpla las normas sintácticas del XML, puede utilizar otras herramientas de XML como XSLT, un lenguaje para transformar contenido XML. La separación del contenido de un documento de su formato, es decir, de cómo es presentada esa información, fue una de las mejoras que aportó el XHTML que luego se han incorporado en versiones posteriores, incluso de otros lenguajes.

1.2 > Estándares y versiones

Dado el rápido crecimiento de la Web y la evolución gradual del HTML, surgió la necesidad de estandarizarlo para que tanto los autores como los navegadores pudieran reconocer la versión de HTML a utilizar. HTML se convirtió en estándar en 1995, y desde entonces ha seguido en constante desarrollo. En los últimos años, la versión de HTML recomendada por el W3C era HTML 4.01.

Al diseñar una página web conviene especificar qué versión de HTML se está usando; esto se hace indicando la etiqueta `<!DOCTYPE>` en la primera línea, información útil para que el navegador sepa interpretarla correctamente. El **HTML 4.01** tenía **tres variantes de DTD**:

- HTML 4.01 Strict: (*Strict DTD*), la más restrictiva porque no se pueden usar las etiquetas obsoletas, solo las definidas en HTML 4.01. Para utilizar esta versión, la primera línea del documento será la siguiente:

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

- HTML 4.01 Transitional: (*Transitional DTD*) es una mezcla de todos los HTML, en la que se aceptan las etiquetas obsoletas, la mayoría de los cuales están relacionados con la presentación visual. Se llama Transitional porque está pensado para los que no se atreven a usar el Strict, pero les gustaría hacerlo en el futuro. Para utilizar esta versión, pondremos la siguiente línea al principio del documento:

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

- HTML 4.01 Frameset: (*Frameset DTD*) admite lo mismo que el Transitional, y además soporta marcos (*frames*), que actualmente se consideran algo obsoleto y anticuado. Para utilizar esta versión, pondremos lo siguiente en la primera línea del documento:

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

La **última versión** de HTML es **HTML5**, para la cual el W3C está preparando el futuro estándar de referencia. Esta versión está ya implementada en las últimas versiones de los navegadores más populares, y es muy utilizada por los desarrolladores web por los avances, mejoras y ventajas que ofrece (herramientas gráficas de geolocalización, de arrastrar y soltar, nuevos formularios, gráficos, animación, etc.). Además, es muy compatible con XHTML y etiquetas de HTML 4.

Para HTML5 no se define una DTD. La declaración de la etiqueta `<!DOCTYPE>`, que podía ser omitida en las versiones anteriores, se simplifica mucho. Solo indica a los navegadores que el documento utiliza la versión HTML5, pero no hay que indicar ningún sistema de identificación público; simplemente debe aparecer la declaración:

```
<!DOCTYPE html>
```

Referencia de la DTD

Si se trabaja con **XHTML Strict**, la referencia de la DTD es:

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

Si se usa **XHTML Transitional**, la referencia de la DTD es:

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

Para **XHTML Frameset**, la referencia de la DTD correcta es:

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



Agrupación de elementos

HTML5 dispone de varios elementos que no se representan ni se visualizan en pantalla, pero que permiten agrupar a otros elementos dándoles un significado semántico común. Este significado común permite a los navegadores y los buscadores presentar y usar los documentos o aplicaciones en una gran variedad de contextos. Esto introduce importantes cambios respecto a lo que conocíamos hasta ahora en HTML 4.01, donde se solía estructurar los documentos en divisiones con etiquetas `<div>`, y para distinguir el contenido de cada una de ellas se les asignaba un atributo `id`, con valores como header, nav, footer, etc., para dar una idea de los datos contenidos en dichas divisiones.

Algunas etiquetas nuevas de HTML5 son precisamente: **article**, **aside**, **header**, **footer**, **nav**, **figure**, **dialog**, **section**, **hgroup**, **video**, **audio**, **embed**, **mark**, **progress**, **meter**, **time**, **canvas**, **command**, **details**, **output**, **input**.

1.3 > Normas fundamentales

Antes de comenzar a utilizar cualquier lenguaje de marcas es necesario conocer sus normas básicas; veamos las de los elementos fundamentales en HTML5.

Etiquetas

Las marcas o etiquetas definen los elementos que conforman el léxico del lenguaje HTML. Se delimitan por los signos “menor que” (<) y “mayor que” (>). Las etiquetas de HTML pueden ser de dos tipos, cerradas o abiertas.

Las **etiquetas cerradas** son aquellas formadas por una palabra reservada del lenguaje para indicar el principio de la etiqueta (etiqueta de apertura) y otra para el final (etiqueta de cierre); en esta última se incluye el carácter “/” antes de la misma palabra reservada de dicha etiqueta. Ejemplo de una etiqueta cerrada es: `<p>Párrafo bien cerrado</p>`

Se denominan **etiquetas abiertas** a las que constan de una sola palabra reservada que indica a la vez el principio y fin del elemento, es decir, no tienen la correspondiente etiqueta de cierre. Ejemplos de etiquetas abiertas son: `<hr>`, `
`, ``, `<input>`

En algunas versiones de HTML, como XHTML 1.0, lo correcto sería poner una barra (/) justo antes del cierre de cada etiqueta abierta, por ejemplo `
`; así se indicaría que se abre y se cierra a la vez el mismo elemento.

Atributos

Las etiquetas pueden contener atributos, los cuales sirven para configurar diferentes características de las propias etiquetas donde se indiquen. Los atributos se deben especificar a continuación de la palabra reservada de la etiqueta de apertura (nunca de la de cierre) separados por un espacio, y antes del signo de cierre de etiqueta “mayor que” (>) que la delimita. Detrás del nombre del atributo, se indicará el valor del atributo que se quiera asignar, separando el atributo y su valor por el signo “igual” (=).

En HTML5 se recomienda indicar el valor entre comillas dobles para aumentar la legibilidad del código y su compatibilidad con otras versiones de HTML, aunque no sea obligatorio, salvo que el valor contenga caracteres de espacio. Ejemplo: `<input type="text" name="nombre">`

Tanto las etiquetas como los atributos se pueden indicar en minúsculas o en mayúsculas, en cambio los valores de los atributos sí se diferencian según se indiquen en mayúsculas o en minúsculas. Se recomienda usar minúsculas tanto para las etiquetas como para los atributos.

Comentarios

Los comentarios son líneas de texto que no son interpretadas por el navegador y, por lo tanto, no serán visualizadas al navegar sobre la página.

Permiten realizar anotaciones y aclaraciones sobre el código fuente de una página web, muy útiles en futuras revisiones y modificaciones. Los comentarios se incluyen entre los símbolos `<!--` y `-->`. Ejemplo: `<!-- Esto es un comentario sobre mi pagina Web -->`

Se recomienda seguir estas normas básicas al crear páginas en HTML5, dado que al ser también reglas utilizadas en la sintaxis de XHTML, permiten reutilizar código de otras páginas web de versiones anteriores, ya sean de HTML o de XHTML.

1.4 > Estructuración básica

Un documento escrito en HTML debe tener una estructuración que contenga los siguientes elementos básicos o **etiquetas fundamentales**:

<code><html></code>	Indica el inicio del documento.
<code><head></code>	Inicio de la cabecera del documento.
<code><title></code>	Inicio del título del documento.
<code></title></code>	Final del título del documento.
<code></head></code>	Final de la cabecera del documento.
<code><body></code>	Inicio del cuerpo del documento.
<code></body></code>	Final del cuerpo del documento.
<code></html></code>	Final del documento.

HTML5 introduce nuevas características y elementos, a través de nuevas etiquetas, para ayudar a los autores de aplicaciones web. Estos elementos se basan en la investigación y análisis de muchas páginas web, detectando prácticas predominantes utilizadas por los autores, prestando especial atención a la definición de criterios claros e intentando mejorar la interoperabilidad entre navegadores y motores de búsqueda.

HTML5 propone una estructuración avanzada que permite definir los contenidos agrupando elementos con nuevas etiquetas, utilizando nombres que tienen un significado relacionado con la estructura de la página. Algunas de estas **nuevas etiquetas** son:

<code><header></code>	: encabezado de página.
<code><nav></code>	: enlaces de navegación.
<code><article></code>	: artículo publicado.
<code><section></code>	: sección dentro de un artículo o documento.
<code><aside></code>	: barra lateral.
<code><footer></code>	: pie de página.
<code><dialog></code>	: diálogos y comentarios.

En este sentido, HTML5 mantiene elementos como `<div>` y ``, que representan una agrupación de sus elementos hijos, y utiliza atributos como *class*, *id*, *lang* o *title*, que permiten establecer una semántica y estilo común, a través de CSS, al grupo de elementos consecutivos que contienen.

Con HTML5 cambia la funcionalidad de algunos elementos (etiquetas) de versiones anteriores (menú, cite, a, address, s, small) mientras, otros desaparecen por ser aplicados de mejor manera con CSS (basefont, big, center, font, strike, tt, u), por no favorecer la usabilidad y la accesibilidad (frame, frameset, noframes), o por haber sido reemplazados por otros o bien provocar ambigüedad o confusión (acronym, applet, isindex, dir).

Las nuevas etiquetas de HTML5 están enfocadas a lograr una web semántica.

Atributos comunes (HTML5)

Algunos **atributos comunes** a todas las etiquetas HTML5 son:

class: permite definir la clase.

id: crea un identificador del elemento.

contextmenu: permite asociar un menú de contexto que deberá estar definido en una etiqueta MENU.

contenteditable: valor booleano que indica si el contenido delimitado por la etiqueta se puede modificar.

dir: dirección del elemento d.

lang: especifica el idioma

draggable: indica si el elemento puede ser arrastrado y soltado.

dropzone: especifica si los datos arrastrados serán copiados, movidos o vinculados, cuando se suelta un elemento tras arrastrarlo.

hidden: elemento oculto que podrá visualizarse con código JavaScript.

accesskey: acceso directo de teclado que activa un elemento.

translate: modo de traducción.

spellcheck: para comprobar la gramática y ortografía.

style: para definir el estilo.

tabindex: orden de tabulación.

title: información de ayuda.

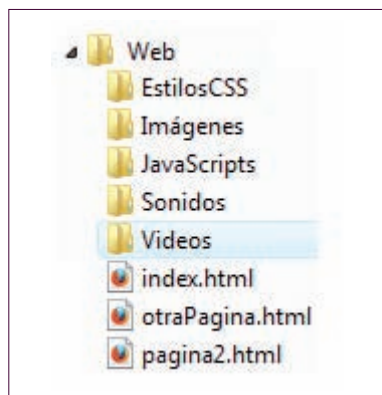
eventos: en HTML son sucesos que ocurren en la interfaz de usuario y que se asocian a un programa, por ejemplo, de JavaScript. Pueden ser asociados a los elementos de HTML para ejecutar el *script* relacionado cuando se produzca el evento, por ejemplo, **onclick**, **ondblclick**, **onmouseup**, **onmouseover**, **onfocus**, **onplay**, **onreset**, **ondrop**, etc.

Anidación de elementos

Las etiquetas de HTML pueden anidarse, de manera que un elemento HTML puede contener a otros. Las etiquetas anidadas deben estar bien "balanceadas", es decir, si se abren en un orden determinado, se deben cerrar en el orden inverso al de apertura, debiéndose cerrar primero las etiquetas de nivel inferior antes que la del nivel superior que la contiene.

```
<body>
<h1>Ejemplo
incorrecto</body>
</h1>
<body>
<h1>Ejemplo
correcto</h1>
</body>
```

Es conveniente seguir estas normas al crear código HTML5, ya que así también se cumplen las reglas de la sintaxis de XHTML, lo que permitirá reutilizar ciertas partes del código HTML para utilizarlo en el diseño de páginas web que utilicen versiones anteriores del lenguaje, ya sean de HTML o de XHTML.



Ejemplo de estructura de directorios de un sitio web.

Cabecera del documento

La directiva opcional **<head>** **</head>** genera el primer elemento dentro del elemento superior global **<html>**; delimita la cabecera del documento. Dentro de la cabecera se deben definir los elementos generales, como el título de la página, por medio de la etiqueta **<title>** (título que aparecerá en la barra superior del navegador web), la codificación de caracteres a utilizar (UTF8, etc.), la hoja de estilos asociada, u otros que se quiera que afecten a todo el documento HTML. Ejemplo: **<title>Mi primera página web</title>**

Dentro de la cabecera podemos incluir otras directivas adicionales. La directiva **<meta>** indica al visor de Internet las palabras clave y contenido de nuestra página web, entre otras posibles cosas. Muchos de los buscadores de páginas web de Internet (Google, Bing, Yahoo, Lycos, etc.) utilizan el contenido de esta etiqueta para incluir la página en sus bases de datos. La directiva **<meta>** lleva generalmente dos atributos o parámetros, *name* y *content*, que pueden indicar el nombre de la página y sus contenidos principales, o bien las palabras clave, para los buscadores de Internet. Ejemplos:

```
<meta name = "description" content = "Página personal
de Javi con sus gustos y aficiones">
<meta name = "keywords" content = "deportes cine
música actualidad">
```

Otro uso de la etiqueta **<meta>** es la de indicar documentos con "refresco automático". Si se indica una URL se sustituirá el documento por el indicado una vez transcurridos el número de segundos especificados. Si no se incluye ninguna URL, se volverá a cargar en el visor el documento en uso transcurridos los segundos indicados. Esto es útil para páginas que cambian de contenido con mucha frecuencia o para redireccionar, a la persona que visita nuestra página web, hacia una nueva dirección donde quizás se encuentra una versión actualizada de la página web, o bien hacia otra página diferente que pueda ser de su interés. Ejemplo:

```
<meta http-equiv = "refresh" content = "15"; url =
http://www.google.es">
```

Transcurridos 15 segundos se accederá a la página web de Google.

El elemento **<base>** permite indicar la URL base de los documentos, gráficos, sonidos, etc., a los que se hace referencia en la página web. Si no se incluye esta directiva, el navegador entiende que dichos elementos se encuentran en el mismo directorio donde se encuentran los ficheros con extensión .htm o .html que componen la página web. Si se incluyen varias directivas **<base>**, solo se tiene en cuenta la primera. Ejemplo:

```
<base href = "http://www.mipagina.es/recursos/">
```

Antes de crear una página web es conveniente planificar su diseño y ordenar la información y los recursos que se pretende ofrecer; para ello resulta de gran utilidad generar una **estructura de directorios** que almacene de manera coherente y ordenada todos los ficheros que contengan las imágenes, sonidos, vídeos, o cualquier otro tipo de contenido que se ponga a disposición a través de la página web.

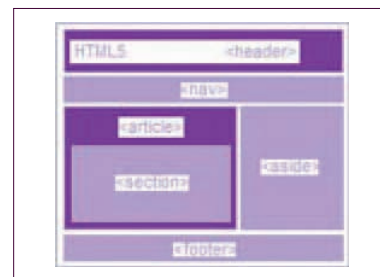
Cuerpo del documento

La directiva **<body></body>** siempre va detrás de **<head>**. Es el segundo elemento dentro del elemento **<html>**, y contiene el cuerpo de una página web, incluyendo otros elementos como los contenidos de la página, los textos, gráficos, enlaces, etc.

El cuerpo de una página también se puede utilizar para definir las acciones que se ejecutan ante determinados eventos. Algunos de los eventos propios de **<body>** son:

- **onload**: cuando se ha descargado el contenido completo de la página.
- **onunload**: cuando el documento va a ser descargado.
- **ononline**: cuando existe conexión a Internet.
- **onoffline**: cuando no existe conexión a Internet.
- **onafterprint**: después de imprimir el documento.
- **onbeforeprint**: después de la petición de impresión pero antes de imprimir el documento.

Una serie de atributos opcionales de **<body>** permitían configurar algunos aspectos de la apariencia del documento; actualmente están obsoletos desde la versión 4.01 y no son soportados en HTML5, dado que la idea es separar el contenido de la forma en que se presenta utilizando CSS. En cualquier caso, aún se puede probar su funcionamiento en navegadores actuales. Algunos de estos atributos obsoletos son: *background*, *bgcolor*, *text*, *link* y *vlink*.



Atributos obsoletos de <body>

background: permitía poner una imagen de fondo en la página web.

bgcolor: para indicar un color para el fondo del documento.

text: para poner un color para el texto; por defecto es negro.

link: para indicar el color de los hiperenlaces; por defecto es azul.

vlink: para el color de los hiperenlaces visitados; por defecto púrpura.

Casos prácticos

1

Creación de la primera página web

- Crea una primera página web que muestre el mensaje "HOLA MUNDO".

Solución ••

Para probar la estructuración básica del lenguaje HTML, abre un editor de texto (Bloc de notas, Gedit o similar) y edita un nuevo documento que contenga las siguientes líneas de HTML:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <title>Hola mundo</title>
</head>
<body>
<p>¡Hola, Mundo!</p>
</body>
</html>
```

¡Hola, Mundo!

Este será el mensaje que se visualizará en el navegador.

Una vez escrito, guarda el documento con extensión .html, por ejemplo, *resuelto1.html*. Para ver el resultado de la página web en un navegador, haz doble clic sobre el fichero guardado *resuelto1.html*.

Colores en hexadecimal

#000000	Negro
#FF0000	Rojo
#00FF00	Verde
#0000FF	Azul
#FFFFFF	Blanco

Colores seguros

Son aquellos que se muestran de igual manera en todos los navegadores y en todos los sistemas operativos de cualquier dispositivo. Existen 216 colores seguros para la Web.

En general, se recomienda el uso de colores seguros, para asegurar que la visualización de los colores de la web diseñada sea idéntica en todos los dispositivos, tanto en los que no puedan soportar millones de colores, como en los que tengan el valor predeterminado de 256 colores.

1.5 > Colores

El código de color RGB está compuesto por el símbolo almohadilla seguido de tres pares de dígitos hexadecimales, que indican la intensidad de los colores primarios, rojo, verde y azul, respectivamente (#RRVVAA). Cada par de cifras hexadecimal (de **00** a **FF**) proporcionan un rango de 0 a 255 valores de intensidad distintos. Cuanto mayor sea el valor del par, mayor será también la intensidad del color correspondiente a ese par. Así, combinando las proporciones de cada color primario se obtienen diferentes colores.

Hay otras formas de expresar colores en HTML. Una es utilizar una notación hexadecimal más corta, con un dígito para cada color (#RVA), lo que supone un rango inferior de intensidades de cada color, entre 0 y 15. También se puede indicar el nombre de color directamente, *Red*, *Green*, *Blue*, *White*, *Black*, *Yellow*, *Pink*, *Orange*, *Brown*, *Gray*, etc.

Los editores de HTML también suelen permitir obtener el código de color deseado escogiéndolo entre una paleta de colores.

1.6 > Espaciados, saltos de línea y párrafos

En HTML de manera ordinaria solo se reconoce un espacio entre palabra y palabra, el resto de los espacios serán ignorados por el visor. Asimismo, tampoco se respetan las tabulaciones y los retornos de carro incluidos entre el texto, es decir, estos no tienen ningún efecto a la hora de la presentación final del documento.

Para alterar este normal funcionamiento existe una directiva que permite conservar el formato indicado en los elementos de texto. La directiva es **<pre></pre>**, elemento que obliga al navegador a visualizar el texto respetando las tabulaciones, los espacios y los retornos de carro, tal y como han sido escritos en el editor.

Ejemplos

Probar el funcionamiento de espaciados y las etiquetas <pre> y

Contenido de ejemplo	Se verá como...
Esto es una frase .	Esto es una frase.
<pre><pre> este texto ha sido preformateado . </pre></pre>	este texto ha sido preformateado .
Este texto tiene un salto de línea.	Este texto tiene un salto de línea.

La etiqueta **
** representa un salto de línea, para ser utilizada en contenidos como poemas o direcciones, no para separar texto en párrafos.

La etiqueta **<hr>** representa un salto de párrafo de nivel temático, por ejemplo, un cambio de escena en una historia, o una transición a otro tema dentro de una sección; muestra una línea horizontal que ocupa todo el ancho de la página. Este elemento es de los que introducen separaciones de líneas, tanto antes como después del elemento; por tanto, en estos casos no será necesario introducir un elemento de nuevo párrafo.

La etiqueta **<blockquote>** se usa para insertar citas en forma de bloque. Permite mostrar la cita con un margen izquierdo mayor, lo que produce un efecto de sangrado al bloque de texto, diferenciándolo del resto. Admite el atributo *cite*, para indicar la fuente relacionada al texto.

La directiva **<p> </p>** representa un nuevo párrafo. Se usará al comienzo y al final de un párrafo de texto e introduce un espaciado de separación con el próximo texto o elemento que se indique. En versiones anteriores existía el parámetro *align*, que permitía indicar la forma de “justificar” el párrafo. Algunos valores posibles de este parámetro eran *left*, *right*, *center* y *justify*. En HTML 4.01 quedó obsoleto y en HTML5 no existe, no está soportado.

Etiquetas

Avance de línea y “retorno de carro”

`
`

Línea horizontal

`<hr>`

Tabulación

`<blockquote> </blockquote>`

Párrafo

`<p> </p>`

Ejemplos

Probar el funcionamiento de las etiquetas `<hr>`, `<p>` y `<blockquote>`

Contenido de ejemplo	Se verá como...
Una frase célebre es: <code><blockquote cite="http://www.w3.org/People/Berners-Lee/"></code> <code><p></code> El poder de la web está en su universalidad. El acceso para todos sin importar su discapacidad es un aspecto esencial. (Tim Berners-Lee, Director W3C e inventor de la Web) <code></p> </blockquote></code>	Una frase célebre es: El poder de la web está en su universalidad. El acceso para todos sin importar su discapacidad es un aspecto esencial. (Tim Berners-Lee, Director W3C e inventor de la Web)
Este texto tiene una línea debajo. <code><hr/></code>	Este texto tiene una línea debajo. <hr/>
<code><p align=right></code> Este es un ejemplo de un párrafo de texto justificado a la derecha. <code></p></code>	Este es un ejemplo de párrafo de texto justificado a la derecha
<code><p align=center></code> Este es un ejemplo de párrafo de texto centrado. <code></p></code>	Este es un ejemplo de párrafo de texto centrado

1.7 > Cabeceras

En un documento de HTML se pueden indicar seis tipos de elementos de cabecera, también denominados elementos de encabezado. Por medio de las directivas, `<h1> </h1>`, `<h2> </h2>`, `<h3> </h3>`, `<h4> </h4>`, `<h5> </h5>` y `<h6> </h6>`, se pueden crear títulos o encabezados, cada uno con un tamaño diferente. El texto que se escriba entre el inicio y el fin de la directiva se convertirá en texto de cabecera. El elemento `<h1>` es el de rango más alto, por lo que mostrará la cabecera de texto de mayor tamaño, mientras el `<h6>` muestra la de menor tamaño.

Los textos incluidos en elementos de “cabeceras” provocan automáticamente un doble avance de línea, separando automáticamente el siguiente elemento; por ello, no es necesario incluir detrás un elemento `
`.

Ejemplos

Probar el funcionamiento de las etiquetas de cabecera

Contenido de ejemplo	Se verá como...
<code><h1>Texto de Prueba</h1></code>	Texto de prueba
<code><h2>Texto de Prueba</h2></code>	Texto de prueba
<code><h3>Texto de Prueba</h3></code>	Texto de prueba
<code><h4>Texto de Prueba</h4></code>	Texto de prueba
<code><h5>Texto de Prueba</h5></code>	Texto de prueba
<code><h6>Texto de Prueba</h6></code>	Texto de prueba
<code><h3>Pagina de José</h3>Esta es mi página personal.</code>	Pagina de José Esta es mi página personal.
<code><h3>Pagina de José</h3>
Esta es mi página personal.</code>	Pagina de José Esta es mi página personal.

1.8 > Semántica en textos

Son elementos de HTML que permiten dar algún significado a una parte del contenido de un texto, mostrando ese fragmento en negrita (), en letra cursiva (<i> </i>), subrayado (<u> </u>), etc. Estos son solo algunos ejemplos de los diversos elementos que se pueden utilizar en HTML5 para dar cierto contenido semántico a las diferentes partes de un texto.

En HTML5 han desaparecido las etiquetas de elementos que permitían dar un formato de presentación a los contenidos, dado que para ello se deben utilizar las hojas de estilo CSS, con las que podemos separar el contenido a mostrar de la forma en la cual se muestra.

Una etiqueta eliminada, ya obsoleta en la versión 4.01 de html, es <center> </center>, que permitía centrar un texto.

También se ha eliminado , elemento que permitía variar el tamaño, el color y el tipo de letra de un texto, a través de sus atributos size, color y face.

Etiquetas de estilo a textos

Existe una serie de etiquetas muy similares entre sí y con una funcionalidad muy simple: sirven para dar un estilo de letra especial a un fragmento de texto.

La diferencia es el tipo de letra que se utiliza en cada caso. Son las etiquetas: <cite>, , <dfn>, , <code>, <samp>, <kbd> y <var>.

Ejemplos

Probar elementos HTML relacionados con los elementos semánticos para el texto

Etiqueta	Semántica	Contenido de ejemplo	Resultado
 	Negrita	 Texto de prueba 	Texto de prueba
<i> </i> <cite> </cite> 	Cursiva	<i> Texto de prueba </i>	Texto de prueba
<u> </u> <ins> </ins>	Subrayado Insertar	<u> Texto de prueba </u>	Texto de prueba
<s> </s> 	Tachado Borrar	<s> Texto de prueba </s>	Texto de prueba
	Superíndice	^{Texto de prueba}	Texto de prueba
	Subíndice	_{Texto de prueba}	Texto de prueba
<center> </center>	Variable	<center> Texto de prueba </center>	Texto de prueba
 	Fuente	 Texto de prueba	Texto de prueba

Menú

- Inicio
- Noticias
- Artículos
- Contacto

Instrucciones

1. Escuchar atentamente
2. Anotar ideas principales
3. Realizar esquema o resumen

Character name:

Ejemplos de listas de elementos.

1.9 > Listas de elementos

Hay varios tipos de listas: numeradas, sin numerar, de definición, etc.

Las **listas numeradas** representarán los elementos de la lista numerando cada uno de ellos según el lugar que ocupan en la lista. Para este tipo de lista se utiliza la directiva `` ``. Cada uno de los elementos de la lista irá precedido de la directiva `` ``.

La directiva `` puede llevar los siguientes parámetros:

- **start** = número — Indica qué número será el primero de la lista. Si no se indica se entiende que empezará por el número 1.
- **type** = tipo — Indica el tipo de numeración utilizada. Si no se indica se entiende que será una lista ordenada numéricamente.

Los tipos posibles en HTML5 son:

- **1**: para expresar números (1, 2, 3, etc.).
- **a**: para letras minúsculas. (a, b, c, etc.).
- **A**: para letras mayúsculas. (A, B, C, etc.).
- **i**: números romanos en minúsculas. (i, ii, iii, etc.).
- **I**: números romanos en mayúsculas. (I, II, III, etc.).

Las **listas sin numerar** representan los elementos de la lista con un “topo” o marca que antecede a cada uno de ellos. Se utiliza la directiva `` `` para delimitar la lista, y `` `` para indicar cada uno de los elementos. La directiva `` puede contener el parámetro **type** que indica la forma del “topo” o marca que antecede a cada elemento de la lista. Los valores de **type** pueden ser “disk”, “circle” o “square”, con lo que el topo o marca puede ser un disco, un círculo o un cuadrado.

Las **listas de definición o glosario** muestran los elementos como un diccionario, expresan términos junto con su respectiva definición. Para crear una lista de definición se utiliza la directiva `<dl>` `</dl>`. Los términos se indican en la directiva `<dt>` `</dt>`, y su definición dentro de la directiva `<dd>` `</dd>`.

Los diferentes tipos de listas se pueden anidar entre sí, es decir, se puede incluir una lista dentro de otra, lo que permite conseguir listas estructuradas más complejas que mezclan los diferentes tipos de listas.

Ejemplos

Glosario de términos creado con las etiquetas `<dl>`, `<dt>` y `<dd>`

```
<dl>
<dt>SGML</dt> <dd>Metalenguaje para la definición de otros lenguajes de marcado</dd>
<dt>XML</dt> <dd>Lenguaje basado en SGML y que se emplea para describir datos</dd> </dl>
```

SGML:

Metalenguaje para la definición de otros lenguajes de marcado

XML:

Lenguaje basado en SGML y que se emplea para describir datos

1.10 > Imágenes

Hay varios formatos de imágenes que son manejados y reconocidos por los navegadores actuales, como son los mapas de bits estáticos (archivos PNG, GIF, JPEG), los documentos vectoriales de una sola página (archivos PDF de una sola página, o archivos XML con un elemento raíz SVG), los mapas de bits animados (APNG, GIF animados), los gráficos vectoriales animados (archivos XML con animación SMIL), etc. Sin embargo, se excluyen y no se reconocen como imágenes archivos SVG con guion, archivos PDF de varias páginas, archivos interactivos MNG, documentos HTML, ni los documentos de texto sin formato.

El elemento **** permite representar una imagen con la ayuda de su atributo obligatorio **src**, donde se indicará la dirección válida donde se encuentre la imagen a visualizar, una ruta relativa, si la imagen reside localmente, o una dirección URL si es una imagen externa, que reside en otra página web.

Con el atributo **alt** podemos indicar un texto alternativo, que represente el contenido de la imagen, muy útil cuando el navegador no sea capaz de visualizarla o bien tenga deshabilitado la carga de imágenes.

Las etiquetas **<figure>** y **<figcaption>** son nuevas en HTML5. Permiten englobar una imagen (u otro tipo de contenido) y su información adicional o leyenda.

Formatos de imágenes

PNG: *Portable Network Graphics*

GIF: *Graphics Interface Format*

JPEG: *Joint Photographic Experts Group*



APNG: *Animated Portable Network Graphics*

SVG: *Scalable Vector Graphics*

MNG: *Multiple-image Network Graphics*

Ejemplos

Probar la visualización de imágenes con la etiqueta

Contenido de ejemplo	Se verá como...
<pre></pre>	
Si el visor no pudiese visualizar esta imagen, al menos obtendría el texto alternativo, "¡Cuidado!".	
<pre><figure>
 <figcaption>Fig. 1.- Esto se ve desde mi terraza. </figcaption> </figure></pre>	 <p>Fig. 1.- Esto se ve desde mi terraza.</p>

Contexto de navegación

Hace referencia al sitio donde un navegador puede visualizar la información de una página web. Así, algunos posibles valores válidos para el atributo **target** son:

"_blank": nueva ventana.

"_self": misma ventana.

"_parent": ventana padre.

"_top": ventana superior.

1.11 > Vínculos o hiperenlaces

Un vínculo o hiperenlace es un elemento del lenguaje HTML que permite acceder a otro recurso, es decir, permite navegar desde una página web a otra, ubicarse en un lugar concreto dentro de una página web, acceder a un archivo, etc.

Para incluir un hiperenlace se utiliza la directiva `<a> `, de tal manera que el texto o imagen que se encuentre dentro de esta directiva se puede convertir en un hipervínculo, es decir, que si se pulsa con el ratón sobre él, se intentará acceder a la página web o al recurso indicado en la dirección vinculada a dicho texto o imagen.

Si el hiperenlace está indicado por un texto, este aparecerá subrayado y en un color distinto; si se trata de una imagen, en ella puede aparecer un borde.

El atributo **href** permite crear el hiperenlace. Para ello hay que indicar una URL, a la cual se accederá cuando el vínculo se pulse con el ratón. Si en el elemento `<a>` no se indica este atributo **href**, entonces el elemento representa un marcador de posición a la que otro hiperenlace podrá hacer referencia en su respectivo atributo **href**.

El atributo **target** es opcional y sirve para indicar cuál será el destino donde se mostrará la información disponible en la dirección vinculada.

Ancias y vínculos internos

Un vínculo interno es un hiperenlace que permite acceder a un lugar concreto dentro de la propia página web. Para ello se debe establecer previamente un ancla, es decir, el lugar de la página web al cual posteriormente se pretende poder acceder referenciándolo por medio de un hiperenlace.

En HTML 4.01 para definir un ancla se solía asignar un valor al atributo **name**, dentro de una etiqueta `<a>`, marcando así el lugar de comienzo de una parte del documento al cual posteriormente se referenciará.

En HTML5 el atributo **name** no existe, no está soportado, pero de manera muy parecida se pueden establecer anclas utilizando el atributo común **id** en lugar de **name**, indicando un identificador para el ancla que se defina con la etiqueta `<a>`, de la siguiente manera:

```
<a id="ancla1"> TEXTO ANCLADO </a>
```

Luego hay que generar un hiperenlace al ancla creado dentro de la página, anteponiendo el carácter almohadilla (#) al valor indicado en el atributo **id**, como valor del atributo **href** del hiperenlace (`<a>`), así:

```
<a href="#ancla1">ARRIBA</a>
```


También se puede hacer un enlace a una parte concreta de una página diferente, para ello es necesario definir el pertinente ancla en la página a la que se quiere acceder y definir el hiperenlace incluyendo la almohadilla, colocándola entre la URL de la página y el nombre del ancla:

```
<a href = http://www.dominio.es/pag.html#ancla2">
Primera Parte</a>
```


También se puede crear un hipervínculo a un fichero que no sea una página web. En este caso, el navegador preguntará si se desea abrir el fichero o guardarlo en el ordenador, permitiendo a los visitantes de la página descargar ficheros desde esa página web a su ordenador.

Ejemplos

Probar la creación de hipervínculos con la etiqueta <a>

Contenido de ejemplo	Se verá como...
<pre> Pulse para ir a la página de EducaMadrid</pre>	<p>Pulse para ir a la página de EducaMadrid</p>
<p>Si se sitúa y se pulsa el ratón encima de la frase subrayada, el navegador accederá a la página web indicada en el parámetro <i>href</i>, es decir, se accederá a la página situada en http://www.educa.madrid.org/</p>	
<pre>Para buscar en Internet : </pre>	<p>Para buscar en Internet : </p>
<p>Pulsando sobre la imagen se accedería a la página situada en http://www.yahoo.com/</p>	
<pre><nav> Inicio Noticias Avisos </nav></pre>	<ul style="list-style-type: none"> • Inicio • Noticias • Avisos
<p>Ejemplo de una barra de herramientas de navegación que consiste en enlaces a diferentes páginas.</p>	
<pre>Pulsa aquí para llevarte una copia del manual.</pre>	<p>Pulsa aquí para llevarte una copia del manual.</p>
<p>Este ejemplo permitirá al navegante descargar o abrir el fichero "manual.zip" al pulsar el hipervínculo.</p>	
<pre>Noticias de actualidad</pre>	<p>Ejemplo que muestra un hipervínculo que permite abrir un grupo de noticias.</p>
<pre> Envíame tus sugerencias</pre>	<p>Ejemplo que muestra un hipervínculo que permite el envío de un correo electrónico.</p>

Uso básico de *colspan* y *rowspan*

```
<table border="1">
<tr><td>1</td></tr>
<tr><td>2</td></tr>
</table><br>
<table border="1">
<tr><td>1</td>
    <td>2</td>
</tr>
</table><br>
<table border="1">
<tr><td>1</td>
    <td>2</td></tr>
<tr><td>3</td>
    <td>4</td></tr>
</table><br>
<table border="1">
<tr><td rowspan="2">1</td>
    <td>2</td></tr>
<tr><td>3</td></tr>
</table><br>
<table border="1">
<tr><td colspan="2">1
    </td></tr>
<tr><td>3</td>
    <td>4</td></tr>
</table>
```

1.12 > Tablas

Las tablas son un elemento del HTML que permiten representar datos tabulados con más de una dimensión, en forma de matriz, es decir, con la información dispuesta en filas y columnas.

Históricamente, muchos diseñadores de páginas web también utilizaban las tablas como una herramienta para controlar el diseño, aunque esto hacía difícil extraer los posibles datos tabulados almacenados. Así, a los usuarios de herramientas de accesibilidad, como algunos lectores de pantalla, les resulta especialmente difícil navegar por páginas que contienen tablas utilizadas solo para maquetar la presentación de la página web.

Un objetivo del HTML5 es separar el contenido a mostrar en una página web de la forma en que es presentado, recogiendo esta última mediante hojas de estilo CSS; por ello, aunque las tablas permiten ordenar los contenidos de una página web, **no deben ser utilizadas como herramienta de diseño.**

En cualquier caso, si una tabla se pretende utilizar para ayuda al diseño, el borrador de HTML5 sugiere asignar el valor “*presentation*” en el atributo **role**, para indicar que el autor creó la tabla exclusivamente como elemento de diseño, y así ayudar a las herramientas que desean extraer los datos tabulados que contengan las tablas de los documentos.

Una tabla se define mediante la directiva **<table>** **</table>**. En versiones anteriores esta etiqueta admitía los atributos *align*, *bgcolor*, *cellspacing*, *cellpadding*, *width*, *frame*, *rules* y *summary*. En cambio, HTML5 solo soporta el atributo **border** dentro de la etiqueta **<table>**, que especifica si las celdas de la tabla deben tener fronteras o no, su valor puede ser “1” o “”, y ambos indican que se muestre el borde de las celdas.

El primer elemento que puede aparecer dentro de una tabla es el representado por la etiqueta **<caption>** **</caption>**, que representa el título de la tabla que lo contiene y permite contextualizar la tabla para hacer más fácil de entender los datos que contenga.

Para definir la estructura de filas y columnas que componen la tabla se utiliza la directiva **<tr>** **</tr>**, que define una nueva fila. Luego, dentro de cada fila se incluyen elementos **<td>** **</td>**, que permiten generar columnas con datos dentro de una fila, o bien **<th>** **</th>** para hacer columnas de cabecera donde el texto contenido se muestra centrado, en negrita y de tamaño ligeramente superior.

Es posible combinar varias celdas creadas con **<td>** y **<th>** a través de los atributos **rowspan** y **colspan**, indicando un valor que represente el número de filas o columnas, respectivamente, que se desea combinar.

Hay otros elementos que se pueden incluir dentro de una tabla que afectan a su definición: **<colgroup>** para representar un grupo de columnas y **<col>** para indicar las columnas que componen un grupo. También hay elementos que permiten representar la cabecera, los bloques de filas que forman la tabla y el pie de la misma; estos elementos del lenguaje HTML son **<thead>**, **<tbody>** y **<tfoot>**, respectivamente.

1.13 > Formularios

Cuando se desea recopilar cierta información que puedan aportar los visitantes de una web, por ejemplo sus datos personales de registro, o bien su opinión sobre una encuesta, o cualquier otro dato de interés, una herramienta que facilita esta tarea, son los formularios.

Los formularios se componen de varias entradas o elementos con los que el usuario puede interactuar, como cajas de texto, casillas de verificación (*checkboxes*), de opción (*radio buttons*), listas desplegadas, etc., tantas como diferentes datos se quieran obtener. Toda la información recopilada podrá ser enviada, bien a través de una URL, a un programa que sea capaz de procesarla y resida en un servidor web, bien a través de un botón de validación, al correo del administrador de la página, por ejemplo. Para ello es necesario tener instalado un sistema cliente de correo tipo Thunderbird, Outlook, Apple Mail, etc.

Un formulario se define mediante la directiva `<form>` `</form>` y dentro de ella se incluirán el resto de elementos que compongan el formulario de entrada. A la etiqueta de apertura `<form>` le pueden acompañar varios atributos relacionados con el envío del formulario como son: *action*, para indicar el programa o lugar al que se van a enviar los datos del formulario, *method*, que indica el método de transferencia de los datos en http, *enctype*, que permite indicar cómo deben ser codificados los datos al enviarlos a un servidor cuando se utiliza el método post, y *target*, que especifica dónde se debe mostrar la respuesta que se recibe después de enviar el formulario. También se utiliza el atributo *name*, para asociarle un nombre unívoco que permita diferenciarle entre otros formularios. Veamos ahora los elementos que pueden contener los formularios.

La directiva `<fieldset>` `</fieldset>` sirve para agrupar un conjunto de elementos dentro de un formulario bajo un nombre, nombre que se indica entre las etiquetas `<legend>` `</legend>`, como primer elemento hijo del grupo; el resto de los descendientes serán las entradas que forman el grupo de elementos del formulario agrupados en el conjunto "fieldset".

Con el elemento `<label>` `</label>` podemos etiquetar y enmarcar con un borde los controles de un formulario, organizando así el contenido.

Atributo *method*

Los posibles métodos a indicar en el atributo *method* son:

"post": envía los datos de manera codificada al usuario.

"get": envía los datos agregándolos a la dirección URL detrás de un signo de interrogación.

Los posibles valores de encriptación del atributo *enctype* son:

"application/x-www-form-urlencoded"

"multipart/form-data"

"text/plain"

Los valores válidos para el atributo *target* son los mismos que para los hiperenlaces:

"_blank": nueva ventana.

"_self": misma ventana.

"_parent": ventana padre.

"_top": ventana superior.

Ejemplos

Crear un formulario simple

```
<html>
<head> <title> Ejemplo básico de un formulario </title> </head>
<body>
<h1> Formulario básico</h1>
<form action="http://algunsitio.com" method="post">
Nombre: <input type="text" name=" nombre "> <br>
<input type="submit"><input type="reset">
</form>
</body>
</html>
```

Formulario básico

Nombre:

Etiqueta <input>

La sintaxis general de un elemento de entrada <input> suele ser así:

```
<input
type="TipoDeEntrada"
name="Nombre"
value="Valor">
```

Atributos comunes de los controles de un formulario

name: para dar nombre a un elemento de entrada del formulario.

disabled: permite desactivar los controles que conforman un formulario si se incluye este atributo lógico. Así se consigue que no tengan efectos los eventos de ratón que el usuario intente realizar sobre las entradas del formulario desactivadas con este atributo.

autofocus: atributo lógico para indicar que un control sea seleccionado en el momento de cargar la página, lo que permite al usuario comenzar a escribir sin tener que seleccionar con el ratón alguna entrada del formulario. Se suele indicar en el primer control del formulario.

maxlength: establece el límite en el número de caracteres de una entrada de usuario; el valor del atributo debe ser un entero positivo.

Otros atributos de formularios:

checked: especifica que un elemento de entrada aparezca pre-seleccionado cuando se cargue la página (para entradas "checkbox" o "radio").

src: indica la URL de una imagen.

alt: establece un texto alternativo para la imagen.

La etiqueta <input> es una etiqueta esencial para los formularios, ya que permite crear diferentes elementos de entrada interactivos, con los que el usuario puede editar la información del formulario. Su atributo *name* especifica el nombre del elemento y permitirá reconocer posteriormente el valor asignado a las diferentes entradas del formulario. Si el usuario no introdujo ningún valor, el valor será el predeterminado por el atributo *value*. Su atributo *type* es el más importante, ya que hace posible especificar diferentes controles en el formulario, de tal forma que según el tipo especificado se creará un tipo de entrada u otro en el formulario. Los posibles valores de *type* son:

- *hidden*: es una entrada oculta, cuyo valor se envía con el formulario.
- *text* y *search*: controles para editar una línea de texto. El tamaño de la casilla se puede definir con el atributo *size* y la extensión máxima del texto con el atributo *maxlength*. Solo el estilo les diferencia.
- *tel*: representa un control para introducir un número de teléfono. A diferencia de los tipos *url* y *email*, el tipo *tel* no obliga a cumplir una sintaxis particular, dado que los campos de un número de teléfono pueden tener múltiples formas válidas. Si se necesita hacer cumplir un formato en particular será necesario utilizar el atributo *pattern* del elemento <input> o el método `setCustomValidity()`, para realizar un mecanismo de validación en el lado del cliente.
- *url*: una dirección de un localizador de un recurso de Internet.
- *email*: una dirección de correo electrónico o una lista de direcciones.
- *password*: una línea de texto que oculta el valor, para contraseñas.
- *datetime*: una fecha y una hora, consta de año, mes, día, hora, minuto, segundo y fracción de segundo, con la zona horaria en UTC.
- *date*: una fecha compuesta de año, mes y día.
- *month*: consta de un mes y un año, sin zona horaria.
- *week*: un número de semanas, sin zona horaria.
- *time*: consta de hora, minuto, segundo y fracción, sin zona horaria.
- *datetime-local*: una fecha y una hora, consta de año, mes, día, hora, minuto, segundo y fracción de segundo, sin zona horaria.
- *number*: un valor numérico, puede ser entre un mínimo y un máximo.
- *range*: un valor numérico elegido entre un rango de valores definido.
- *color*: un color RGB (rojo, verde, azul) con 8-bit, valor entre 0 y 255.
- *checkbox*: un conjunto de valores de una lista de casillas de selección.
- *radio*: un valor elegido entre varias opciones enumeradas.
- *file*: para enviar con el formulario una lista de nombres de archivos. Los tipos de archivo (audio, vídeo, image, *MIME_type*) válidos deben especificarse utilizando el atributo *accept* de la etiqueta <form>.
- *submit*: botón de confirmación para enviar los datos del formulario. El texto del botón puede definirse usando el atributo *value*.
- *image*: botón de imagen personalizado; la ubicación de la imagen se define con el atributo *src*.
- *reset*: botón de restauración, restablece los valores predeterminados del formulario y elimina la información introducida.
- *button*: botón con ningún comportamiento predeterminado.

La etiqueta <textarea> se usa para definir un cuadro de texto más grande que el obtenido con la etiqueta <input> utilizando los valores *text* y *search* en su atributo *type*.

Algunos atributos válidos de la etiqueta `<textarea>` son:

- *name*: representa el nombre asociado con este área de texto; permite su identificación posterior entre los elementos del formulario.
- *cols*: es el número de caracteres que puede contener cada línea.
- *rows*: representa el número de líneas del área de texto.
- *readonly*: atributo lógico para impedir al usuario editar el texto.

La etiqueta `<select>` sirve para crear una lista desplegable con opciones, que serán especificados con diferentes etiquetas `<option>` dentro de ella. Los atributos de esta etiqueta `<select>` son:

- *name*: representa el nombre asociado con la lista desplegable y permitirá diferenciarlo entre todos los elementos del formulario.
- *size*: representa el número de opciones a mostrar en la lista, valor que puede ser diferente al número de opciones que componen la lista.
- *multiple*: atributo lógico que permite seleccionar varias opciones de la lista; si no está presente, solo se podrá seleccionar una única opción.

Rango de valores

Las entradas (`<input>`) de tipo *number* y *range* permiten utilizar los atributos *min* y *max* para establecer un rango de valores válido en la entrada. Ejemplo:

```
<input type="number"
name="EdadEmpleado" min="16"
max="67">
```

Etiqueta	Atributo	Valor	Resultado	Efecto visual
<FORM>... </FORM>	method	post get		
	action		Envía a la dirección mostrada	
	enctype		Especifica el tipo de código	
<INPUT>	type	submit	Realiza la ACTION de la etiqueta <FORM>	<input type="submit" value="Enviar consulta"/>
		text	Línea simple de texto cuya longitud se especifica por el atributo SIZE	<input type="text"/>
		Reset	Elimina el contenido del formulario	<input type="button" value="Restablecer"/>
		Radio	Botón de radio	<input type="radio"/>
		Checkbox	Casilla de selección	<input type="checkbox"/>
	name		Nombre	
	size		Tamaño del texto	
<TEXTAREA>... </TEXTAREA>	name		Casilla de texto	<input type="text"/>
	rows			
	cols			
<SELECT> <OPTION>... </OPTION> </SELECT>	name			<input type="text" value="Opción 1"/>
	multiple		Múltiples selecciones posibles	<input type="text" value="Opcion1"/>
<OPTION>... </OPTION>	selected	Elección predeterminada		<input type="text" value="Opcion1"/> <input type="text" value="Opcion2"/> <input type="text" value="Opcion3"/>
	value	Valor forzado		

Cuadro resumen de etiquetas básicas de formularios.

Casos prácticos

Creación de un formulario dentro de una tabla

- Crea un formulario que permita registrar los datos de un usuario.

Solución • Estudia las siguientes líneas de HTML y edita un nuevo documento, que contenga cada una de las entradas del formulario de registro dentro de cada fila de una tabla, tal como muestra la imagen de la derecha:

```
<!DOCTYPE html>
<html>
<body>
<form method=post action="caso_form.asp">
<h3>Registro de usuario</h3>
<table border=0>
  <tr><td>Nombre</td>
    <td><input type=text name="nombre"></td>
  </tr>
  <tr><td>Apellidos</td>
    <td><input type=text name="apellido"></td>
  </tr>
  <tr><td>Género</td>
    <td>Hombre:
      <input type=radio name="género" value="m"><br>
      Mujer:
      <input type=radio name="género" value="f"></td>
  </tr>
  <tr><td>Edad</td>
    <td><input name="edad" type="number" min="0"></td>
  </tr>
  <tr><td>Ocupación</td>
    <td>
      <select name="ocupación">
        <option value="1">Trabajador</option>
        <option value="2">Estudiante</option>
        <option value="3">Jubilado</option>
        <option value="otro">Otro</option>
      </select>
    </td>
  </tr>
  <tr>
    <td>Comentarios</td>
    <td><textarea rows="3" name="comentarios">Escriba aquí sus comentarios</textarea>
  </td>
  </tr>
  <tr>
    <td colspan=2>
      <input type="submit" value="Enviar">
    </td>
  </tr>
</table> </form></body></html>
```

Registro de usuario

Nombre	<input type="text"/>
Apellidos	<input type="text"/>
Género	Hombre: <input type="radio"/> Mujer: <input type="radio"/>
Edad	<input type="text"/>
Ocupación	Trabajador ▼
Comentarios	<div>Escriba aquí sus comentarios</div>
<input type="button" value="Enviar"/>	

Escribe y guarda el documento con extensión .html, por ejemplo, *resuelto2.html*. Abre la página web con diferentes navegadores y observa si hay diferencias entre ellos.

Casos prácticos

3

Creación de un formulario con agrupación de elementos y patrones de entrada de datos

•• Crea un formulario que utilice las etiquetas *fieldset*, *legend*, *label*, y utiliza patrones de filtro con la etiqueta *pattern*.

Solución •• Edita un nuevo documento con las siguientes líneas de HTML y comprueba el funcionamiento de *pattern* guardándolo como *patrones.html* y abriéndolo utilizando diferentes navegadores.

```
<!DOCTYPE html>
<html>
<body>
<form method=post action="caso_form.asp">
<fieldset name="Club de jugadores" >
  <legend> <label> <input type=checkbox name="club" checked>
  Utilizar tarjeta del club </label> </legend>
  <p><label>Nombre de la tarjeta:
  <input name="nombreclub" required></label></p>
<fieldset name="numfields">
  <legend> <label> <input name="tipoclub" type=radio checked>
  Mi tarjeta contiene números</label> </legend>
  <p><label>Número de tarjeta:
  <input name="numeroclub" required pattern="[-0-9]+"> </label></p> </fieldset>
<fieldset name="letfields">
  <legend> <label> <input type=radio name="tipoclub">
  Mi tarjeta contiene letras </label> </legend>
  <p><label>Código de tarjeta:
  <input name="letraclub" required pattern="[A-Za-z]+"> </label></p></fieldset>
</fieldset>
<input type="submit" value=" Enviar">
</form>
</body>
</html>
```

Actividades propuestas

1•• Realiza una página web en HTML llamada *primera.html*, que muestre un rotulo de cabecera y una imagen de fondo, imagen que se guardará previamente en una carpeta local denominada imágenes. La página se direccionará a otra página web denominada *segunda.html* automáticamente cuando hayan transcurrido cinco segundos. Esta segunda página web mostrará otro rótulo de cabecera, una lista numerada y un enlace a *primera.html*.

2•• Coloca en una página web imágenes representativas de cada uno de los navegadores más utilizados: Internet Explorer, Mozilla Firefox, Chrome, Safari y Opera. Para ello, ordénalas en filas y columnas en una tabla poniendo las imágenes centradas una debajo de otra. Añade el nombre de cada navegador debajo de cada imagen y, a través de hiperenlaces, vincula cada nombre a su página web oficial.

3•• Modifica la página anterior, convirtiendo cada imagen en un hiperenlace, que también permita acceder a la página web oficial del navegador de Internet que representa. Separa las imágenes verticalmente hasta que sea necesario el uso del desplazamiento vertical para bajar al final de la página web, para poder ver todas las imágenes. Añade un ancla debajo de la última imagen que permita subir al principio de la página.

2 >> Hojas de estilos CSS

Dado que en HTML no solo se indican los contenidos que se quieren visualizar, sino que se mezclan con las instrucciones de formato y estilo, el W3C creó un sistema mediante el cual las instrucciones del estilo y formato estuvieran separadas del contenido. Así nació CSS (*Cascading Style Sheets*, hojas de estilo en cascada.), un mecanismo para dotar de estilo a los elementos HTML, que desaconseja mezclar su contenido con la forma en que estos son presentados.

2.1 > Introducción

Por tanto, el objetivo fundamental de CSS es **separar el contenido y la estructura de un documento HTML, de su estilo y formato de presentación**. La versión recomendada era CSS 2.1, pero actualmente está en desarrollo el nuevo estándar CSS3, y el W3C recomienda el uso de los módulos CSS3 ya implementados.

Para entender el objetivo de CSS, pongamos un ejemplo: sabemos que el elemento de HTML `<h1>` indica que un bloque de texto es un encabezamiento y que es más importante (de mayor tamaño) que un bloque etiquetado como `<h2>`. Versiones más antiguas de HTML permitían atributos dentro de la etiqueta para añadir estilo extra (como el color o el tamaño de fuente). Pero cuando se utiliza CSS, la etiqueta `<h1>` no debería proporcionar información sobre cómo va a ser visualizado, solamente indicar la estructura del documento. La información de estilo, recogida por ejemplo en una hoja de estilos, debe especificar cómo se ha de mostrar `<h1>`, es decir, su color, fuente, alineación del texto, tamaño y otras características visuales.

Por consiguiente, CSS es la herramienta con la que se debe dar formato y estilo a los elementos de HTML. La información de estilo puede ser recogida tanto en un documento separado del HTML, opción recomendada, como en el propio documento HTML. En este último caso pueden definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo *style*, como más adelante se explica.

2.2 > Hoja de estilos

Una hoja de estilos es un conjunto de especificaciones que definen el formato con el que presentar los elementos HTML de una página web.

Está compuesta por **reglas de estilo**, cada una de las cuales declara los formatos que adoptarán los elementos de una página web. Así, un conjunto de reglas de estilo conforman una hoja de estilos, información que normalmente se recoge en un fichero con extensión `.css`, aunque existen otras maneras de recoger las reglas que den estilo a los elementos que componen un documento HTML.

Una regla de estilo tiene dos partes: el **selector**, que determina a qué elementos se asignará el estilo definido en la regla, y la **declaración**, que está compuesta por uno o más pares (*propiedad:valor*), que asignan un valor a la propiedad de los elementos seleccionados por el selector.



Estructura de una regla de estilo.

2.3 > Formas de aplicar estilo

CSS proporciona tres formas diferentes para aplicar las reglas de estilo a los elementos HTML de una página web:

Estilo local o en línea

Este método permite insertar el estilo, directamente, dentro de cada etiqueta HTML, añadiendo en ellas el atributo común *style*.

Es la forma menos adecuada, dado que especificar el estilo dentro de cada etiqueta del documento HTML, es contrario al objetivo de CSS, y además se convierte en una tarea larga, tediosa y poco elegante de expresar el formato y estilo de una página web. Solo se recomienda usar este modo de trabajo, de manera excepcional, por ejemplo cuando se necesita aplicar un estilo o formato de manera urgente en una página.

Hoja de estilos interna

Es una hoja de estilo que está incrustada (etiqueta `<style>`) dentro de un documento HTML. Suele ponerse dentro del elemento `<head>`, aunque puede ir en cualquier parte de la página. Esta forma permite agrupar toda la información sobre el estilo, pero dentro del mismo fichero HTML que contiene el contenido. Una desventaja es que hay que modificar todas las páginas que conforman un sitio cuando se quiera realizar un cambio que afecte a todas ellas; por lo tanto, será útil cuando se aplique estilo a una única página.

Hoja de estilos externa

Es una hoja de estilo que está almacenada en un archivo diferente al archivo donde se almacena el código HTML de la página web. Esta es la mejor manera de aplicar estilo y la que sigue más fielmente el objetivo de CSS, porque separa completamente las reglas de estilo y formato para la página HTML, de su contenido y estructuración básica.

La etiqueta `<link>` se coloca en la cabecera `<head>` del documento y permite enlazarlo con otro recurso.

Se debe indicar el atributo *rel*, para especificar la relación entre el documento y el recurso.

El atributo opcional *type* especifica el tipo del documento a vincular, para una hoja de estilos *“text/css”*.

El atributo *href* debe indicar una URL, la del fichero CSS que tenga la definición del estilo.

Es recomendable utilizar hojas de estilo externas, pero sobre todo en sitios web que utilizan un mismo estilo para todas sus páginas, dado que una modificación de una regla de estilo afectará a todas ellas.

Uso de hoja de estilo externa `<link>`

```
<html>
<head>
...
<link
rel="stylesheet"
type="text/css"
href="hojaEstilo.css"
...
</head>
<body>
...
</body>
</html>
```

Comentarios

Cuando se crea una hoja de estilos (interna o externa), para facilitar la revisión del código CSS es conveniente añadir comentarios entre las reglas de estilo. Para incluir un comentario es necesario incluir el texto a comentar entre los caracteres */* y */*, por ejemplo:

```
/* Esta es la forma de incluir
un comentario válido entre las
reglas de estilo CSS */
```

Actividades propuestas

4.. Indica una etiqueta HTML que permita vincular la hoja de estilos del fichero *“miHoja.css”*.

Alto y ancho de objetos CSS

Las propiedades generales *height* y *width* permiten establecer el alto y el ancho de los elementos.

```
#tamaño100
{height:100px;
width:100px;}
```

2.4 > Conceptos fundamentales

Los conceptos fundamentales que permiten realizar hojas de estilos en CSS son:

Identificadores

Todos los elementos HTML tienen un atributo opcional llamado identificador (*id*), cuya finalidad es poder identificar a cada uno de los diferentes elementos que componen la página web. Con estos identificadores podemos aplicar un estilo exclusivo a cada etiqueta, según se seleccione su identificador en el selector de una regla CSS.

Para identificar un determinado elemento en el código HTML, es necesario darle un nombre por medio del atributo *id*, y, posteriormente, será necesario acceder a este elemento desde una regla CSS para aplicarle el diseño. Esto se hace a través del selector, que se formará con el nombre de la etiqueta seguido del símbolo # y del nombre del atributo *id*.

Clases

Existe otro atributo que se puede utilizar en todos los elementos HTML, el atributo *class*. Su utilidad es la de poder agrupar los elementos por clases o grupos para que, posteriormente, CSS los pueda seleccionar y distinguirlos de los demás. El uso de *class* también es opcional, pero, como el atributo *id*, es muy recomendable y permite precisar selectores.

Ejemplos

Aplicar estilo de manera local, con una hoja de estilos interna y a través de clases

Este es un ejemplo de código HTML que utiliza una **hoja de estilo local** al párrafo, haciendo uso del atributo *style*:

```
<html> <body> <p style="color:green"> HOLA MUNDO </p>
</body> </html>
```



Este ejemplo utiliza una **hoja de estilos interna**, a través de la etiqueta *<style>*:

```
<html> <head> <title>Mi primera página con estilo</title>
  <style type="text/css">
    body {color: purple;
    background-color: #d8da3d}</style>
</head> <body>
```



En este otro ejemplo utiliza **clases** para aplicar estilo, a través del atributo *class*; su sintaxis en HTML sería:

```
<p class="bordeRojo">HOLA</p>
```

Un elemento puede pertenecer a varias clases a la vez, ya que estas no son excluyentes. Para ello se especifica dentro del atributo *class* los nombres de las clases separados por espacios:

```
<p class="letraAzul bordeAzul">HOLA</p>
```

Si un elemento HTML ha sido identificado siguiendo este ejemplo, puedes darle formato con reglas de estilo así:

```
.bordeRojo {
  border:solid 1px #000000;
  border-color:red;}
.bordeAzul {
  border:solid 1px #000000;
  border-color:blue;}
.letraAzul {color:blue;}
```

Casos prácticos

4

Creación de estilo usando identificadores

•• Crea un estilo, poniendo en negrita el primer párrafo, utilizando el atributo *id* dentro del elemento `<p>`.

Solución •• Para probar cómo dar estilo a elementos, abre un editor de texto (Bloc de notas, Gedit o similar) y edita dos documentos, uno que incluya las líneas de "Código HTML", y el otro con el "Código CSS" indicado. Hay que identificar con un nombre a la etiqueta `<p>` del primer párrafo y añadir un identificador al selector del código CSS.

Código HTML:

```
<link rel="stylesheet"
type="text/css"
href="negrita.css">...
<p id="EnNegrita">
Párrafo 1</p>
<p>Párrafo 2</p>
<p>Párrafo 3</p>...
```

Código CSS:

```
p#EnNegrita {font-weight:bold;}
```

Párrafo 1

Párrafo 2

Párrafo 3

Comprueba que incluyes el atributo *id* en el código HTML y utiliza su valor en el código CSS, incluyéndolo en el selector. Al ser un identificador, tienes que anteponerle la almohadilla al indicarlo en la regla de estilo. Así se puede dar un diseño exclusivo a cualquier elemento de una página web, tablas, listas, etc.; en este caso práctico se hace solo sobre el primer párrafo. Crea identificadores para los otros párrafos y aplícales reglas de estilo CSS.

Pseudo-clases

Las pseudo-clases permiten clasificar los elementos HTML según el estado en el que se encuentren, es decir, los elementos serán seleccionados por una regla de estilo CSS en función de otros sucesos que hayan ocurrido, por ejemplo haber visitado un vínculo, o que el cursor del ratón se sitúe sobre un elemento, o en función del idioma especificado con el atributo común *lang*, en el elemento HTML.

El elemento `<a>` de HTML, utilizado para crear hiperenlaces, tiene definidas las siguientes pseudo-clases:

```
a:visited {color:blue;}
a:hover {color:blue;}
a:link {color:blue;}
a:focus {color:blue;}
a:active {color:blue;}
```

Vínculo visitado
Vínculo donde está el ratón
Vínculo no visitado
Vínculo seleccionado por teclado
Vínculo que recibe clic

La pseudo-clase para dar estilo a párrafos según el idioma sería así:

```
p:lang(en) {color:blue;}
```

Y la pseudo-clase para asignar un estilo al primer hijo de un elemento:

```
p:first-child {color:blue;}
```

Selectores CSS3

Una de las diferencias en el uso de los selectores CSS3 es utilizar el símbolo (`::`), doble dos puntos, en lugar de los dos puntos (`:`) que se usaba para los pseudo-elementos en CSS 2.1. Algunos ejemplos son:

```
p::first-letter
{color:blue}
```

Aplica la propiedad a la primera letra de cada párrafo.

```
p::first-line {color:blue}
```

Aplica la propiedad a la primera línea de cada párrafo.

Pseudo-elementos

Los pseudo-elementos son una parte de algún elemento HTML, que, aunque no tienen identidad propia, pueden ser identificados por las reglas de estilos y ser formateados de manera distinta a los elementos a los que pertenecen; algunos son: `::first-letter` y `::first-line`.

2.5 > Criterios de selección, selectores

Con los criterios de selección podemos indicar cuál es el selector de una regla de estilo CSS y así expresar a qué elementos de HTML se le aplicará el valor sobre la propiedad indicada en dicha regla CSS. Los diferentes criterios establecen, por lo tanto, distintos tipos de selectores:

Selector universal

Si se utiliza el asterisco (*) como selector, las propiedades especificadas afectarán a todos los elementos de la página; el asterisco es el selector universal. Para declarar el color azul a todos los elementos sería:

```
* {color:blue}
```

Selector de tipo o nombre de elemento

Permite aplicar un determinado estilo a todos los elementos del tipo seleccionado, con independencia de posibles clases, identificadores, etc. Para poder afinar más la selección se deben utilizar otros criterios de selección. Un ejemplo que declara el color azul a todos los párrafos sería:

```
p {color:blue}
```

Selector por combinación de elementos

Este criterio permite seleccionar en función de los elementos ascendientes/descendientes, padres/hijos o hermanos del elemento al cual se quiere dar un estilo. Estos elementos se pueden definir como:

- **Ascendientes/Descendientes:** ascendiente es el elemento que contiene al que se desea formatear, el descendiente. Puede haber más de un elemento entre uno y otro. Si B es un descendiente arbitrario de algún elemento antepasado A, entonces A se considera ascendiente de B. Para seleccionar elementos en función del ascendiente, se usan dos selectores separados por un espacio (), siendo el primero de ellos el ascendiente y el segundo el descendiente. Por ejemplo, si se quiere formatear de color rojo los elementos *p* que estén dentro de un elemento *div*, se debe escribir la siguiente regla:

```
div p {color : #00FF00}
```

- **Padres/Hijos:** padre es el elemento que contiene directamente al que se desea formatear, al que llamamos hijo. Para seleccionar elementos en función del padre, se usan dos selectores separados por el símbolo “mayor que” (>), siendo el primero el padre y el segundo el hijo. Por ejemplo, con la siguiente declaración solo se verían afectados los elementos *p* que estuvieran directamente dentro de un elemento *div*, es decir, los elementos *p* que son hijos de *div*, los cuales serán seleccionados para ponerles el color verde; sería así:

```
div > p {color : #00FF00;}
```

– **Hermanos:** hay dos combinaciones de hermanos, la combinación de hermanos adyacentes y la combinación de hermanos en general.

- **Hermanos adyacentes**, son elementos que se sitúan de forma que uno precede inmediatamente al otro, perteneciendo ambos al mismo elemento padre.

Para seleccionar elementos en función de hermanos adyacentes, se usan dos selectores separados por el símbolo suma (+), el primero de ellos representa el hermano precedente y el segundo el hermano al cual se quiere dar el formato. Por ejemplo, para seleccionar y poner en color verde el contenido de un elemento *pre*, que es hermano adyacente de un párrafo *p*:

```
p + pre {color : #00FF00;}
```

- **Hermanos en general**, son elementos que comparten el mismo elemento padre. Para indicar un selector en función de hermanos en general, se usan dos selectores separados por el símbolo de tilde (~), siendo el primero el elemento que precede (no necesariamente inmediatamente) al segundo selector, que es al que se quiere dar formato. Por ejemplo, para seleccionar un elemento de tipo *pre*, que es hermano de un elemento *h1*, sin serlo de manera adyacente, y ponerlo con color verde:

```
h1 ~ pre {color : #00FF00;}
```

Selector por clase o identificador de elemento

Forma de selección muy utilizada y de gran precisión, que se utiliza para dar formato y estilo solo a aquellos elementos que se hayan etiquetado con una clase (atributo *class*) o con un identificador (atributo *id*) concreto.

Para ello se escribirá el nombre del elemento seguido de un separador (el punto para las clases y la almohadilla para los identificadores), y detrás el nombre de la clase o del identificador correspondiente.

Si al usar un selector por clase o *id*, se omite el nombre del elemento, entonces se verán afectados todos los elementos que pertenezcan a dicha clase o todos los elementos con ese identificador. Por ejemplo, para alinear a la izquierda todos los elementos de la clase *Clase_Primer*a y a los que tienen como identificador *Nombre_Identificador*:

```
.Clase_Primer {text-align:left;}
#Nombre_Identificador {text-align:left;}
```

Para dar estilo a los párrafos de la clase “uno”, y a los que tengan el identificador “dos”, definidos en un documento HTML, se tienen que escribir las reglas CSS:

```
p.uno {text-align:#00FF00;}
p#dos {color : #00FF00;}
```

Selector por pseudo-clase del elemento

Las pseudos-clases clasifican elementos basándose en el estado en el cual se encuentren los mismos. Se utilizan a menudo con los hiperenlaces.

Elementos Hermanos

```
<h1>Título</h1>
<p>Párrafo</p>
<pre>Elemento seleccionado
para dar formato</pre>
```

Así, en función de si, por ejemplo, el enlace ha sido visitado o no, si el usuario está pasando por encima con el cursor, poder darles diferente estilo. Por ejemplo: `a:link {color:red;}`

Selector por atributo o valor del elemento

Se utiliza cuando se pretende aplicar estilo a elementos que tienen un determinado atributo o un valor de atributo dado. La forma de declararlo es *elemento[atributo]*. Por ejemplo, para seleccionar las celdas de una tabla (*td*) que tengan definido el atributo *fgcolor*, cualquiera que sea el valor del atributo se escribiría: `td[colspan]{color:#007777}`

Otros modos de uso del selector por atributo o valor del elemento son:

[att = valor] Representa un elemento con el atributo *att*, cuyo valor es exactamente “valor”.

[att ~ = valor] Representa un elemento con el atributo *att*, cuyo valor es una lista de palabras separadas por espacios en blanco, una de las cuales es “valor”.

[att | = valor] Representa un elemento con el atributo *att*, que tiene su valor exactamente como “valor” o que comienzan con “valor” e inmediatamente le sigue un guión (“-”).

2.6 > Unión y combinación de selectores

Con el objetivo de afinar en la definición de los selectores, a veces es conveniente realizar agrupaciones o combinaciones de selectores.

Unión de selectores

Cuando sea necesario aplicar las mismas reglas de estilo a varios elementos, se pueden reutilizar declaraciones, agrupando selectores, en lugar de repetir las mismas reglas de estilo con diferentes selectores. Para ello se separan los selectores con una coma (.). Por ejemplo, si se quiere formatear los elementos *ul* y *ol* con las mismas propiedades, se puede escribir la siguiente regla: `ul, ol {color:#00FF00;}`

Combinación de selectores

Los criterios de selección desarrollados en el punto anterior pueden combinarse para realizar una selección más concreta. Un ejemplo, que permite formatear a todos los elementos *li* que estén dentro de elementos *ul*, que pertenezcan a la clase *datos* y que tengan el atributo *id*, es esta combinación de selectores: `ul li.datos[id] {color:#00FF00;}`

2.7 > Elementos HTML <div> y

Los elementos de HTML `` y `<div>`, junto con los atributos *class* e *id*, permiten seleccionar o agrupar elementos de un documento HTML y así precisar a qué elementos se quieren asignar las reglas de estilo CSS. Ambos son elementos que no implican ningún cambio de estilo por sí mismos, y tampoco añaden contenido al HTML, sino que sirven para ayudar a organizar su estructura y servir como selectores en reglas CSS.

Agrupación con

El elemento **** permite definir una especie de gancho, una parte de un texto, para añadirle posteriormente un estilo diferente:

Casos prácticos

5

Creación de estilo usando la etiqueta

•• Dale estilo a un párrafo destacando determinadas palabras, haciendo que aparezcan enfatizadas con colores.

Solución •• Crea un nuevo documento, *span.html*, e incluye en el cuerpo un párrafo que contenga la cita de Benjamín Franklin donde considera las ventajas de madrugar, tal y como puedes leer en el siguiente párrafo del documento:

```
<p>El que pronto se acuesta y pronto se levanta, es hombre saludable, rico y sabio.</p>
```

Modifica el párrafo incluyendo cada una de las ventajas dentro de un elemento `` ``, así:

```
<p>El que pronto se acuesta y pronto se levanta, es hombre <span> saludable </span>, <span> rico </span> y <span> sabio </span>.</p>
```

A continuación, añade el atributo **class** a cada elemento ``, asignándole en los tres elementos el mismo valor; por ejemplo, el valor **"verde"** con los respectivos atributos, de la siguiente manera:

```
<p>El que pronto se acuesta y pronto se levanta, es hombre <span class = "verde"> saludable </span>, <span class = "verde"> rico </span> y <span class = "verde"> sabio </span>.</p>
```

Ahora crea un documento CSS, denominado *reglaSpan.css* que contenga esta sencilla regla de estilo CSS; con ella el selector permite seleccionar todos los elementos `` que pertenezcan a la clase **"verde"**.

```
span.verde {color:green;}
```

Añade una cabecera al documento HTML, que contenga una línea de código, para enlazar con la hoja de estilos externa recién creada:

```
<head><link rel="stylesheet" type="text/css" href="regalSpan.css"></head>
```

El efecto producido es este: **El que pronto se acuesta y pronto se levanta, es hombre saludable, rico y sabio.**

Consigue ahora que el color asignado sea diferente para cada una de estas tres ventajas. Para ello, modifica cada elemento ``, elimina los atributos de tipo **class**, y, posteriormente, añade un atributo **id** a cada uno de ellos, asignándoles diferente valor de identificador; deja el párrafo así:

```
<p>El que pronto se acuesta y pronto se levanta, es hombre <span id="EnVerde"> saludable </span>, <span id="EnRojo"> rico </span> y <span id="EnAzul"> sabio </span>.</p>
```

Modifica ahora el documento CSS denominado *reglaSpan.css* y añade estas reglas de estilo CSS, que tienen como selectores los nombres de los identificadores, para conseguir el efecto y asignar el color deseado:

```
span#EnVerde {color:green;} span#EnRojo {color:red;} span#EnAzul {color:blue;}
```

El que pronto se acuesta y pronto se levanta, es hombre saludable, rico y sabio.

Agrupación con <div>

El elemento **<div>** permite crear bloques, capas o cajas, es decir, una agrupación de elementos para darles a todos ellos el mismo estilo y con ello poder estructurar una página web con estas divisiones.

Los elementos **<div>** y **** sirven para realizar cosas muy sencillas, como, por ejemplo, dar estilo a los textos y colores. Pero ambos elementos tienen un gran potencial para realizar cosas mucho más avanzadas, sobre todo la etiqueta **<div>**, que se utiliza a menudo para crear efectos visuales en la información contenida, a modo de cajas que agrupan los elementos, para ser representadas en un lugar concreto que interese para facilitar la navegación del usuario de la página web.

Ejemplos

Diferenciar elementos por división (etiqueta <div>) con una hoja de estilo externa

Supongamos un fichero HTML que contenga estas dos listas de presidentes de los Estados Unidos, una para cada afiliación política mayoritaria, los demócratas y los republicanos, agrupadas ambas con **<div>**.

```
<div id="democratas">
<ul>
<li>Franklin D. Roosevelt</li>
<li>Harry S. Truman</li>
<li>John F. Kennedy</li>
<li>Lyndon B. Johnson</li>
<li>Jimmy Carter</li>
<li>Bill Clinton</li>
</ul>
</div>

<div id="republicanos">
<ul>
<li>Dwight D. Eisenhower</li>
<li>Richard Nixon</li>
<li>Gerald Ford</li>
<li>Ronald Reagan</li>
<li>George Bush</li>
<li>George W. Bush</li>
</ul>
</div>
```

El ejemplo mostrará cada lista con un color diferente, **azul** para la lista de presidentes demócratas y **rojo** para representar la lista de los presidentes republicanos.

Para ello se recomienda definir el estilo con una hoja de estilos externa y utilizar las agrupaciones realizadas con **<div>** y sus respectivos identificadores, en este ejemplo *"democratas"* y *"republicanos"*. Estas son las dos reglas de estilo CSS necesarias en este ejemplo:

```
#democratas {background:blue;}
#republicanos {background:red;}
```

Así será la visualización en el navegador; se consigue resaltar, con fondo en color azul, a los presidentes demócratas y, en color rojo, a los presidentes republicanos.

- Franklin D. Roosevelt
- Harry S. Truman
- John F. Kennedy
- Lyndon B. Johnson
- Jimmy Carter
- Bill Clinton

- Dwight D. Eisenhower
- Richard Nixon
- Gerald Ford
- Ronald Reagan
- George Bush
- George W. Bush

2.8 > Colores, fondos, textos y fuentes

Para dar estilo a un documento HTML existen muchas propiedades CSS. Estas son algunas de las que permiten establecer colores y tipos de letra:

Propiedad	Utilidad	Valor	Declaración
color	Color de texto	Hexadecimal RGB RGBA HSL HSLA Predefinido inherit	color:#ff0000; color:rgb(255,0,0); color:Cyan;
background-color	Color de fondo	color transparent inherit	background-color:#00ff00; background-color:rgb(255,0,255) background-color:yellow;
background-image	Imagen de fondo	url('URL')	background-image: url('paper.gif');
text-align	Justificar texto	left right center justify inherit	text-align:center
text-decoration	Decorar texto	none underline overline blink line-through inherit	text-decoration:underline
text-indent	Indentar o sangrar texto (margen)	longitud % inherit	text-indent
font-family	Tipo de fuente	family-name generic-family	font-family: 'Arial', 'Times-New-Roman', sans-serif;
font-size	Tamaño de fuente	xx-small x-small small medium large x-large xx-large smaller larger unidad de medida	font-size:1cm; font-size:12px; font-size:80%;
font-weight	Grueso de los caracteres de texto	normal bold bolder lighter 100 200 300 400 500 600 700 800 900	font-weight:normal;
font-style	Estilo de fuente	normal italic oblique	font-style:normal; font-style:italic;
font-size-adjust	Ajusta el tamaño, con independencia del tipo de fuente de un texto	none número	font-size-adjust:0.5; font-size-adjust:none;
inherit: indica que el valor de la propiedad será el heredado del elemento padre.			
unidad de medida: pixels (px), centímetros (cm), milímetros (mm), pulgadas (in), picas (pc) o porcentaje (%) sobre el tamaño del padre.			

Actividades propuestas

5•• Escribe la etiqueta `<h1>SALUDOS CSS</h1>` y aplica localmente el estilo necesario para que salga en rojo y con fondo amarillo. Haz lo mismo con una hoja de estilos interna y añade una imagen de fondo al documento.

6•• Crea un documento HTML que contenga un párrafo de texto, y enlázalo a una nueva hoja de estilos que incluya en ella los ejemplos contenidos en la columna DECLARACIÓN de la tabla superior.

2.9 > Bloques, capas o cajas

Sobre los elementos HTML se pueden definir los bloques o capas, también denominadas cajas. Son muy utilizadas con la etiqueta `<div>`, y permiten representar con el mismo estilo a los elementos agrupados, según las propiedades específicas relacionadas con las cajas. Algunas de estas propiedades se recogen en la siguiente tabla:

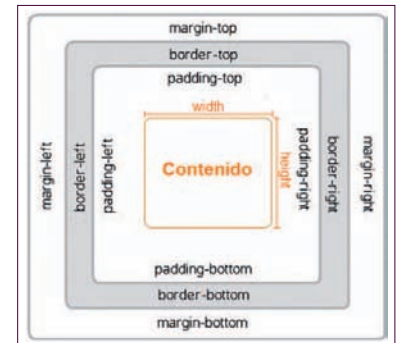
Propiedad	Utilidad	Valor	Declaración
border	Borde, ancho, estilo y color de la caja	border-width border-style color	border:5px solid red;
border-top	Borde superior	border-width border-style color	border-top:thick double #ff0000;
border-right	Borde derecho	border-width border-style color	border-right:medium solid #00ff00;
border-width	Propiedades del tamaño del ancho de la caja. Según el número de valores utilizados se aplica a diferentes zonas del margen	top right bottom left top (right and left) bottom (top and bottom) (right and left) (top right bottom left)	border-width:thin 10px thick 10px; border-width:thin medium thick; border-width:thin medium; border-width:thin;
border-style	Propiedades del estilo de la caja. Se aplica según el número de valores utilizados	top right bottom left	border-style:dotted solid double dashed;
border-color	Propiedades del color de la caja. Se aplica según el número de valores utilizados	top right bottom left	border-color:red green blue pink;
margin	Propiedades del tamaño del margen de la caja. Según el número de valores utilizados se aplica a diferentes zonas del margen	top right bottom left top (right and left) bottom (top and bottom) (right and left) (top right bottom left)	margin:10px 5px 15px 20px; margin:10px 5px 15px; margin:10px 5px; margin:10px;
margin-bottom	Margen inferior	unidad de medida inherit	margin-bottom:1cm;
margin-left	Margen izquierdo	unidad de medida inherit	margin-left:2cm;
padding	Propiedades del tamaño del relleno de la caja. Según el número de valores utilizados se aplica a diferentes zonas del margen	top right bottom left top (right-left) bottom (top-bottom) (right-left) (top-right-bottom-left)	padding:10px 5px 15px 20px; padding:10px 5px 15px; padding:10px 5px; padding:10px;

Actividades propuestas

7· Crea cajas con elementos `<div>` que agrupen elementos HTML básicos, e incluye en una hoja de estilos las declaraciones de ejemplo de la columna DECLARACIÓN de la tabla anterior, y pruébalas sobre las cajas creadas.

2.10 > Tamaño y posicionamiento

Existen propiedades CSS que permiten establecer la posición de los diferentes elementos de contenido de un fichero HTML. El método de posicionamiento de elementos se establece con la propiedad *position*, siendo su valor por defecto *static*, que muestra los elementos en el mismo orden en el que aparecen en el documento HTML, sin afectarle otras propiedades de posicionamiento. Con el valor *absolute*, los elementos se sitúan en la posición indicada, en las propiedades *top*, *bottom*, *left* y *right*, con respecto a su primer elemento antecesor que tenga un posicionamiento diferente a *static*. Con el valor *fixed*, los elementos se sitúan en una posición indicada con respecto a la ventana del navegador, unos encima de otros. Con el valor *relative*, los elementos se sitúan en una posición (admite valores negativos) relativa a su posición normal por defecto (*static*). Se resumen en la tabla la propiedad *position* y otras propiedades utilizadas:



Propiedades CSS para cajas con div.

Propiedad	Utilidad	Valor	Declaración
<i>position</i>	Método de posicionamiento de los elementos HTML en la ventana del navegador	<i>static</i> <i>absotute</i> <i>fixed</i> <i>relative</i> <i>inherit</i>	<i>position: absolute;</i>
<i>top</i>	Distancia de la parte superior de un elemento respecto a su posición normal. Si <i>position</i> es " <i>static</i> ", esta propiedad no tiene efecto	unidad de medida <i>inherit</i>	<i>top: 5px;</i>
<i>right</i>	Distancia de la parte derecha de un elemento respecto a su posición normal. Si <i>position</i> es " <i>static</i> ", esta propiedad no tiene efecto	unidad de medida <i>inherit</i>	<i>right: 5px;</i>
<i>bottom</i>	Distancia de la parte inferior de un elemento respecto a su posición normal. Si <i>position</i> es " <i>static</i> ", esta propiedad no tiene efecto	unidad de medida <i>inherit</i>	<i>botton: 5px;</i>
<i>left</i>	Distancia de la parte izquierda de un elemento respecto a su posición normal. Si <i>position</i> es " <i>static</i> ", esta propiedad no tiene efecto	unidad de medida <i>inherit</i>	<i>left: 20px;</i>
<i>float</i>	Modifica el posicionamiento normal desplazando el elemento hacia el lado especificado. Permite colocar texto alrededor de imágenes. Si <i>position</i> es " <i>absolute</i> ", esta propiedad no tiene efecto	<i>left</i> <i>right</i> <i>none</i> <i>inherit</i>	<i>float: left;</i>
<i>clear</i>	Indica los lados de un elemento donde otros elementos flotantes no están permitidos	<i>right</i> <i>left</i> <i>both</i> <i>none</i> <i>inherit</i>	<i>clear: both;</i>
<i>display</i>	Define cómo mostrar un determinado elemento, el tipo de caja	<i>inline</i> <i>block</i> <i>list-item</i> <i>none</i>	<i>display: block;</i>
<i>z-index</i>	Nivel de profundidad en la colocación superpuesta de elementos. Cuanto mayor sea el valor indicado más arriba aparecerá el elemento	<i>valor</i>	<i>z-index: 2;</i>
<i>visibility</i>	Especifica si el elemento es visible o no. El valor predeterminado es visible.	<i>visible</i> <i>hidden</i> <i>collapse</i>	

Casos prácticos

6

Hacer flotar una imagen respecto a los párrafos con *float* y *clear*

•• Poner una imagen flotando a la izquierda de un párrafo, y hacer que otro párrafo no lo permita.

Solución •• Crea el siguiente documento HTML que expresa dos reglas de estilo para practicar el uso de las propiedades *float* y *clear*. Observa que al elemento imagen se le aplica la propiedad *float*, para que flote a la izquierda respecto al siguiente elemento que le sigue, que en este caso es el primer párrafo; este párrafo permite que los elementos floten sobre cualquiera de sus lados, ya que no tiene definida la propiedad *clear*. Fíjate que al segundo párrafo, que pertenece a la clase "clear", se le asigna esa misma propiedad *clear* y el valor *both*, para que no permita elementos flotantes a ninguno de sus lados, ni a su derecha ni a su izquierda.

```
<html>
<head>
  <style>
    img {float:left;}
    p.clear {clear:both;}
  </style>
</head>
<body>
  
  <p>Este es un párrafo que permite que el resto de elementos floten sobre él.
  Este es un párrafo que permite que el resto de elementos floten sobre él.</p>

  <p class="clear">Este párrafo NO permite que el resto de elementos floten sobre
  él. Este párrafo NO permite que el resto de elementos floten sobre él.</p>
</body>
</html>
```

Guarda este fichero con el nombre float.html y ábrelo desde el navegador. Observarás el siguiente resultado:



Este es un párrafo que permite que el resto de elementos floten sobre él. Este es un párrafo que permite que el resto de elementos floten sobre él.

Este párrafo NO permite que el resto de elementos floten sobre él. Este párrafo NO permite que el resto de elementos floten sobre él.

Modifica las propiedades *float* y *clear*, asignándoles los valores *right* y *none*, respectivamente, y comprueba los resultados obtenidos y cómo se posiciona la imagen respecto al segundo párrafo.

Actividades propuestas

8•• Modifica la hoja de estilos asociada de la actividad número 7, incluyendo nuevas reglas de estilo para cambiar los tamaños de las cajas creadas en esa actividad.

9•• Añade reglas de estilo que modifiquen el posicionamiento de las cajas del ejercicio anterior.

10•• Coloca superpuestas varias cajas de las creadas en los ejercicios anteriores, utilizando z-index en cada una de ellas.

3 >> Lenguajes de *script* de navegador

Para dotar de una mayor funcionalidad a las páginas web es conveniente complementarlas con algún lenguaje de *script* de navegador. Estos lenguajes, como, por ejemplo, **JavaScript**, permiten modificar de manera dinámica el contenido del documento, validar los datos de los formularios, obtener efectos gráficos, etc., todo ello desde el entorno del cliente de la web. Pero estos lenguajes tienen sus limitaciones, ya que por ejemplo no pueden realizar cosas como manejar bases de datos; para ello, sería necesario utilizar un lenguaje de *script* de servidor, por ejemplo **PHP**.

3.1 > Introducción a JavaScript

Es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos, sino que es el propio navegador quien lo interpreta al cargar la página o al interactuar el usuario con la misma.

Un método para incluir código JavaScript es utilizar las etiquetas **<script></script>** en cualquier parte del documento HTML, incluyendo dentro de ellas las instrucciones del lenguaje, por ejemplo, así:

```
<script>
    alert("HOLA MUNDO <br> JAVASCRIPT");
</script>
```

HOLA MUNDO
JAVASCRIPT

El principal inconveniente de este método es que si se quiere hacer una modificación en el bloque de código, es necesario modificar todas las páginas que incluyen ese mismo bloque de código JavaScript.

Otro método es almacenar las instrucciones del lenguaje en un **documento externo** de tipo JavaScript que los documentos HTML enlazan mediante la etiqueta **<script>**. Se pueden crear todos los archivos JavaScript que sean necesarios y cada documento HTML puede enlazar tantos archivos JavaScript como necesite, de la siguiente manera:

```
<html>
<head>
<title>Código JavaScript en documento externo</title>
<script type="text/javascript" src="codigo.js"></script>
</head>
<body>
    <p>HOLA MUNDO JAVASCRIPT.</p>
</body>
</html>
```

El documento *codigo.js* contendrá, por ejemplo, la siguiente instrucción, `alert("Un mensaje de prueba");` El resultado será mostrar un mensaje emergente con el texto indicado en la función `alert()`, y tras pulsar en el botón de *Aceptar* se mostrará el texto del párrafo (`<p>`):



HOLA MUNDO JAVASCRIPT.

Lenguaje JavaScript

El lenguaje de *script* **JavaScript** fue muy utilizado inicialmente, pero disminuyó su popularidad cuando apareció **Flash**, que permitía realizar acciones imposibles de llevar a cabo con JavaScript.

Posteriormente, con la aparición de las aplicaciones **AJAX** programadas con JavaScript, ha vuelto su popularidad dentro de los lenguajes de programación web.

JavaScript es el lenguaje de *script* por defecto en **HTML5** y en todos los navegadores modernos. Por lo tanto, ya no es necesario utilizar el atributo `type` con el valor "text/javascript", dentro de la etiqueta `<script>`.

Comentarios en JavaScript

Para introducir **comentarios** en JavaScript hay que colocar dos barras inclinadas (`//`) para comentar una línea.

```
//Esto es un comentario
```

También se pueden comentar varias líneas de instrucciones si se incluyen entre los símbolos barra asterisco (`/*`) y los mismos símbolos, pero en orden inverso (`*/`).

```
<script>
/* Este es un ejemplo de varias
líneas de código comentadas */
</script>
```

Los comentarios se deben utilizar para añadir información útil sobre el código fuente del programa; el navegador no los visualiza.

Palabras reservadas

Son las palabras (en inglés) que se utilizan para construir las instrucciones de un lenguaje de programación y que, por tanto, no pueden ser utilizadas libremente. Algunas de las palabras reservadas por JavaScript son: *break, case, catch, continue, default, delete, do, else, finally, for, function, if, in, instanceof, new, return, switch, this, throw, try, typeof, var, void, while, with*.

Ubicación de los <script>

Los *scripts* pueden estar dentro de cualquier elemento de HTML: en la cabecera (<head>), en el cuerpo (<body>) o en ambos elementos.

Es recomendable declarar e implementar las funciones ubicándolas en la sección <head>, o en la parte inferior de la página externa de JavaScript, haciendo más fácil su mantenimiento y modificación.

Creación de funciones

```
function nombre_función()
{ //Bloque instrucciones
  [return valor]}
```

Llamada a una función

```
variable=nombre_función()
```

3.2 > Sintaxis y características de JavaScript

La sintaxis de cualquier lenguaje de programación define el conjunto de normas para poder escribir instrucciones correctas en ese lenguaje.

El código de un programa JavaScript se compone de un número ilimitado de sentencias o instrucciones, que le indican al navegador lo que debe hacer, cumpliendo con la sintaxis del lenguaje; la sintaxis de JavaScript es muy flexible y sus **reglas básicas** son:

- Se ignoran los espacios en blanco superfluos, lo que permite ordenar el código añadiendo espacios, tabulaciones o nuevas líneas.
- Se distinguen las mayúsculas y minúsculas.
- No es necesario declarar el tipo de dato que almacenará una variable; se pueden almacenar diferentes tipos de datos con la misma variable.
- Las sentencias de JavaScript pueden agruparse en bloques. Estos bloques de instrucciones se delimitan entre llaves ({}). Un buen ejemplo del uso de bloques, son las funciones de JavaScript.
- El punto y coma (;) ayuda a separar sentencias JavaScript. Es conveniente agregar un punto y coma al final de cada sentencia.

Cada una de las instrucciones es interpretada en el orden en que están escritas; por ejemplo, la siguiente declaración JavaScript indica al navegador que escriba un texto de cabecera y después un nuevo párrafo.

```
<script>
    document.write("<h1>Esto es una cabecera
    H1</h1>");
    document.write("<p>Esto es un párrafo</p>");
</script>
```

Las sentencias de JavaScript se ejecutan mientras se carga la página web, pero en muchas ocasiones no interesa ejecutar el código en ese proceso de carga, sino en otro momento, dando cierto dinamismo a la página web.

Para ello hay que ubicar las sentencias dentro de una función (*function*), que posteriormente será llamada para ejecutar sus instrucciones, cuando se produzca un determinado evento, por ejemplo, cuando el usuario haga clic con el ratón (*onclick*) en un determinado elemento HTML.

Ejemplos

Creación de una página dinámica básica, basada en un evento y en el uso de funciones

Se sobrescribe el texto del párrafo identificado por "demo", por la fecha actual.

```
<!DOCTYPE html>
<html>
<head>
<script>
function muestraFecha(){
document.getElementById("demo").innerHTML
= "Hoy es: " + Date();}
</script>
</head>

<body>
<h1>Mi primer JavaScript</h1>
<p id="demo"> Mi primera función JS.</p>
<button type = "button"
onclick="muestraFecha()">
Mostrar Fecha</button>
</body>
</html>
```

3.3 > Variables, operadores y estructuras básicas de control

Para desarrollar utilidades con JavaScript es necesario conocer los elementos básicos, necesarios para poder entender posteriormente la programación avanzada, con los que se construyen grandes aplicaciones.

Variables

Una variable es un elemento que se emplea para almacenar un valor o poder hacer referencia a otra variable. Según su **ámbito**, lugar donde se puede utilizar una variable, estas pueden ser variables **locales** o **globales**.

La declaración de **variables locales** JavaScript se hace mediante la palabra reservada *var*, seguida del nombre de la variable a crear. Si son varias las variables a declarar, se deben separar por comas (,):

```
var numero1, numero2; //Declara dos variables locales
```

Una vez declarada una variable hay que inicializarla, asignarle un valor inicial con el operador de asignación, el símbolo igual (=). El proceso de inicialización se puede hacer en el momento de la misma declaración:

```
var numero1 = 3; var numero2 = 1;
var resultado = numero1 + numero2;
```

O bien se puede realizar una inicialización posterior a la declaración:

```
var numero1, numero2, resultado;
numero1=3; numero2=1; resultado=numero1 + numero2;
```

La palabra reservada *var* solamente se debe indicar al definir por primera vez una variable; luego solamente es necesario indicar su nombre.

JavaScript es un lenguaje muy flexible en el tratamiento de variables, ya que permite asignar un valor a una variable que no haya sido declarada anteriormente con la palabra reservada *var*; así se consigue declarar e inicializar las denominadas **variables globales**, aquellas que podrán ser utilizadas en cualquiera de las funciones creadas en el *script*, por ejemplo:

```
numero3=6; //Declara e inicializa una variable global
numero4=2; //Declara e inicializa numero4 como global
resultado = numero3 + numero4; //Ídem para resultado
```

Se recomienda declarar las variables locales de manera general, y solo declarar variables globales cuando sea imprescindible y necesario. Una variable será de un tipo u otro según el valor asignado a dicha variable; pueden ser números, cadenas de caracteres, valores lógicos, etc.

Para asignar una cadena de caracteres a una variable, se debe encerrar dicha cadena entre comillas dobles o simples, para delimitar el comienzo y el final de dicha cadena de caracteres:

```
var bienvenida = "Bienvenido a nuestro sitio web";
var letraSeleccionada = 'c';
```

Las variables lógicas son las que almacenan un valor *booleano*, es decir, verdadero (*true*) o falso (*false*), ejemplo: `var ivaIncluido=true;`

Variables

Siguen una lógica similar a las variables utilizadas en el ámbito de las matemáticas; si en las matemáticas no existieran las variables no se podrían definir las ecuaciones y fórmulas; de igual forma, en programación no se podrían hacer programas realmente útiles sin utilizar variables.

Identificadores

El nombre de una variable también se conoce como identificador y debe cumplir las siguientes normas:

- Solo puede estar formado por letras, números y los símbolos dólar (\$) y guion bajo (_).
- El primer carácter no puede ser un número.

Secuencias de escape

JavaScript define un mecanismo para incluir de forma sencilla caracteres especiales dentro de una cadena de texto. Consiste en sustituir el carácter especial por una combinación simple de caracteres, una secuencia de escape, como las de esta tabla:

Para incluir...	Secuencia...
Una nueva línea	\n
Un tabulador	\t
Una comilla simple	\'
Una comilla doble	\"
Una barra inclinada	\\

Operadores Aritméticos

Suma	+
Resta	-
Multipliación	*
División	/
División entera	\
Resto de división (módulo)	%
Cambio de signo	-
Incremento en una unidad	++
Decremento en una unidad	--

Operadores Relacionales

Mayor que	>
Menor que	<
Igual a	==
Distinto a	!=
Mayor o igual a	>=
Menor o igual a	<=

Operadores Lógicos

Y lógico (and)	&&
O lógico (or)	
Negación lógica (not)	!

Arrays

También se les llama vectores o matrices. Un *array* es un conjunto de variables, agrupadas bajo un mismo identificador, que pueden ser todas del mismo tipo o cada una de un tipo diferente.

Los *arrays* se pueden declarar e inicializar de tres formas:

```
var nom_array=new Array();
nom_array[0]="valor1";
nom_array[1]="valor2";
nom_array[N-1]="valorN";

var nom_array =new Array
("valor1",...,"valorN");

var nom_array = [valor1,
valor2,...,valorN];
```

Para recorrer el *array* se puede utilizar una estructura For-In, así:

```
for(i in dias) {
    alert(dias[i]);}
```

Operadores

Los operadores permiten asignar el valor a las variables, realizar operaciones matemáticas con sus valores y comparar diferentes variables. Los operadores fundamentales en JavaScript, además del operador de asignación (=), son los operadores aritméticos, los operadores relacionales y los operadores lógicos.

Los operadores aritméticos pueden ser utilizados como operadores de asignación; por ejemplo, la instrucción `x += y` asignará a la variable `x` el valor resultado de la operación de `x + y`, lo que sería igual que indicar `x = x + y`.

Estructuras de control del flujo básicas

Son instrucciones que permiten alterar el orden de ejecución de las instrucciones que forman un *script*, realizar cálculos complejos, tomar decisiones lógicas, en función del resultado de evaluar una condición, o repetir un mismo bloque de instrucciones de forma eficiente. Así los programas dejan de ser una sucesión lineal de instrucciones y se convierten en programas inteligentes que toman decisiones en función del valor de las variables. Algunas de ellas son:

- **Condicional simple:** si la *condición* se cumple (su valor es *true*), se ejecutarán las instrucciones que se encuentran dentro del bloque (`{...}`), mientras que si no se cumple (su valor es *false*), no se ejecutarán y el programa continuará ejecutando el resto de instrucciones del *script* que haya detrás del bloque de la estructura condicional.

```
if(condición) { //Bloque de instrucciones...}
```

- **Condicional doble:** si se cumple la *condición*, se ejecutarán las instrucciones que se encuentran dentro del *Bloque1* de instrucciones, pero si no se cumple, las del bloque.

```
if(condición) { //Bloque1 de instrucciones...}
else { //Bloque2 de instrucciones...}
```

- **Repetitiva Mientras-Hacer:** si se cumple la condición, se ejecutarán repetidamente las instrucciones del bloque hasta que deje de cumplirse.

```
while(condición) { //Bloque de instrucciones...}
```

- **Repetitiva Hacer-Mientras:** se ejecutarán las instrucciones del bloque al menos una vez y se repetirán mientras se cumpla la condición.

```
do { //Bloque de instrucciones...} while(condición)
```

- **Repetitiva For:** primero se ejecuta la sentencia de *inicialización*; luego, si se cumple la condición, se ejecutarán las instrucciones del bloque de instrucciones y se ejecuta la sentencia de *actualización*; después, mientras la *condición* se cumpla, se repetirá la ejecución de las instrucciones del bloque y de la sentencia de actualización.

```
for(inicializacion; condición; actualización)
    { //Bloque de instrucciones...}
```

- **Repetitiva For-In:** permite recorrer los valores de un *array*.

```
for(índice in array) { //Bloque de instrucciones...}
```

3.4 > Ventanas emergentes

JavaScript ofrece tres formas de mostrar ventanas emergentes:

- **Ventana de alerta:** muestra una *cadena* de caracteres y un botón de “Aceptar” que cierra la ventana de alerta. Función ***alert(cadena)***.
- **Ventana de confirmación:** muestra una *cadena* de caracteres y dos botones, “Aceptar” y “Cancelar”, que cierran la ventana y provocan que la función devuelva el valor, *true* o *false*, respectivamente, según el botón que se pulse. Función ***confirm(cadena)***.
- **Ventana de entrada:** muestra una *cadena* y una caja de texto, inicializada a un valor *inicial*, que el usuario puede modificar. La función devolverá una cadena de caracteres con el contenido introducido por el usuario. Función ***prompt(cadena, inicial)***.

Otras funciones JavaScript

toUpperCase(cadena): convierte la cadena de entrada a mayúsculas.

toLowerCase(cadena): convierte la cadena de entrada a minúsculas.

parseInt(cadena): devuelve la cadena convertida en un entero, y el valor ***NaN*** (“Not a Number”) si la cadena no es un número.

Infinity: representa el valor infinito, positivo o negativo.

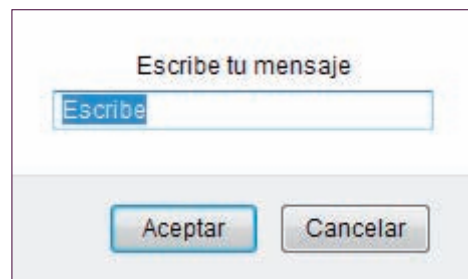
Ejemplos

Creación de ventanas emergentes

Dos ejemplos para mostrar mensajes con ventanas emergentes.

```
<!DOCTYPE html>
<html>
<body>
<script>
mensaje = "Esto es un mensaje
de alerta";
alert("Ventana de alerta
- " + mensaje);
</script>
</body>
</html>

<!DOCTYPE html>
<html><body>
<script>
mensaje = "Esto es un mensaje de alerta";
resp=confirm("¿Quieres ver el mensaje?");
if (resp==true) {alert(mensaje);}
else {
mensaje = prompt ("Escribe tu mensaje",
"Escribe");
document.write("Tu mensaje es: "+mensaje)}
</script></body></html>
```



Actividades propuestas

11 Realiza un *script* que adivine cuál es el mayor de dos valores numéricos introducidos por un usuario.

Actividades finales

.: CONSOLIDACIÓN .:

- 1• Define las siglas HTML, SGML, XHTML y W3C.
- 2• ¿Cuáles son las etiquetas de HTML que forman la estructura básica de una página web?
- 3• Entra en la página web del W3C, <http://www.w3c.es/>, visualiza el código fuente de la página web y localiza las etiquetas fundamentales, identificando los elementos HTML que representan.
- 4• Indica la diferencia entre una hoja de estilos interna y una hoja de estilos externa. ¿Cómo es la estructura de una regla de estilo CSS? ¿Cuáles son y para qué sirven los criterios de selección?
- 5• Realiza una lista de las características principales para usar correctamente el lenguaje JavaScript.

.: APLICACIÓN .:

- 1• Crea una página web llamada *refresca.html*, que actualice la página web transcurridos unos segundos. Define el fondo de color verde claro, el texto a visualizar en azul, los enlaces en verde oscuro y los enlaces ya visitados en marrón. Incluir un texto representativo y enlaces de <http://www.google.es> y <http://www.facebook.es> para comprobar los colores. Modifica la página utilizando tus colores preferidos.
- 2• Genera una página web que visualice un párrafo de texto que ocupe varias líneas (puedes copiarlo de alguna página web de Internet). Manipula la información del párrafo incluyendo espacios y saltos de línea; vuelve a guardar la página y comprobarás que al visualizar la página no se aprecian esos espacios o saltos incluidos. Luego, intenta que algunos de esos cambios sí tengan efecto en la visualización. Justifica el primer párrafo a la izquierda, añade un segundo párrafo con el texto centrado y separa su visualización por varias líneas y añade alguna línea horizontal que ocupe el ancho de la página entre ellos.
- 3• Realiza una página web que muestre tu horario personal de clases, estructurado en filas (horas del día) y columnas (días de la semana), realízalo a través de tablas HTML y dale color a tu gusto.
- 4• Diseña una página web que contenga un formulario que permita incluir los siguientes datos personales: Nombre, Apellidos, Dirección, Código postal, Población, Provincia, Teléfono, Correo electrónico y Fecha de nacimiento (día, mes y año). Añade a la página web botones radio, para que el usuario indique su sexo, y casillas de selección, para que pueda expresar sus gustos o aficiones.
- 5• A partir del código del fichero *selectores.html*, que encontrarás en el CD *Recursos*, añade los selectores CSS que faltan para aplicar los estilos tal y como marcan los comentarios.
- 6• Diseña un conjunto de páginas web con información sobre tu afición favorita. Crea un mismo estilo para que se produzca el efecto de no cambiar aparentemente de página al navegar por ellas. Estarán estructuradas de la siguiente manera. La parte superior ocupará el ancho de la página, tendrá una altura de 55 px, y en ella se insertará una imagen de fondo y un rótulo general a modo de cabecera general del conjunto de páginas web. Debajo, en la parte izquierda, ubica un menú de navegación que ocupe 200 px de ancho. Se destinará a alojar una lista de al menos cinco epígrafes de texto o iconos que estén enlazados (Inicio, Historia, Noticias, Enlaces y Formulario de registro de visitas), y que permitan navegar a través del resto de páginas que informan sobre tu afición favorita. A la derecha del menú de navegación, ocupando el resto del ancho de la pantalla, se mostrará el contenido de cada una de las páginas seleccionadas en el menú de navegación. Por defecto, la página inicial del sitio web se denominará *index.html*. Utiliza todos los elementos HTML que puedas de los explicados en esta unidad didáctica, así como las reglas CSS necesarias para darle tu estilo original deseado.
- 7• Crea una página web que calcule y muestre las tablas de multiplicar del 1 al 10.

Caso final

7

Controlar posicionamiento de bloques o cajas

• Realiza una página web como la de la imagen, manteniendo el control sobre el posicionamiento de cada uno de los cinco bloques coloreados.

Cuando se pase el cursor del ratón por cualquiera de los bloques, se debe producir un degradado en el color de la caja.

Solución •

a) Crear el siguiente documento HTML:

```
<html>
<head>
<title>Posicionamiento de bloques o cajas</title>
</head>
<body>
<div id="contenido">
<div id="rojo">Rojo</div>
<div id="azul">Azul</div>
<div id="negro">Negro</div>
<div id="amarillo">Amarillo</div>
<div id="verde">Verde</div>
</div>
</body>
</html>
```

b) Crear este documento CSS:

```
body{ background-color: #EEE}
```

```
#contenido{
  position: absolute;
  width: 500px;
  height: 500px;
  top: 50%;
  left: 50%;
  margin-top: -250px;
  margin-left: -250px;
  background-color: orange;}
```

```
#rojo{
  width: 150px;
  height: 350px;
  background-color: red;
  float: left;}
```

```
#rojo: hover{background-color: #f55;}
```

```
#azul{
  width: 350px;
  height: 150px;
  background-color: blue;
  float: left;}
```

```
#azul: hover{background-color: #55f;
}
```

```
#negro{
  width: 200px;
  height: 200px;
  background-color: black;
  float: left;}
```

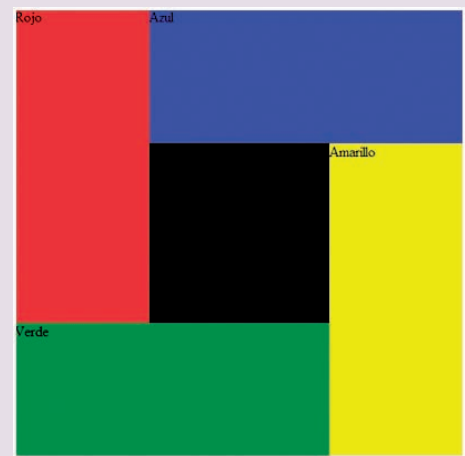
```
#negro: hover{background-color: #333;}
```

```
#amarillo{
  width: 150px;
  height: 350px;
  background-color: yellow;
  float: right;}
```

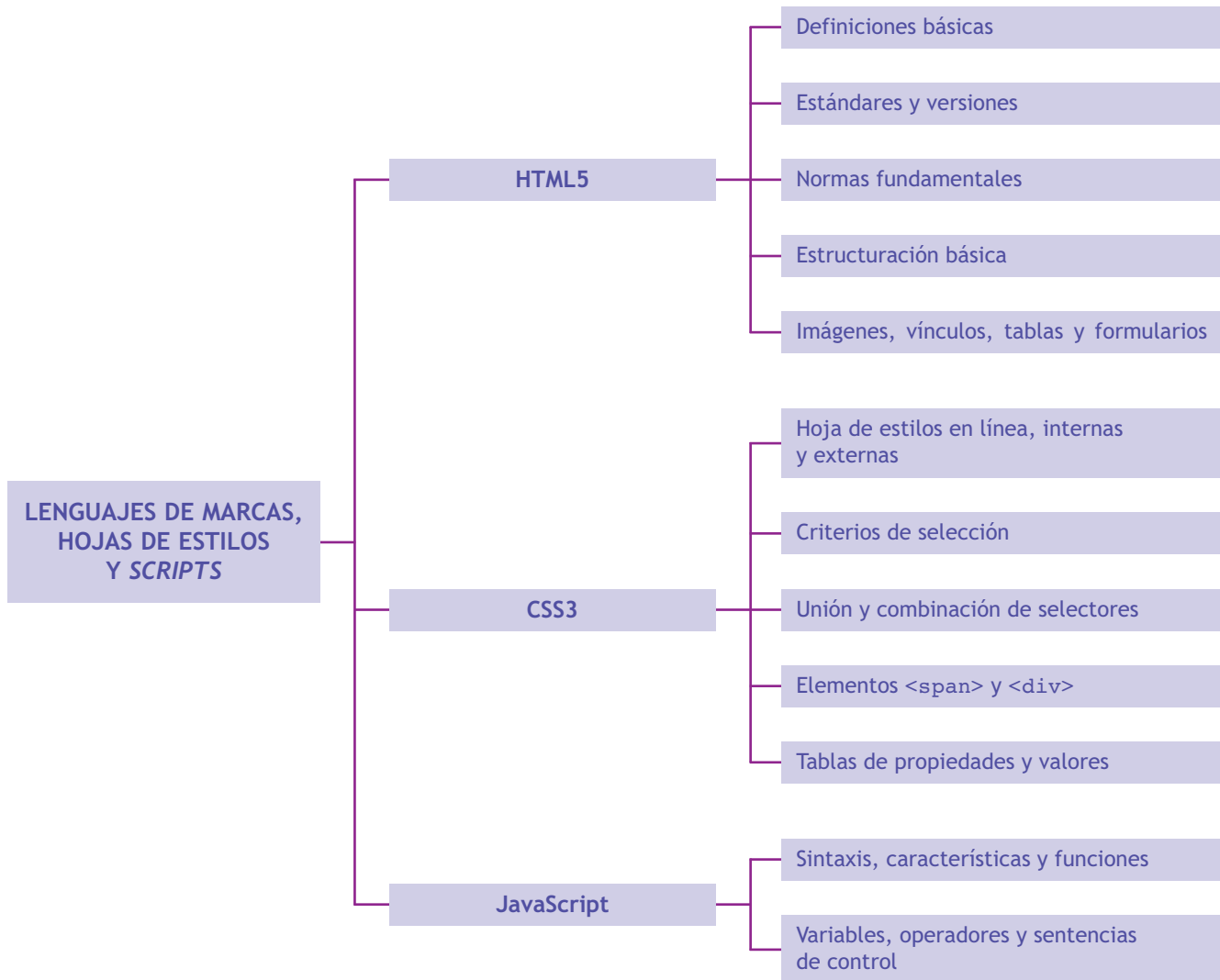
```
#amarillo: hover{background-color:
#FE8;}
```

```
#verde{
  width: 350px;
  height: 150px;
  background-color: green;
  float: left;}
```

```
#verde: hover{background-color: #5f5;}
```



Ideas clave



HTML5 en la Web



HTML5 es el futuro, así lo suponía y defendía Steve Jobs y terminó convenciendo a Adobe de que pasaría en un futuro no muy lejano. Queda claro que la transición a HTML5 es un proceso lento pero, al parecer, seguro, puesto que cientos de empresas de software lo apoyan y desde hace unos meses están ofreciendo alternativas a lenguajes como Flash, por ejemplo.

La mejor muestra de que HTML5 cambiará las reglas del juego es Microsoft Surface, una tecnología increíble que hasta ahora no cuenta con aplicaciones para aprovecharla, además de su gran limitación: su precio, pero eso es harina de otro costal. Gracias a HTML5, Microsoft Surface será capaz de contar con gestos multitáctiles en cualquier navegador que permita la aclaración por hardware. Es decir, que una aplicación web que lo permita podrá correr y soportar sin problemas en Internet Explorer como lo demuestran en este vídeo:

http://www.youtube.com/watch?v=aaRm4hiXbCg&feature=player_embedded#t=0s

Según el representante de **IntuiLab**, **Massive Multi-touch** permitirá gestos multitáctiles en navegadores de tabletas, teléfonos inteligentes y superficies planas como Microsoft Surface mientras cuente con un navegador compatible con aceleración por hardware.

Ahora, ¿en qué beneficia esto? En mucho, si vemos cómo reaccionan los gestos en un ordenador con Windows, nos damos cuenta que todo depende del software pero, como sucede en los dispositivos móviles como el iPad o la Galaxy Tab (que no se mostró porque la aplicación mata Chrome), aún falta desarrollo y hardware que permita lo que se quiere adherir a la web con *Massive Multi-touch*.

Según lo que se comenta en el vídeo, la tecnología pronto estará disponible. Con mejoras como esta (aunque por ser una muestra inicial no nos da mucha alegría, por lo menos en los dispositivos móviles) a HTML5 le espera un futuro brillante para la programación de aplicaciones web basadas en HTML5.

Los navegadores web y los dispositivos móviles se hacen cada vez más sensibles al tacto. Esto hace que los desarrolladores web intenten crear aplicaciones más intuitivas y sofisticadas que se adapten de forma natural con el dispositivo que se utilice. Las interfaces táctiles tienen una larga historia que se remonta a la década de 1960, pero ahora un hardware más rápido, menos costoso y de reciente despliegue en dispositivos móviles ha llevado a una proliferación de diferentes enfoques para el diseño de software de interfaz. Las interfaces táctiles con frecuencia hacen uso de los gestos personalizados para indicar la intención del usuario. La misión del grupo de trabajo de Eventos Web, dentro del W3C, es proporcionar métodos para desarrollar el uso de dispositivos de entrada multi-

táctil, lápiz de Tablet PC, o de otros tipos. El objetivo pasa por determinar un conjunto apropiado de funciones para normalizar y definir las características, de forma que pueden ser desplegados rápidamente, ampliamente, y definen la interoperabilidad entre los dispositivos. Los eventos táctiles (*touch*) son simples, pero pueden tener diferentes características de los eventos del ratón, incluyendo sensibilidad a la presión, sobre los lápiz de TabletPC, se pueden añadir aspectos modales de eventos táctiles, en eventos multitáctiles (*Multitouch*) añadir otra capa, que permite al usuario utilizar dos o más dedos para realizar una tarea, e incluso puede permitir a los usuarios múltiples manipular la interfaz de usuario utilizando el mismo dispositivo de entrada, por ejemplo, una misma pantalla.

Fuentes: <http://gizmologia.com/2011/12/massive-multi-touch-html5> y <http://www.w3.org>



Actividades

1. Realiza una monografía sobre dispositivos multitáctiles y aplicaciones desarrolladas para ellos.

Herramientas de comunicación y colaboración

SUMARIO

- Herramientas de colaboración
- Sindicación web
- Marcadores sociales
- Blogs, wikis y foros
- Redes sociales

OBJETIVOS

- Conocer las aplicaciones web de comunicación y colaboración.
- Instalar y configurar blogs.
- Instalar y configurar wikis.
- Instalar y configurar foros.
- Saber las principales características de las redes sociales.



SOCIAL MEDIA

1 >> Herramientas de comunicación y colaboración

Con la aparición de la Web 2.0 aumentaron las posibilidades de comunicación y de compartir información. Así, se pasó del envío de información a través de los viejos protocolos de comunicación, que permiten el uso del correo electrónico (**SMTP**, **IMAP/POP3**) y de los sistemas de transferencias de ficheros (**FTP**), a las nuevas herramientas de comunicación colaborativas, que presentan un gran potencial y facilitan el trabajo en equipo en cualquier ámbito.

Los usuarios de la Web 2.0 no solo interactúan sino que también participan colaborando en su crecimiento, dado que gestionan sus propios datos y aportan nuevos contenidos.

Así pues, las aplicaciones sociales profundizan en nuevos modelos de comunicación y relaciones sociales, hasta el punto que muchos cambios en la manera de relacionarnos derivan de las tecnologías de la comunicación que se ofrecen con la Web 2.0. La comunicación retroalimenta la Red y permite crear comunidades que colaboran y aportan contenidos enriqueciendo el conocimiento en la Web. De esta forma, podemos usar la Red para buscar y dar a conocer todos aquellos contenidos que nos interesan.

Todo esto es posible gracias a la difusión de contenidos, los marcadores sociales, etc., así como a otras herramientas que permiten la creación y gestión de contenidos, como blogs, wikis, foros y redes sociales.

1.1 > Sindicación web

La **sindicación** o **redifusión web** permite la distribución de contenidos informativos (*feeds*) entre sitios web; la página que ofrece el contenido se denomina **fuentes web** o **canal web**, y proporciona a terceros la actualización de los mismos (por ejemplo, noticias de un periódico, nuevos artículos en un blog, los últimos comentarios en un foro, etc.).

Para leer los *feeds* de un canal web hay que suscribirse al canal, lo que permite acceder a la información de la fuente y mostrarla en la web que la solicitó. Hay dos principales formatos de redifusión web, RSS y Atom; ambos utilizan el lenguaje XML para estructurar la información que comparten.

RSS, siglas de la expresión *Really Simple Syndication* (sindicación realmente simple), se puede considerar un formato de redifusión web que permite acceder a información actualizada de páginas web favoritas y distribuirla, sin necesidad de tener que visitarlas una a una. Permite a una página publicar información a través de los canales o fuentes RSS para que otras páginas puedan acceder a los titulares, noticias o actualizaciones realizadas en la fuente.

Atom es otro formato de redifusión que permite la distribución de contenidos y noticias de sitios web; fue desarrollado como una alternativa a las incompatibilidades de las versiones de RSS. Presenta mejoras respecto a RSS. Un documento Atom es más consistente y puede contener mayor información, ya que se puede indicar si el contenido se trata de texto plano, con etiquetas o si es contenido audiovisual, mientras en RSS no.

SMTP

Son las siglas de *Simple Mail Transfer Protocol*, un protocolo simple de transferencia de correo, que se utiliza para el intercambio de mensajes de correo electrónico.

POP3

Siglas de *Post Office Protocol*, protocolo que permite conectarse a un servidor de correo para descargarse los mensajes de correo electrónico que almacena, eliminándolos del servidor.

IMAP

Siglas de *Internet Message Access Protocol*, otro protocolo de acceso a mensajes de Internet, que también permite acceder a los mensajes electrónicos almacenados en un servidor de correo electrónico, sin descargarlos del mismo.

FTP

File Transfer Protocol, protocolo que permite la transferencia de archivos entre dispositivos conectados en red. Normalmente es necesario un servidor que almacene los archivos y proporcione servicios, a través de este protocolo, como transferir, copiar, modificar o borrar ficheros.



Marcadores sociales



<http://www.delicious.com/>



<http://www.diigo.com/>



<http://www.mister-wong.es/>

Mediante los servicios de marcación social, cualquier persona, incluso sin estar registrada, puede averiguar el número de usuarios que han marcado y etiquetado una página como relevante.

1.2 > Marcadores sociales

Los marcadores sociales, también llamados *social bookmarking*, permiten localizar, almacenar, etiquetar y anotar recursos de Internet para reutilizarlos desde cualquier ordenador, y compartirlos con la comunidad web.

Los navegadores de Internet permiten almacenar y organizar los marcadores deseados, es decir, las direcciones URL que enlazan con las páginas web de especial interés para el usuario del navegador. Por ejemplo, en el navegador Internet Explorer, para localizar una página ya visitada, hay que utilizar la opción *Favoritos*, en cambio en otros navegadores, como Firefox o Chrome, la opción que se ofrece al respecto se denomina de la misma manera, *Marcadores*. Pero los usuarios tienen un problema al utilizar dispositivos diferentes al habitual para navegar por Internet, ya que no tienen acceso a los marcadores que previamente almacenaron y organizaron en su navegador. Lo mismo ocurre cuando se utiliza un navegador diferente al habitual.

Los marcadores sociales permiten solventar estos problemas poniendo a disposición de los usuarios los enlaces que previamente han almacenado, organizado y etiquetado, facilitando la posterior localización y compartición de esos recursos de la web.

Existen multitud de sitios web que ofrecen servicios de gestión de marcadores sociales; unos son generales, orientados a almacenar cualquier tipo de recurso disponible en Internet, mientras que otros están más especializados en algún área en concreto. Algunos ejemplos representativos son:

- **Delicious:** es una aplicación web que sirve para gestionar los marcadores sociales de la Web. Permite guardar, ordenar, compartir y buscar enlaces a páginas web de interés personal, almacenándolas en una colección. Utiliza un sistema de etiquetas para poder ordenar los enlaces en diferentes categorías, lo que facilita encontrar nuevos sitios web y seguir los últimos descubrimientos de personas que comparten intereses comunes. Es la herramienta de marcadores sociales líder en cuanto al número de usuarios y es muy sencilla de usar.
- **Diigo:** es un marcador social muy versátil, no solo ordena y clasifica enlaces favoritos, sino también imágenes y notas personales. Permite comentar las páginas marcadas así como añadir notas flotantes y remarcar fragmentos de una página estática. Se trata de una herramienta muy completa, con la que también podemos clasificar documentos, audio, etc.
- **Mister Wong:** ofrece interfaz en castellano y permite crear grupos y coleccionar, junto con otros usuarios, marcadores relacionados con diferentes temas, discutir los marcadores y crear una red de contactos.

Actividades propuestas

- 1• Busca en tus páginas web favoritas el icono del canal RSS y suscríbete a ellas en tu navegador.
- 2• Crea una cuenta de usuario en Delicious, y elabora una colección de enlaces haciendo clic en *Create Stack*.
- 3• Clasifica los enlaces recién creados añadiendo sobre ellos etiquetas que los representen.

1.3 > Blogs

Un blog es un sitio web que permite compartir experiencias y conocimientos.

Se suele actualizar de manera periódica, a diario, semanalmente, mensualmente, etc. Es un medio de recopilar y compartir artículos, compuestos de texto, imágenes y cualquier otra información. Una de sus características principales es que todas las entradas publicadas en el blog se almacenan y muestran de manera cronológica, apareciendo primero la más reciente que el autor ha compartido. El autor tiene siempre la libertad de mantener, eliminar o modificar los artículos publicados. El término blog proviene de las palabras inglesas *web* y *log*, ya que consiste en un registro de eventos a través de una página web. Normalmente, dispone de un sistema de comentarios, que facilita a los lectores del blog establecer una conversación con el autor y otros lectores, acerca de lo publicado.

Entre las aplicaciones web diseñadas para crear blogs hay que destacar Blogger y WordPress.

Microblogging

Es otra forma de comunicación en Internet, cuya fuerza radica en su sencillez. Consiste en enviar mensajes breves, de 140 caracteres como máximo, lo cual agiliza la lectura y la comunicación. Es una combinación de foros, blogs, chats y el “estado” (que se utiliza en el Messenger), todo ello desde cualquier dispositivo con acceso a Internet. *Twitter* es quizás el sistema de *microblogging* más popular, pero no es el único. Actualmente, se están sentando las bases para diseñar un sistema estándar y abierto en el cual no dependamos de que nuestros amigos estén en nuestro mismo sistema de *microblogging*, de forma que se convierta en un servicio más de Internet, al igual que la Web, el FTP, el correo electrónico, el P2P, etc. Hay sistemas que no se quedan en el simple texto e integran contenidos fotográficos y multimedia, con lo que la diferencia entre las redes sociales y el *microblogging* cada vez es menor.



Cuaderno de bitácora

Se denomina así a los blogs en los que el autor escribe como si fuese un diario de su vida y lo comparte en la Red. El nombre de *cuaderno de bitácora* tiene su origen en los cuadernos o libros en los que se anotaban todos los acontecimientos de importancia que ocurrían durante una navegación marítima, los cuales se guardaban en un armario, denominado bitácora, para protegerlo de las inclemencias del tiempo.



Crea un blog. Es gratuito.



Plantillas y diseños atractivos y personalizables [Probar el diseñador de plantillas](#)



Estadísticas actualizadas al minuto.



Gana dinero con AdSense.



Explora los Blogs importantes.

www.blogger.com



WORDPRESS.COM

A better way to blog.

Get started here

[Learn more](#) or [sign up now](#)

<http://es.wordpress.com/>

Casos prácticos

1

Creación de un blog desde Blogger

•• Practica la creación de un blog con la aplicación web de Google, Blogger.

Solución ••

- Para utilizar la herramienta de Blogger hay que acceder al sitio web www.blogger.com
 - Para crear un blog en Blogger es necesario tener una cuenta de correo de Google. Si ya dispones de ella, introduce tu cuenta de correo de Google y la contraseña, y después pulsa en *Iniciar sesión*. Si no la tienes, pulsa en *Comenzar* para crearte una.
- Si dispones de una cuenta de correo de Google e inicias sesión por primera vez en Blogger, debes completar algunos datos para registrarte en Blogger con dicha cuenta de correo.

1

Registrarse en Blogger



Quando haya completado este proceso, podrá acceder a Blogger mediante la dirección de correo electrónico y contraseña de su cuenta de Google.

Dirección de correo electrónico:	AlumnoAW@gmail.com	Utilizar una cuenta diferente
Nombre:	Alumno AW	
Nombre visible:	Alumno AW	Nombre utilizado para firmar sus entradas del blog
Sexo:	Hombre	
Aceptación de las condiciones:	<input checked="" type="checkbox"/> Acepto las Condiciones del servicio .	Indica que has leído y comprendes las condiciones de servicio de Blogger.

Si por el contrario no dispones de cuenta de correo de Google y haces clic en *Comenzar*, el proceso de crear una cuenta en Google, se te solicitarán algunos datos para registrarte en Blogger.

1

Crear una cuenta de Google



Este proceso creará una cuenta de Google que podrás utilizar con otros servicios de Google. Si ya tienes una cuenta de Google, como de Gmail, Grupos de Google u orkut, [inicia sesión](#).

2

Asignar un nombre al blog

Título del blog:	Aplicaciones Web	El título de tu blog aparecerá en el blog publicado, en el escritorio y en el perfil.
Dirección del blog (URL):	http://WebApls.blogspot.com	Comprobar la disponibilidad

2

Escoger una plantilla de inicio

Puedes cambiar la plantilla más tarde e incluso personalizarla con el Diseñador de plantillas.


Vistas dinámicas

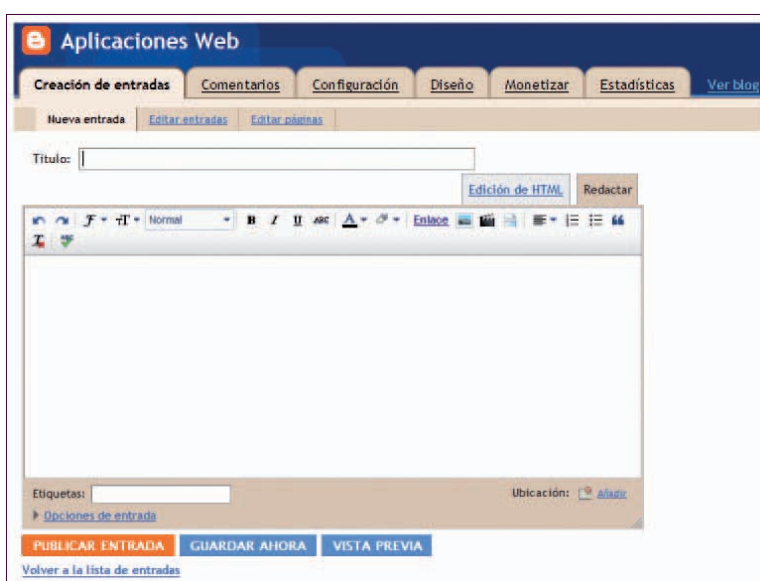

Sencillo


Picture Window


Fantástico, S.A.



- Una vez que hayas escogido una plantilla de inicio puedes empezar a publicar tu primer blog; para ello, haz clic en el enlace naranja **EMPEZAR A PUBLICAR**.



En este entorno tienes varias pestañas. En la pestaña que ves en la imagen superior puedes crear la primera entrada en tu blog. Utilizando el resto de pestañas se puede gestionar la configuración del blog.

- Escribe un título para la primera entrada de tu blog y redacta su contenido. Por ejemplo, puedes expresar tus sensaciones sobre el proceso de creación de un blog. Utilizando los enlaces de la parte inferior, guarda y publica tu entrada. Posteriormente, visualiza la entrada haciendo clic en el enlace de **VISTA PREVIA**.
- A la derecha de estas pestañas verás que hay un enlace como este [Probar la interfaz actualizada de Blogger](#), que te permite utilizar otro entorno de edición de Blogger más actualizado.

Actividades propuestas

- 4.. Pulsa en el enlace que permite acceder al nuevo entorno de edición que ofrece Blogger, investiga y explora por las diferentes opciones de esta actualizada interfaz.
- 5.. Compara las ventajas y desventajas de ambas interfaces, en cuanto a la utilidad y accesibilidad de sus diversas opciones.
- 6.. Utilizando la nueva interfaz de Blogger, crea un blog y publica entradas en él sobre tus vivencias del centro.

Wiki

Término hawaiano, que significa “rápido”. Normalmente, una página web concreta se denomina “**página wiki**”, mientras que un conjunto de páginas interconectadas mediante hipervínculos, es “**el wiki**”, pero ambos términos se usan indistintamente.

Wikipedia

El wiki más grande del mundo. Toda esta enciclopedia virtual ha sido escrita por voluntarios. Alguien interesado en un tema inicia un artículo y el resto de la comunidad puede añadir contenidos, editar el trabajo de otro o añadir otra página de contenidos. Está creada con MediaWiki, una herramienta sencilla, que tiene ciertas reglas de edición de vínculos; por ejemplo, si se escribe el título de una página que queremos enlazar de la propia wiki, en algún lugar entre el texto del wiki, encerrando el nombre entre dobles corchetes (Ej. `[[paginaWiki]]`), esta palabra se convierte en un hipervínculo interno a otra página del wiki.

1.4 > Wikis

Es una herramienta de comunicación colaborativa, simple, flexible y potente de la Web 2.0. Está compuesto de varias páginas web, cada una de las cuales almacena información de interés de cualquier tipo. Se puede utilizar para crear repositorios digitales, listas de enlaces web debidamente organizadas o incluso enciclopedias. En un wiki, la información no sigue un orden cronológico, como en los blogs. La creación de las páginas se hace de una manera colaborativa, de tal manera que diferentes usuarios, a través de un navegador web, según los permisos otorgados, contribuyen para crear y/o editar las páginas web que conforman el wiki.

Cada página del wiki tiene un título que es único, de tal manera que sirve para diferenciarla del resto; así, el título de una página se puede convertir en un enlace a la página web que referencia. En el wiki, puede existir un sistema histórico de almacenamiento de las páginas modificadas para poderlas recuperar si fuera necesario.

Hay wikis públicos, donde se puede colaborar sin tener registrada una cuenta de usuario, aunque a veces se pueden requerir permisos para realizar ciertas ediciones, convirtiéndose entonces en wikis privados que requieren autenticación de usuario y no permiten ediciones anónimas.

Algunos sitios web donde ofrecen el servicio gratuito y se pueden crear wikis de manera fácil, son, por ejemplo, Wikispaces y PBwiki. Otra opción es instalar aplicaciones específicamente diseñadas para la creación de wikis, que permiten el control absoluto del wiki, como, por ejemplo, Docuwiki y MediaWiki.



<http://www.wikispaces.com>



<http://pbworks.com>



<http://www.mediawiki.org/wiki/MediaWiki/es>



<http://www.dokuwiki.org>

Actividades propuestas

7•• Accede a <http://creandowikis.wikispaces.com> y consulta los tutoriales sobre wikis que aparecen a la izquierda.

8•• Accede al sitio de Wikispaces, regístrate y crea un wiki a través de sus servicios.

1.5 > Foros

Un foro es otra aplicación web orientada a la comunicación, anterior a los blogs y a los wikis, que permite establecer un diálogo abierto en torno a un tema de interés común, fomentando la participación.

Los usuarios de un foro, también llamado **foro de discusión**, pueden expresar sus ideas y escribir sus comentarios, a través de los diferentes hilos de opinión. Estos comentarios pueden registrarse de manera cronológica bajo su correspondiente hilo.

En muchas ocasiones, los foros se integran dentro de los portales de información de Internet, lo que permite a su comunidad de usuarios colaborar y opinar en el propio portal, participar en discusiones sobre temas de interés común y compartir información en el sitio web.

En la misma aplicación, el administrador del sistema puede definir varios foros, cada uno de los cuales contendrá todas las discusiones que los usuarios van generando, bien al responder a las opiniones ya mostradas sobre algún tema, o bien comenzando un nuevo tema.

Las intervenciones suelen estar coordinadas por un moderador, que puede ser el administrador del foro o el mismo usuario que introduce el tema y que, por lo tanto, realiza la primera pregunta o expresa la primera idea con un comentario. A veces interesa que el moderador estimule y guíe la discusión, sin presionar, otorgue la palabra, pida fundamentos y explicaciones, para, con todo ello, mantener el buen clima y poder finalmente sintetizar lo expuesto antes de cerrar la discusión.

Para el buen funcionamiento del foro, un moderador debe controlar a:

- Los *Trolles*, advirtiéndolo a este tipo de usuarios sobre el mal uso que hagan del foro para que no incluyan comentarios no apropiados que alteren la dinámica cordial del foro.
- Los *Leechers*, amonestando a los usuarios que solo desean hacer uso de los recursos que comparten los otros miembros del foro, sin agradecerlos ni aportar nada, para que contribuyan con sus ideas.
- Los *Arqueólogos*, indicando a los usuarios que rescatan antiguas entradas del foro que son temas ya cerrados, por lo que no hace falta abrirlos de nuevo para evitar repetir discusiones.
- Los *Chaters*, aconsejando a estos usuarios que escriban utilizando un lenguaje correcto, que respete la ortografía, para que los mensajes del foro sean entendibles por todos.
- Los *Fakes*, para evitar que se suplante la personalidad de otros usuarios del foro.
- Los *Títeres*, detectando a los usuarios inscritos en el foro con diferentes nombres de usuario.
- Los *Newbies*, ayudando a los recién llegados para que asimilen rápidamente las normas de funcionamiento del foro.
- Los *Bullyers*, amonestando a los usuarios que, sin argumentos, se decantan de manera brusca en contra de opiniones razonadas en los debates, reaccionando con insultos y/o agresividad.

Administrador de un foro

Debe protegerlo de los ataques que vayan orientados a romper los sistemas de seguridad, y debe combatir el *spam*, eliminando la publicación de mensajes no solicitados, publicitarios o contrarios a las reglas del foro.

Diferencia entre *Troll* y *Bullyer*

En ocasiones, se confunde la forma de comportarse de un *Bullyer* con la de un *Troll*, aunque son algo distintas.

El comportamiento de un *Bullyer* puede llegar a amedrentar a un usuario, acosado de manera continua, provocando una situación que desvirtúa la función principal de opinión y debate de los foros, ya que incluso el usuario amenazado por el *Bullyer* puede pensar, por el silencio de los demás usuarios, que el acosador tiene el apoyo del resto.

Por el contrario, este problema no se da tanto con un *Troll*, porque normalmente son los propios usuarios los que se ponen en su contra recriminándole directamente los comentarios fuera de lugar que está escribiendo en el blog.



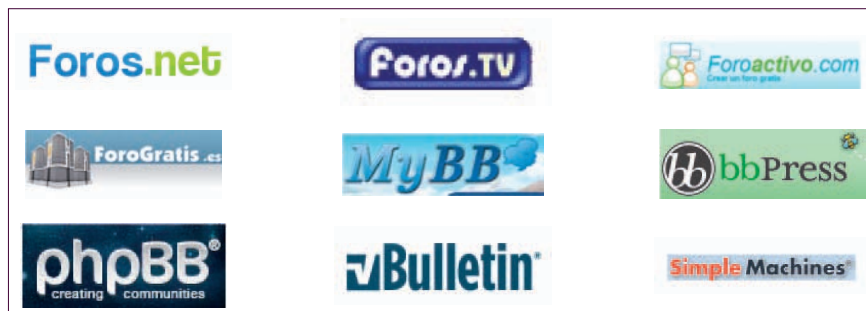
phpBB (PHP Bulletin Board)

Es un paquete de software libre (GNU) que permite gestionar varios foros.

Como indica su nombre, está implementado en el lenguaje PHP, lo que permite que sea fácilmente adaptable y configurable, así como una sencilla personalización a la hora de crear diferentes foros para distintas comunidades de usuarios.

Existen varias posibilidades para crear un foro de discusión. Quizás la más sencilla sea crear un foro a través de alguna de las páginas web que ofrecen este servicio de manera gratuita, aunque este modo suponga tener publicidad impuesta por la plataforma que ofrece el servicio. Algunos ejemplos son: foros.net, foros.tv, foroactivo.com, forogratias.es, etc.

Otra opción es instalar, configurar y mantener alguna aplicación que permita la creación de foros y tener el control sobre ellos; por ejemplo, phpBB, MyBB, bbPress, SMF o vBulletin, este último de pago.



Casos prácticos

2

Instalación de la aplicación de foros phpBB, sobre XAMPP

• Realiza la instalación de la aplicación phpBB, para crear y administrar foros, tanto en un sistema operativo Linux, como en un sistema Windows.

Solución •

Para realizar la instalación de phpBB sobre nuestro XAMPP, en ambos sistemas operativos, lo primero que tienes que hacer es acceder al CD *Recursos* y descargar los ficheros:

- Windows: phpBB-3.0.10.zip
- Linux: phpBB-3.0.10.tar.bz2

En la página oficial en español de phpBB, podrás obtener la última versión.

Para realizar la instalación, seguiremos los siguientes pasos:

1. Descomprime el fichero recién descargado que contiene la carpeta *phpBB3* con todos los ficheros de configuración necesarios en su interior.
2. Copia esta carpeta *phpBB3* en la carpeta donde esté *\xampp\htdocs*
3. Abre el navegador y abre la dirección <http://localhost/phpBB3> para comenzar la instalación de phpBB.



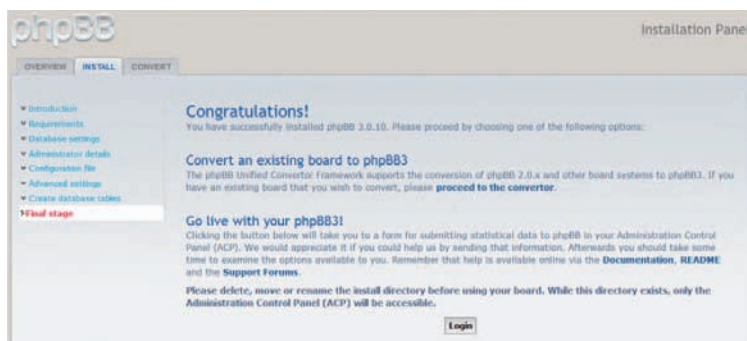
4. Antes de comenzar la instalación de phpBB, necesitaremos una base de datos donde ubicar la información del foro. La crearemos con phpMyAdmin, tal y como lo hicimos en el Caso final de la Unidad didáctica 1.

Para ello, abre otra pestaña del navegador con la dirección <http://localhost/xampp/>; una vez allí, haz clic en el enlace *phpMyAdmin*, la herramienta de administración de las bases de datos MySQL de XAMPP, y crea la base de datos; por ejemplo, le pones el nombre *phpbb3*, y haz clic en *Crear*.



Un mensaje te dirá que la base de datos se ha creado correctamente. Ahora vuelve a la pestaña de la instalación de phpBB3 (<http://localhost/phpBB3>).

5. Una vez allí, selecciona la pestaña *INSTALL*.
6. Haz clic en *Proceed to next step*.
7. Antes de proceder con la instalación de phpBB, comprueba la compatibilidad de la instalación para poder instalar y ejecutar phpBB. Baja hasta el final y, si se cumplen los requisitos exigidos, pulsa en *Start install*.
8. Indica los siguientes datos y haz clic en el botón *Proceed to next step*.
9. Verás el siguiente mensaje: *Test connection: Successful connection*
10. Haz clic en el botón *Proceed to next step* y configura los datos del administrador phpBB rellenando todos los campos solicitados. Recuerda la contraseña que indiques; luego la necesitarás.
11. Haz clic en el botón *Proceed to next step*, si no te falta ningún dato verás el mensaje: *Check administrator settings: Tests passed*; haz clic en el botón *Proceed to next step*.
12. Te confirmará sobre el fichero de configuración, de nuevo haz clic en el botón *Proceed to next step* y en la siguiente ventana otra vez clic, dejando los valores por defecto sobre estos ajustes avanzados.
13. Verás otro mensaje de confirmación. Ahora, sobre la base de datos, haz clic de nuevo en el botón *Proceed to next step* para finalizar la instalación de phpBB.
14. Fíjate en esta ventana final de la instalación; al final hay un mensaje muy importante: **debes borrar, mover o cambiar el nombre del directorio de instalación (carpeta `xampp\htdocs\phpBB3\install`) antes de usar el foro de phpBB**. Mientras que este directorio exista, solo será accesible el panel de administración.



15. Solo queda volver a entrar de nuevo en la dirección <http://localhost/phpBB3> para comenzar a utilizar el foro phpBB.





Instalación del paquete de idioma español en phpBB sobre XAMPP

- Realiza la instalación del paquete del idioma en español para el gestor de foros phpBB.

Solución • Para realizar la instalación del paquete de phpBB que permitirá utilizar el foro en el lenguaje castellano, en ambos sistemas operativos, hay que acceder al CD *Recursos* y descargar la versión del paquete del Lenguaje Español 3.0.10 (phpBB-3.0.10_Es_Usted.zip). En la página oficial en español de phpBB, podrás obtener la última versión.

Para realizar la instalación, seguiremos los siguientes pasos:

1. Descomprime el fichero recién descargado y sube el contenido al servidor donde está instalado **phpBB3**. En el caso de **subSilver2**, subir dicho contenido a esta ubicación `/styles/subsilver2/imageset/es/`; para **proSilver**, el proceso es igual, subir el contenido a `/styles/prosilver/imageset/es/`. No olvides subir el idioma a la ubicación `/language/es/` para que los botones aparezcan correctamente, y todo funcione como es debido.
2. Para instalar el paquete del idioma español en el foro phpBB3, abre el navegador y, a continuación, abre la dirección <http://localhost/phpBB3>. Accede con el nombre de usuario y la clave de usuario administrador que configuraste en el paso 13 del anterior Caso práctico.

3. Una vez dentro, haz clic en el enlace *Administration Control Panel*, que verás abajo del todo.
4. Confirma de nuevo la contraseña para acceder al panel de control de administración.
5. Elige la pestaña *SYSTEM* y después *Language packs*, en la parte de la izquierda; finalmente, haz clic en *Install*, para comenzar a instalar el paquete del idioma español.
6. Un mensaje en verde te informará sobre la correcta instalación del paquete del idioma español. Ahora debes salir del *Administration Control Panel*, haciendo clic en *ACP Logout*, que verás a la izquierda, y entrar en el panel de control de usuario, *User Control Panel*, para acceder y cambiar la configuración del idioma recién instalado.
7. Elige ahora la pestaña *Board preferences*, en el panel de control de usuario, la opción *Edit global settings* de las opciones que te aparecen abajo a la izquierda y, por último, en la lista desplegable de los lenguajes instalados en el sistema, *My language:*, selecciona el idioma español.
8. Confirma los datos haciendo clic en *Submit* y, a partir de ahora, este usuario administrador tendrá configurado su idioma de uso del foro phpBB en el lenguaje español.

Actividades propuestas

- 9• Utiliza el foro phpBB que acabas de instalar agregando temas y miembros.
- 10• Busca e instala la última versión de phpBB, compárala con la 3.0.10 utilizada y destaca sus ventajas.

2 >> Comparativa de blogs, wikis y foros

Si realizamos una comparativa entre estas herramientas colaborativas de la web podemos obtener un resumen de las características que las diferencian.

2.1 > Wikis y blogs

Los wikis y los blogs presentan varias diferencias:

- **Contenido.** Los wikis normalmente se van creando con los contenidos aportados por diferentes autores, que colaboran conjuntamente en el tema del wiki por algún interés común. En cambio, los contenidos publicados en los blogs suelen ser propiedad de una única persona, que se encarga de dotarle del contenido deseado.
- **Utilidad.** Los blogs están orientados a fomentar un debate sobre el tema del contenido publicado inicialmente. Por el contrario, la utilidad de los wikis radica en colaborar y compartir información.
- **Estructura.** La información que compone un wiki se estructura de manera flexible, según el contenido de cada una de las páginas que forman dicho wiki, mientras que en un blog la información se estructura de manera cronológica, de forma que se visualiza primero la última información aportada.
- **Acceso.** Normalmente, la colaboración en los blogs se limita a realizar comentarios a los textos publicados por el autor del blog, pudiéndose configurar para que los comentarios se hagan de forma anónima o bien solo por usuarios registrados. En cambio, los wikis suelen estar abiertos más a la colaboración de todo el mundo, aunque también se pueden establecer protecciones de acceso para hacerlos más restrictivos.
- **Temporalidad.** En un wiki, al ser una herramienta colaborativa, la información es más susceptible de estar en continua evolución (los usuarios continuamente pueden ampliar la información de un artículo). Sin embargo, en los blogs, suele ser el autor el que publica la información (organizada temporalmente), por lo que en raras ocasiones las entradas ya publicadas se sustituyen por otras.

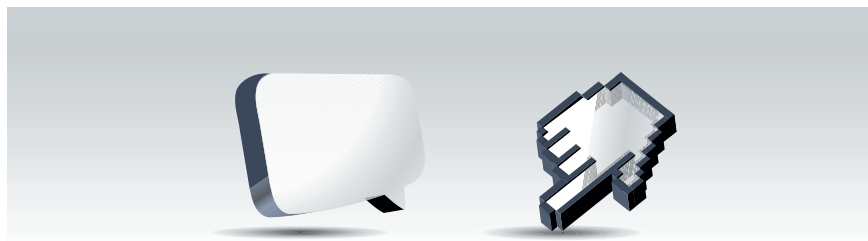
BLOGS	WIKIS
Generalmente único autor	Generalmente muchos autores
Fomentan el debate	Promueven la colaboración
Estructura cronológica	Estructura determinada por el contenido
Más estructurado	Más flexible
Protegido, colaboración limitada	Abierto, puede protegerse
No se suelen editar entradas antiguas	Se suelen actualizar continuamente
Motores de búsqueda internos opcionales	Incluyen mecanismos de búsqueda integrados
Utilizan RSS para informar sobre contenido nuevo	Notificaciones mediante correo electrónico
Editores WYSIWYG (<i>What You See Is What You Get</i>)	Lenguaje de marcas simplificado

- **Búsqueda de información.** Los blogs suelen ofrecer la posibilidad de usar motores de búsqueda de información, mientras que en un wiki esta funcionalidad normalmente viene integrada por defecto.
- **Notificaciones.** En un wiki, las notificaciones sobre nuevos contenidos se hacen a través de e-mail, mientras que en los blogs a menudo se usan servicios de sindicación de contenidos (RSS).
- **Edición.** Los blogs suelen utilizar editores de tipo WYSIWYG (*What You See Is What You Get*), en los cuales puedes ir comprobando cómo se verá la información mientras la estás editando. En los wikis, para editar la información publicada habitualmente se utilizan lenguajes de marcas simplificados.

2.2 > Wikis y foros

Los wikis y los foros también tienen diferencias importantes:

- **Actualización de contenido.** Los contenidos de los foros no suelen cambiarse, pero depende de los permisos otorgados por el administrador; estos permisos se pueden restringir, de forma que solo se permite añadir contenido nuevo al foro. Por el contrario, el contenido de la mayoría de las páginas de un wiki se puede cambiar, ya que colaborar es uno de los objetivos de un wiki, aunque también se pueden controlar las modificaciones publicadas.
- **Información almacenada.** En los foros, los usuarios deben buscar entre la información disponible, que puede ser antigua, incorrecta o estar mal escrita o desactualizada. En cambio, en los wikis, los usuarios pueden corregir la información incorrecta y mejorar así la presentación de la información bajo la supervisión de un administrador para que los artículos no contengan información incorrecta.
- **Enlaces.** En los foros no se pueden añadir enlaces a otros mensajes o entradas del foro, a no ser que se haga de manera manual copiando la ruta, mientras que en un wiki sí se pueden crear enlaces entre las distintas páginas.



Actividades propuestas

- 11• Realiza una comparativa que recoja las semejanzas y diferencias existentes entre los blogs y los foros; para ello, puedes buscar información complementaria en Internet.
- 12• Haz un resumen con las semejanzas y diferencias entre wikis, blogs y foros.
- 13• Debate con tus compañeros sobre qué herramienta (wiki, blog, foro) te parece más útil y por qué.

3 >> Redes sociales

En Internet existen muchas aplicaciones web, denominadas redes sociales, que permiten la creación de comunidades y la cooperación y comunicación entre sus miembros, que pueden así poner en común sus comentarios, mensajes, etc., a través de la Red.

Las redes sociales nos ofrecen la posibilidad de comunicarnos con otras personas, incluso aunque estas sean desconocidas, dado que proporcionan herramientas internas de búsqueda de usuarios, que permiten encontrar todas las personas que pertenecen a la propia red social. En las redes sociales, basta enviar a otro miembro de la red una invitación y que este la acepte para formar parte de su círculo de amistad. En algunas redes, la aceptación es incluso automática.

Las redes sociales se consideran sistemas abiertos, dado que se van construyendo con la información, comentarios, fotografías, enlaces o cualquier otro tipo de contenido, que cada usuario va incluyendo en la red y desea compartir con los contactos con los que se relaciona en la misma.

Un aspecto importante a tener en cuenta es el de la privacidad y la seguridad al registrarse y participar en redes sociales, ya que los usuarios exponen ciertos datos de carácter personal. Así, según las prestaciones de seguridad que ofrezca la red social y la configuración que el propio usuario establezca sobre la privacidad de sus datos, la información que ofrezca un usuario será más o menos pública.

El número de menores que utiliza las redes sociales en la UE sigue en constante aumento: se estima que en torno a un 80% de jóvenes con edades comprendidas entre 13 y 16 años y un 38% entre los de 9 a 12 años. Por ello, es necesario que las redes sociales presenten unas condiciones de registro, utilización y códigos de conducta fáciles de entender por los menores, diseñadas para su mayor protección. En consecuencia, la participación de menores de edad en redes sociales debe ser supervisada por adultos, para mantenerlos protegidos de posibles abusos.

Las redes sociales siguen buscando nuevos modos de posicionarse en el mercado; mientras unas intentan mejorar la parte más social, a otras les preocupa no disponer de una tecnología de búsqueda acertada y tener que depender casi de forma exclusiva de un buscador externo. Las nuevas redes sociales intentan competir ofreciendo una interfaz mucho más visual, para resaltar la participación del contenido multimedia, con búsqueda de vídeos y chateo por videoconferencias.

Redes sociales en Internet

Algunos ejemplos de las redes sociales más representativas y populares son:



Myspace



LinkedIn



Facebook



Google+



Tuenti



Socl

Actividades propuestas

14 Realiza una comparativa sobre las condiciones de registro, utilización y códigos de conducta que ofrecen las redes sociales más populares.


15 Elabora una lista con todas las redes sociales que conozcas y añade otras nuevas buscando información en Internet y compartiéndola con tus compañeros de clase.

Actividades finales

.: CONSOLIDACIÓN .:

- 1•• Suponiendo que te encargan que habilites un sistema para registrar en la Web, de manera cronológica, los diferentes eventos solidarios que realiza tu empresa, para que puedan ser conocidos y comentados, ¿cuál es la herramienta de comunicación que utilizarías? Razona tu respuesta.
- 2•• ¿Cuáles son las características diferenciadoras de un blog?
- 3•• ¿Qué se necesita para poder crear un blog con Blogger?
- 4•• ¿En qué se fundamentan los wikis?
- 5•• ¿En qué se diferencian los wikis de los blogs?
- 6•• Busca información en Internet sobre los tipos de vínculos que se pueden incluir en MediaWiki.
- 7•• Investiga para qué sirven las *Magic Words* de MediaWiki y realiza una clasificación sobre ellas.
- 8•• ¿Qué es un foro de discusión? ¿Cuál es su utilidad?
- 9•• ¿Por qué en un foro resulta necesaria la figura de un moderador?
- 10•• ¿Cuáles son las funciones de un moderador?
- 11•• Haz un listado con los tipos de usuarios que debe controlar un moderador de un foro.

.: APLICACIÓN .:

- 1•• Accede a la página de WordPress (<https://es.wordpress.com/>), haz clic en *Empieza aquí* y crea un blog en WordPress.
- 2•• Publica una entrada en el blog de WordPress que acabas de crear, en el cual resumas las semejanzas y diferencias encontradas respecto al registro, uso y publicación del primer blog en Blogger realizado en el Caso práctico 1 de esta Unidad didáctica.
- 3•• Investiga en Internet y haz una lista con todas las herramientas que encuentres que permiten publicar un blog.
-  4•• Al igual que hiciste en el Caso práctico 2 con la instalación del gestor de foros phpBB, el sistema de foros WordPress también puede ser instalado sobre XAMPP, para poder gestionar y almacenar blogs desde tu ordenador; realiza la instalación del contenido del fichero wordpress-3.3.2-es_ES.zip que encontrarás en el CD *Recursos*.
- 5•• Busca en Internet (<http://es.wordpress.org/download/>) la información necesaria y edita el fichero wp-config-sample.php de configuración para terminar la instalación de tu WordPress.
- 6•• Entra en la página de PBworks, <https://plans.pbworks.com/academic>, selecciona la oferta gratuita, regístrate eligiendo una dirección y crea un wiki.
- 7•• Dile a tu compañero que aporte información a tu wiki y aporta tu información a la suya; para ello tendréis que enviaros solicitudes de edición, las cuales os tendréis que conceder, a través de las gestión de los usuarios, para permitirlos mutuamente acceder al espacio de trabajo y editar el wiki.
- 8•• Entra en el foro phpBB3 que instalaste en el Caso práctico 1 e inicia un nuevo foro de debate donde invites a tus amigos a discutir sobre el uso de los blogs, wikis y foros en Internet.

Caso final

4

Instalación de MediaWiki

•• Vamos a realizar la instalación de la aplicación web MediaWiki sobre XAMPP.

Solución ••

Debemos tener arrancados los servidores Apache y MySQL de XAMPP en nuestro equipo.

- A continuación, hay que **crear la base de datos** para el wiki. Para ello (puedes consultar el Caso final de la Unidad didáctica 1), abre en el navegador la dirección <http://localhost/xampp/>, entra en phpMyAdmin (en la barra lateral izquierda). Crea una nueva base de datos (tendrás que introducir un nombre para la nueva base de datos; por ejemplo, "wiki_bd", y haz clic en *Crear*). Haz clic en la base de datos recién creada, en el panel izquierdo.



- Después, haz clic en *Privilegios* (icono de la barra superior, o dentro de la opción *Más*).
- Haz clic en *Agregar un nuevo usuario*.

Escribe un nombre para el usuario (por ejemplo, usuario_wiki) y una contraseña. Puedes dejar todo el resto en blanco. Anota tu nombre de base de datos, nombre de usuario y contraseña, porque luego lo necesitarás para instalar MediaWiki.

- Comprueba todos los privilegios globales y haz clic en *Continuar*.
- **Configura los ficheros necesarios de MediaWiki (1.19.0)**
- Extrae los ficheros de *MediaWiki-1.19.0.tar.gz*; por ejemplo, en la carpeta `\xampp\htdocs\mediawiki\`. Abre el navegador en la página correspondiente (<http://localhost/mediawiki/>). Haz clic en el enlace **set up the wiki** y sigue las instrucciones para configurar la instalación de MediaWiki.
- Indica tu idioma y el idioma del wiki.



- Haz clic en *Continuar*, MediaWiki te da la bienvenida y realizará comprobaciones básicas; verás que aparece el siguiente mensaje: **El entorno ha sido comprobado. Puedes instalar MediaWiki.**
- Pulsa en *Continuar*, para conectar Mediawiki a la base de datos. Para ello, introduce el nombre de base de datos, nombre de usuario y la contraseña que escribiste durante la "creación de la base de datos". Usa la cuenta administrador, *root*, de MySQL y haz clic en *Continuar*.

Tipo de base de datos

☒ MySQL
☐ SQLite

Configuración de MySQL

Servidor de la base de datos:
[help](#)
localhost

Identifique este wiki

Nombre de base de datos:
[help](#)
wiki_bd

Prefijo para las tablas de la base de datos:
[help](#)

Cuenta de usuario para instalación

Nombre de usuario de base de datos:
[help](#)
root

contraseña de base de datos:
[help](#)

MediaWiki 1.19.0 instalación

Configuración de la base de datos

Cuenta de base de datos para acceso Web:
☒ Utilizar la misma cuenta que en la instalación

Motor de almacenamiento:
☒ InnoDB
☐ MyISAM

Conjunto de caracteres de la base de datos:
☒ Binario
☐ UTF-8

[Atrás](#) [Continuar](#)

- Deja las opciones por defecto en la configuración de la base de datos. Haz clic en *Continuar*, indica el nombre del wiki, y la cuenta de administrador, usuario y contraseña.

Nombre del wiki:
[help](#)
Blogs wikis y foros

Espacio de nombre de proyecto:
[help](#)
☒ Igual como el nombre del wiki: Blogs_wikis_y_foros
☐ Proyecto
☐ Otro (especificar)

Cuenta de administrador

Su nombre:
[help](#)
usuario_wiki

Contraseña:
.....

Opciones

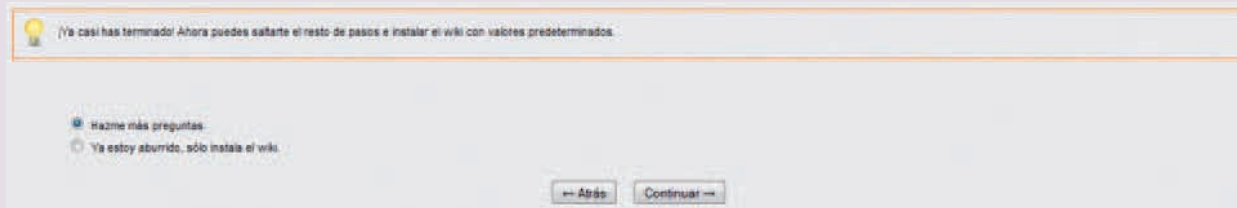
Perfil de derechos de usuarios:
☒ Wiki tradicional
☐ Creación de cuenta reportada
☐ Solo usuarios autorizados
☐ Wiki privado

Wiki work best when you let as many people edit them as possible. In MediaWiki, it is easy to review the recent changes, and to revert any damage that is done by naive or malicious users. However, many have found MediaWiki to be useful in a wide variety of roles, and sometimes it is not easy to convince everyone of the benefits of the wiki way. So you have the choice. A Wiki traditional allows anyone to edit, without even logging in. A wiki with Creación de cuenta reportada provides extra accountability, but may deter casual contributors. The Site administrators authorized scenario allows approved users to edit, but the public can view the pages, including history. A Wiki privado only allows approved users to view pages, with the same group allowed to edit.

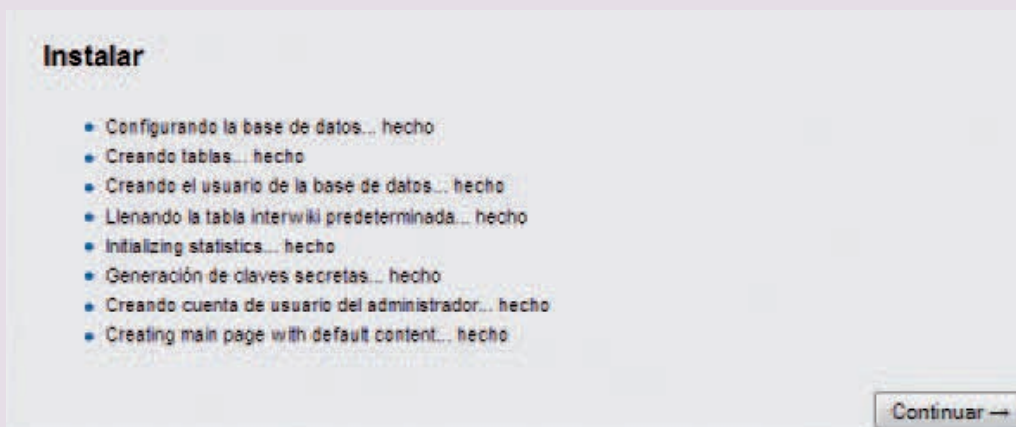
More complex user rights configurations are available after installation, see the [wiki user rights manual](#).

Copyright and license:
☐ Creative Commons Attribution
☐ Creative Commons Reconocimiento Compartir igual
☐ Creative Commons Reconocimiento Compartir igual-no comercial
☐ Creative Commons Zero (Public Domain)
☐ Dominio Público
☐ GNU Free Documentation License 1.3 or later
☒ Propia licencia
☐ Selecciona una licencia personalizada de Creative Commons

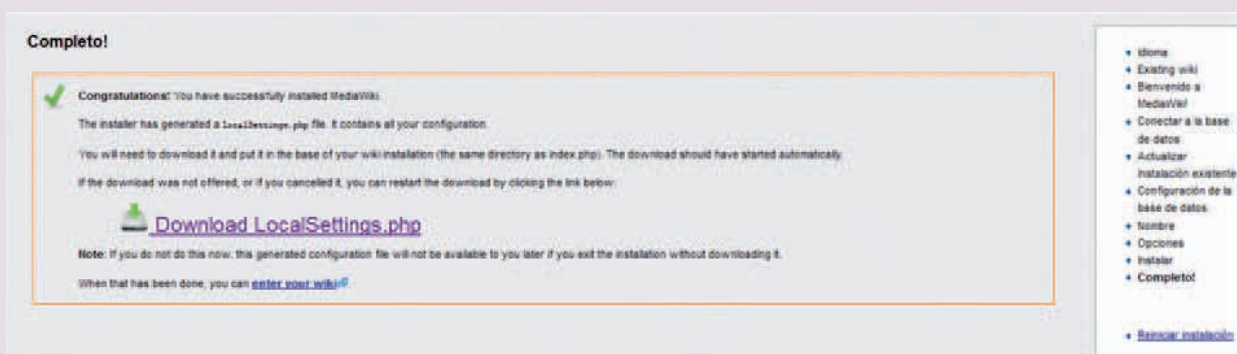
- Pulsa de nuevo en *Continuar*, configura las principales opciones de tu wiki, los derechos de acceso y el tipo de licencia, entre otras opciones, y haz clic en *Continuar*.



Si quieres, puedes volver hacia atrás, para realizar algún cambio en la configuración previa a la instalación; si no quieres, haz clic en *Continuar* para comenzar la instalación de MediaWiki.

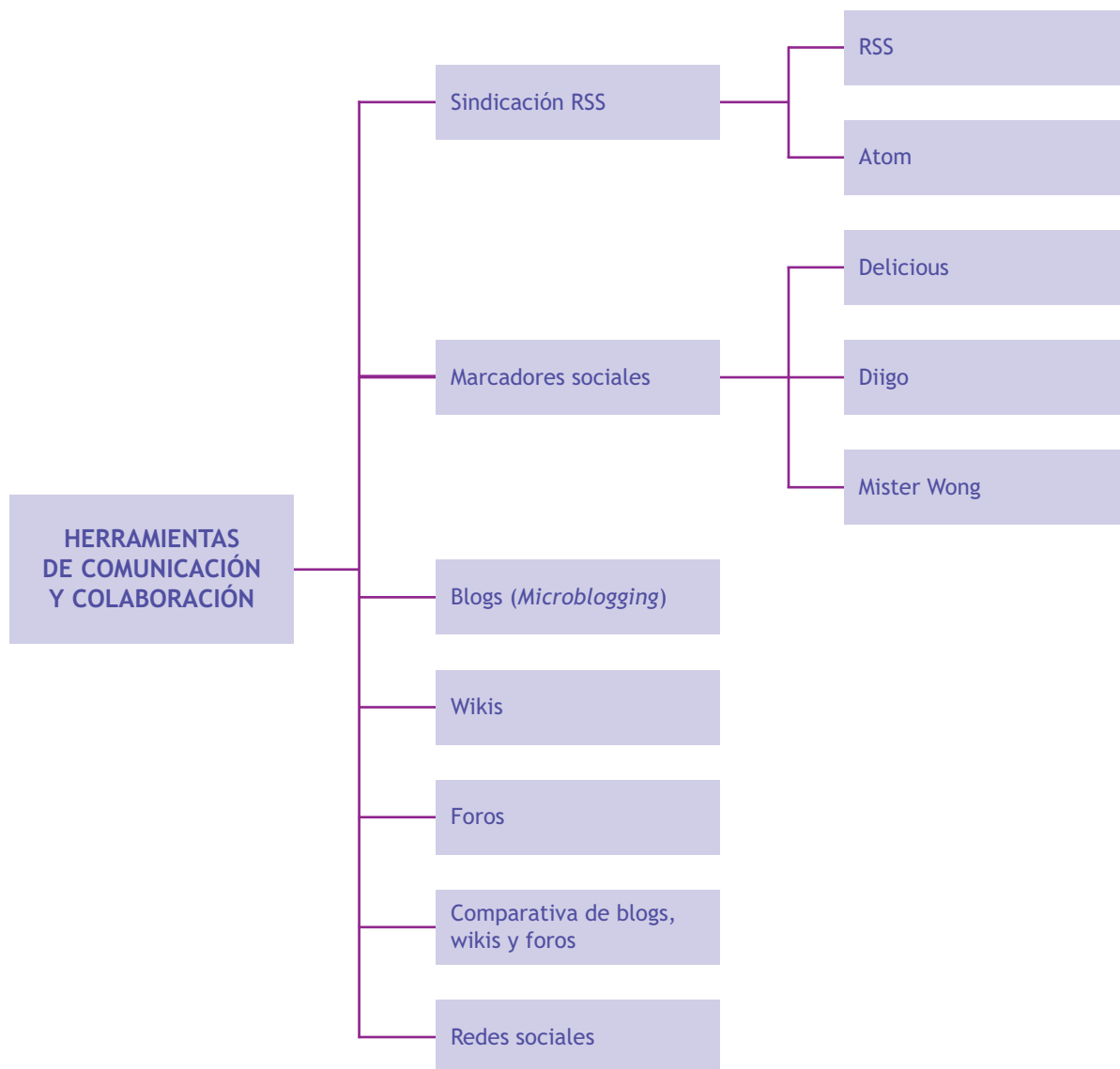


- Haz clic en *Continuar*, para finalizar la instalación.



- Descarga el fichero de configuración *LocalSettings.php*, que la instalación ha generado y ponlo en la base de la instalación del wiki (en el mismo directorio que index.php, en la carpeta `c:\xampp\htdocs\mediawiki`). Si la descarga no se inicia automáticamente, o la cancelas, puedes reiniciar la descarga haciendo clic en el enlace [Download LocalSettings.php](#). Lee la nota situada en la parte inferior: "si no lo hace ahora, este archivo de configuración generado no estará disponible para más adelante si usted sale de la instalación sin descargarlo".
- Ahora ya puedes entrar en tu wiki accediendo a la dirección <http://localhost/mediawiki/index.php>.

Ideas clave



Wikipedia alerta sobre un tipo de *malware* que introduce anuncios en sus páginas



“Nunca incluimos anuncios en Wikipedia”, asegura Philippe Beaudette, director de apoyo a la comunidad en Wikimedia Foundation. “Si estás viendo anuncios de propósito comercial o cualquier otro que no tenga que ver con nuestra habitual búsqueda de apoyo financiero, entonces tienes el navegador web infectado con *malware*”, asegura el directivo.

Un ejemplo de este tipo de *malware* es una extensión falsa de Google Chrome llamada “I want this”, según el propio Beaudette. Sin embargo, este tipo de añadidos maliciosos podrían también existir para Mozilla Firefox, Internet Explorer y otros navegadores, explica.

Este tipo de programas maliciosos son conocidos como *malware* fraudulento y puede atacar múltiples páginas web al mismo tiempo. Además de poder introducir anuncios en páginas web con extensiones falsas, también pueden provocar peticiones de búsqueda falsas para aportar a sus creadores ingresos, según Graham Cluley, consultor de tecnología de Sophos.

Detectar este tipo de acciones es más sencillo en Wikipedia que en otras webs porque este sitio no utiliza ningún tipo de anuncio comercial. “Distribuimos la suma del conocimiento humano a cualquier persona del planeta, libre de anuncios y para siempre”, asegura Beaudette.

Los costes operativos de Wikipedia se cubren con donaciones y la fundación recauda fondos todos los años a través del único *banner* que aparece en sus páginas.

Los usuarios que vean anuncios comerciales en Wikipedia deberían deshabilitar todos los *add-ons* de su navegador para determinar si son estos la fuente del problema, explica Beaudette. En cualquier caso, aunque haciendo esto desaparezcan los anuncios es posible que la solución no sea permanente y no acabe con todos los problemas. Este *malware* podría tener otros componentes internos al sistema que podrían volver a afectar al navegador. La única solución definitiva es utilizar un antivirus actualizado.

16/05/2012 PC WORLD PROFESIONAL

Actividades

1. Accede a la Wikipedia con varios navegadores y observa la existencia de algún anuncio comercial.
2. Busca información en Internet sobre cómo detectar y proteger los navegadores contra el *malware*.

Servicios de gestión de archivos en la nube

SUMARIO

- Gestión de archivos en la nube
- Servicios de almacenamiento
- Gestión de música, fotos y vídeos
- Integración de servicios en la nube

OBJETIVOS

- Conocer la utilidad de un servicio de gestión de archivos web.
- Describir diferentes aplicaciones de gestión de archivos web.
- Instalar y adaptar herramientas de gestión de archivos web.
- Crear cuentas de usuario en las aplicaciones de gestión de archivos web.
- Creación y gestión de recursos compartidos.
- Aplicar criterios de acceso sobre los recursos compartidos comprobando la seguridad del gestor de archivos.

1 >> Gestión de archivos en la nube

Hay muchos servicios web que ofrecen almacenar en Internet ficheros de diversos tipos, como fotografías, canciones, vídeos, documentos ofimáticos, etc. De esta forma, la nube se está convirtiendo en una de las principales alternativas para que las empresas almacenen su información.

Este sistema permite el acceso a dichos ficheros desde donde nos encontremos, es decir, lo mismo da usar el ordenador personal que cualquier otro dispositivo, siempre que se disponga de una conexión a Internet. A través de estos servicios web de gestión de archivos en la nube se puede acceder a la información y a los documentos que se almacenaron previamente; todo esto de una manera segura e independiente de la plataforma o sistema operativo que se utilice.

Actualmente, existen muchos servicios que ofrecen almacenamiento de información en la nube, que incluyen variedad de prestaciones técnicas con costes asumibles, a veces incluso gratuitos; algunos ejemplos son: Picasa, Dropbox, Google Drive, SkyDrive, Flickr, YouTube, etc.



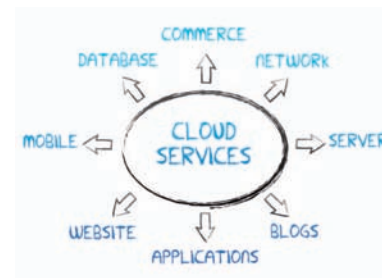
La gestión de documentos en la nube permite a las empresas:

- Almacenar y clasificar los archivos de información que la empresa necesita para su proceso de negocio, utilizando recursos ajenos a ella, lo que puede redundar en un ahorro de costes.
- Compartir los archivos con seguridad, confidencialidad e integridad. Solo accederán los usuarios que tengan las credenciales pertinentes y el proceso utiliza cifrado seguro en las comunicaciones, garantizando que la información no pueda ser alterada por intrusos.
- Disponer de la información de la empresa desde cualquier lugar y dispositivo que tenga acceso a Internet, favoreciendo la flexibilidad y la colaboración de los equipos de trabajo.
- Controlar las diversas versiones de sus documentos, para facilitar la asignación y seguimiento de proyectos y tareas.

El avance en las tecnologías de *cloud computing* ha cambiado la forma en la que los proveedores de tecnología atraen a los clientes, tanto si se trata de empresas como de usuarios individuales. Se prevé que este modelo continuará en crecimiento al menos durante los próximos 15 años. Aparecerá una nueva generación de aplicaciones de negocio, sobre todo en las pequeñas y medianas empresas, que actualmente se llevan la mayoría de la carga de computación en la nube.

Cloud computing

La **computación en la nube** es una propuesta tecnológica que facilita la prestación de los servicios ofrecidos por un sistema informático a través de "la nube", Internet.



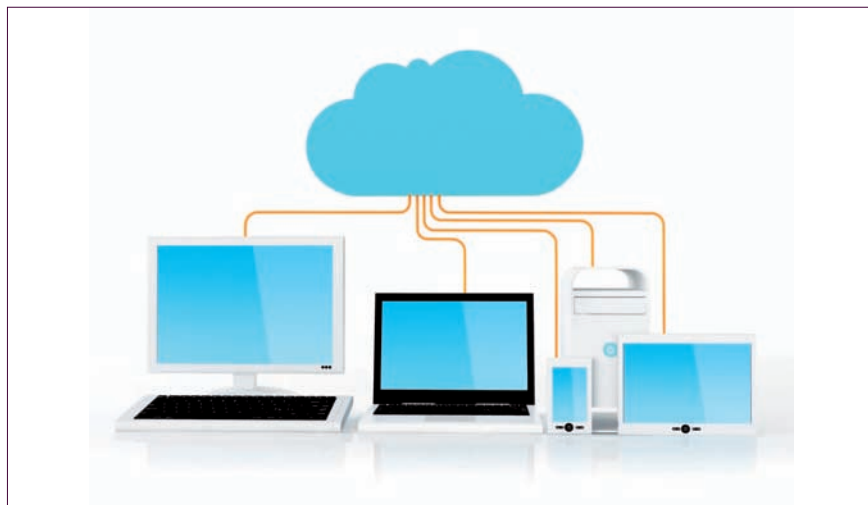
Los **servicios de gestión de archivos en la nube** permiten tener un espacio de almacenamiento (disco virtual) en Internet.

Evidentemente, no todo es positivo cuando se utiliza un sistema de almacenamiento de información en la nube; también hay ciertas desventajas que, en algunas circunstancias, pueden hacer replantearnos este método de almacenamiento de archivos como sistema complementario a los tradicionales. Por eso, siempre conviene leer los términos del servicio que nos ofrecen los proveedores para que no haya sorpresas.

Algunas **desventajas** de la gestión de archivos en la nube son:

- La dependencia de tener una conexión a Internet, que es imprescindible y necesaria para poder acceder a la información almacenada en la nube, así como la demora de la comunicación.
- La necesidad de adaptarse a la interfaz que proporciona la aplicación de almacenamiento en la nube, lo que limita las posibilidades de personalizar la gestión de los ficheros exclusivamente a las características y condiciones que ofrezca el servicio web, así como a sus posibles modificaciones.
- La política sobre derechos de los contenidos, que puede otorgar al proveedor plena potestad para hacer lo que quiera con el contenido subido a la nube a través de sus servicios.
- El hecho de que los datos “sensibles” de nuestro negocio no residan en las instalaciones de nuestra empresa, lo que podría generar un contexto de posible vulnerabilidad facilitando la sustracción o robo de información.
- La dependencia de los proveedores de servicios. A medida que más usuarios empiecen a compartir la infraestructura de la nube, la sobrecarga en los servidores de los proveedores aumentará; entonces, si la empresa no posee un esquema de crecimiento óptimo, se puede producir una degradación en el servicio, e incluso averías o problemas de acceso.

Para registrarse y poder acceder a estos servicios web de almacenamiento de información en la nube, se suelen exigir determinadas características a los usuarios. Esto se hace para proteger a la población infantil y hacer frente a los problemas de seguridad derivados del almacenamiento de información personal, fotografías o cualquier otro documento de carácter personal, así como cualquier otra información susceptible de ser protegida, por tener derechos de autor, derechos de explotación comercial, etc.



2 >> Servicios de almacenamiento en la nube

Existen diversos servicios de almacenamiento web especializados según el uso que se quiera dar a la información alojada en la nube.

Unos son muy buenos a la hora de sincronizar las carpetas del escritorio, otros en organizar y compartir fotos, otros para gestionar vídeos, otros en hacer *streaming* de colecciones de música, otros en la gestión de versiones de documentos, etc.

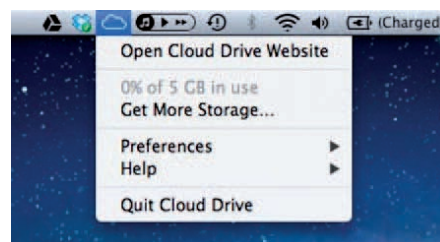
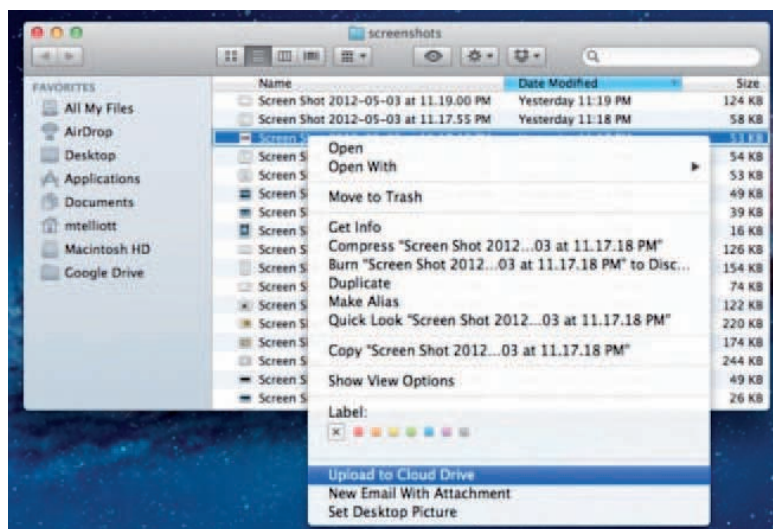
2.1 > Sincronización de ficheros en la nube

Estos servicios ofrecen varios gigabytes de almacenamiento de información y la sincronización de carpetas en la nube, algunos de ellos de manera integrada en el escritorio; entre ellos podemos destacar: Dropbox, Google Drive, Ubuntu One, SkyDrive, SugarSync, Amazon Cloud Drive, iCloud...

Ejemplos

Almacenar y sincronizar carpetas con Amazon Cloud Drive

Lo primero que debes hacer, es descargar e instalar la aplicación Amazon Cloud Drive, disponible para Windows y Mac. Solo podrás acceder a tus archivos a través del navegador, pero podrás subir y sincronizar archivos y carpetas de manera fácil; por ejemplo, en ordenadores Windows, seleccionando Amazon Cloud Drive desde la opción de la carpeta *Enviar a*, y en ordenadores Mac, eligiendo la opción *Subir* del menú Upload to Cloud Drive.



En Cloud Drive también es posible subir los archivos simplemente arrastrando un archivo, una carpeta, o varios archivos o carpetas, al icono con forma de nube que representa Cloud Drive de Amazon, ubicado en la esquina inferior derecha en Windows o en la barra de menú en la parte superior de la pantalla en Mac, como se puede observar en la figura de arriba. Así, los archivos y carpetas se agregan a la carpeta *Uploads*, la cual se crea la primera vez que se suben archivos. Amazon Cloud Drive se inicia con cuatro carpetas predeterminadas: *Documentos*, *Música*, *Imágenes* y *Vídeos*. Al subir archivos, también se puede hacer una pausa o incluso cancelar un proceso de carga, acciones accesibles a través del icono de la nube.

Casos prácticos

1

Creación de una cuenta, instalación de la aplicación y gestión de Dropbox

•• Crea una nueva cuenta en Dropbox, y descarga e instala la aplicación para escritorio.

Solución •• Dropbox es un software que enlaza todos tus dispositivos informáticos mediante una carpeta. De manera muy sencilla podrás almacenar copias de tus archivos y sincronizar estos archivos en el ordenador, el portátil, el smartphone... y la nube.

Creación de una cuenta en Dropbox

Para crear una cuenta y conseguir tu espacio de disco virtual en Dropbox, accede a su web www.dropbox.com.





Haz clic en *Iniciar sesión* y después en *Crear una cuenta*. Introduce tus datos y pulsa en *Crear una cuenta*.

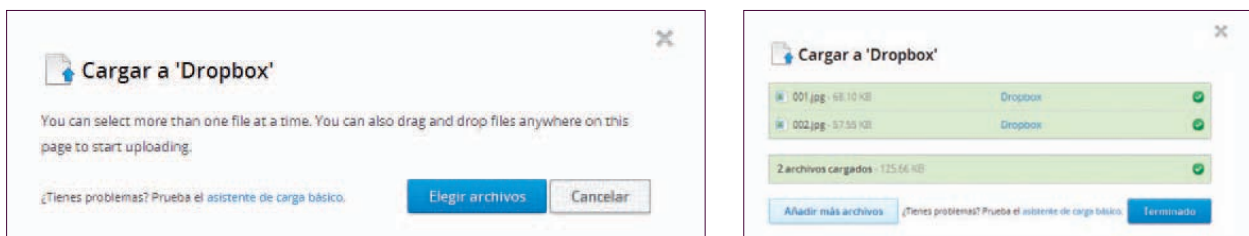
Dropbox te pide confirmación para descargar la aplicación de escritorio adecuada a tu sistema operativo.

Haz clic en *Guardar archivo* para guardar la aplicación de Dropbox que posteriormente instalaremos, en este caso en Windows. Después hazlo también en Linux, descargando la aplicación desde www.dropbox.com/install?os=lnx.

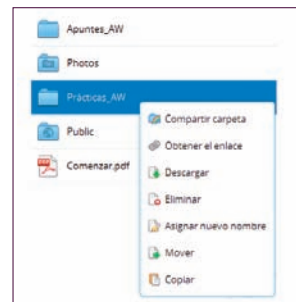
Ahora ya tienes disponible tu espacio de almacenamiento a través del navegador; pulsando en el icono de Dropbox, verás las carpetas *Photos* y *Public* generadas automáticamente. Para organizar la información que vayas subiendo, crearás tu propia estructura de directorios y así podrás clasificar la información en carpetas ordenadas.

A través del icono , crea dos nuevas carpetas, *Apuntes_AW* y *Prácticas_AW*.

Ahora puedes comenzar a subir ficheros a tu cuenta de Dropbox en la nube, a través del navegador. Haz clic en el icono de cargar  y después en *Elegir archivos*; se abrirá una ventana para que selecciones los ficheros que quieras cargar. Intenta elegir al menos dos archivos (puedes seleccionarlos a la vez) utilizando la tecla <Ctrl> o el ratón.

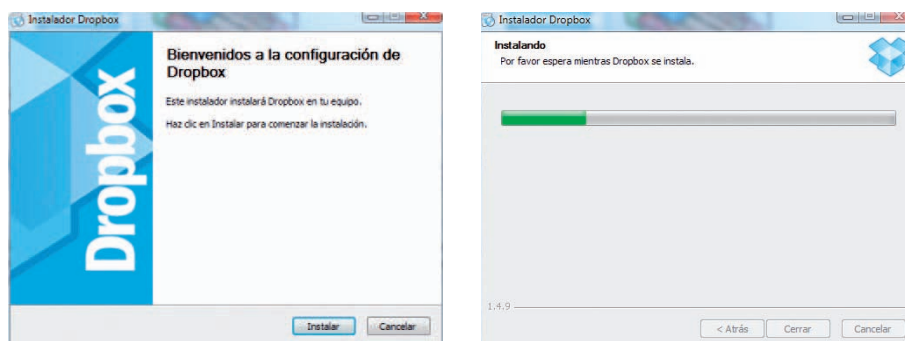


Ten en cuenta que los archivos se almacenarán en la carpeta donde te encuentres trabajando en el momento de subir los archivos. Por lo tanto, antes de dar al icono de cargar archivos, debes acceder a la carpeta donde quieras ubicar los archivos; en cualquier caso, con posterioridad a la carga, siempre podrás cambiar los archivos de carpeta, bien pinchando y arrastrando con el ratón, o bien a través del menú que se ofrece al hacer clic en el botón secundario sobre el documento que quieras mover.



Instalación de la aplicación cliente de Dropbox

Abre el instalador de Dropbox desde la ventana de descargas de tu navegador, haciendo doble clic en el archivo que descargaste. En Windows y en Linux, sigue las indicaciones para instalar y configurar Dropbox en tu equipo. Pronto estarás listo para comenzar a subir y sincronizar tu información con la nube, todo ello desde la propia aplicación cliente de Dropbox que acabas de instalar.



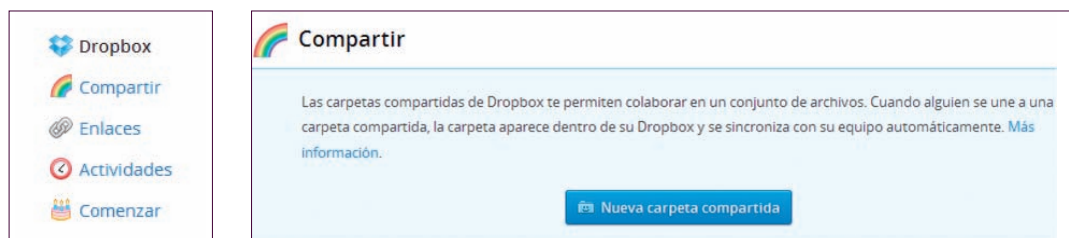
Gestión básica de Dropbox

Dropbox permite fácilmente sincronizar cualquier modificación en los archivos, entre diferentes dispositivos. Cualquier modificación que se produzca en la carpeta que Dropbox ha creado, durante la instalación de la aplicación en el sistema operativo, se sincroniza, pudiendo configurar esta carpeta y ubicarla en el lugar que más te interese. Así, una vez que hayas finalizado la instalación podrás subir un archivo a través de esta carpeta.

Bastará con copiar en ella los archivos que quieras subir. Mientras el fichero se está subiendo o sincronizando, aparecerá un icono azul 🔄 y cuando el archivo o la carpeta haya terminado de sincronizarse aparecerá un icono verde ✅.

Sube dos ficheros a la nube de Dropbox copiándolos en la carpeta creada en el sistema y luego comprueba, a través del sitio web de Dropbox en tu navegador, que puedes acceder a los ficheros subidos desde cualquier otro ordenador del aula, todo ello de manera segura usando tu nombre de usuario y contraseña.

Entra en la opción *Compartir* del menú principal de la izquierda para compartir información.



Pulsa en *Nueva carpeta compartida*, por ser la primera vez que quieres compartir una carpeta. Dropbox pone en marcha un protocolo de seguridad, que, como podrás observar, consiste en enviarte un correo que debes confirmar.

Haz clic en *Send email*, y obtendrás un mensaje de verificación; puedes hacer clic en *Terminado*.

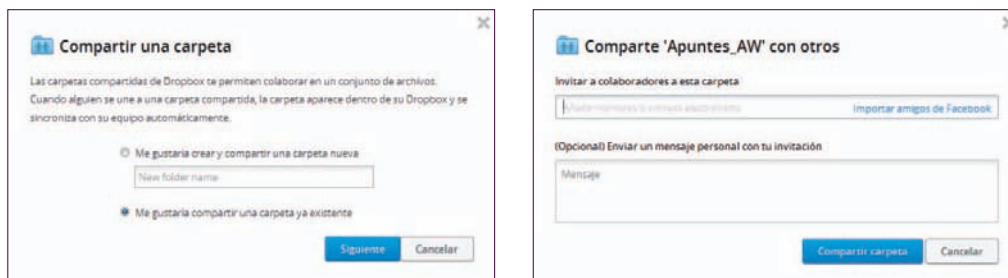
Ahora revisa tu correo electrónico y localiza el correo que Dropbox te ha enviado; dentro del correo recibido, pulsa en el enlace que dice, [Verify your email address](#), y solo entonces podrás compartir tus carpetas de Dropbox.

A continuación, pulsa de nuevo en *Compartir* y en *Nueva carpeta compartida*.

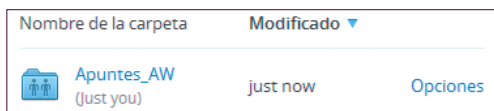
Para colaborar junto a tu compañero, comparte la carpeta que creaste anteriormente, la denominada *Apuntes_AW*.

Para ello selecciona la segunda opción, como aparece en la imagen de abajo y haz clic en *Siguiente*. En la siguiente pantalla te aparecen las carpetas existentes, selecciona la carpeta *Apuntes_AW* y haz clic en *Siguiente*.

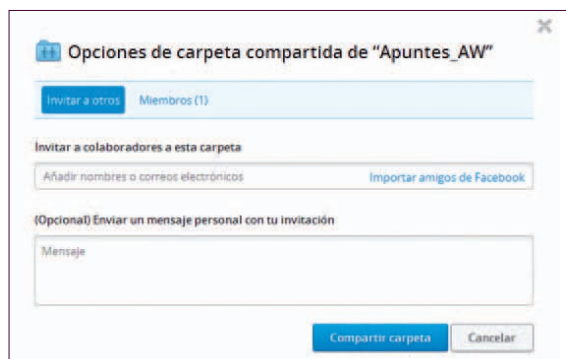
Después, invita a colaborar en esta carpeta a tu compañero; para ello, indica su cuenta de correo y haz clic en *Compartir carpeta*.



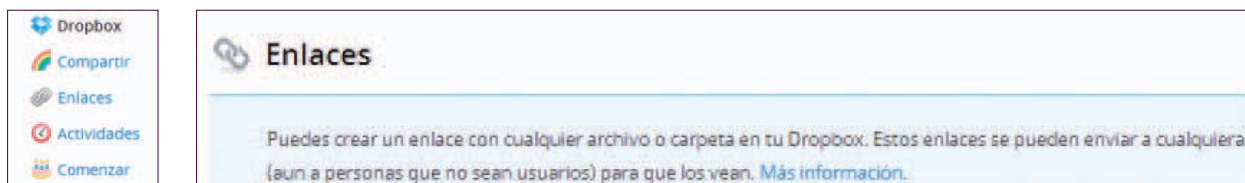
Ahora comprueba que tu carpeta *Apuntes_AW* ha sido correctamente compartida con tu compañero.



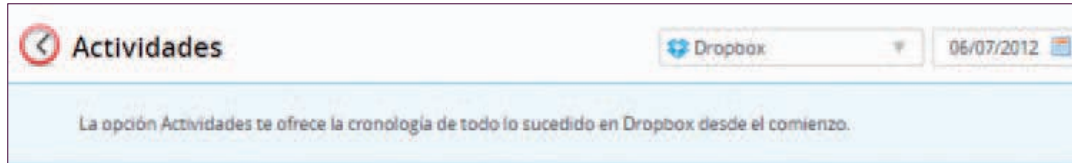
Si haces clic en *Opciones* observarás cómo se puede reconfigurar la carpeta compartida, invitando a otros compañeros (*Miembros*), o eliminando invitaciones, abandonándola o incluso dejándola de compartir.



Entra y explora la opción *Enlaces*. Aquí podrás crear enlaces a archivos y carpetas en tu Dropbox. De esta forma, también podrás compartir tus fotos, documentos y vídeos con amigos y colegas, aunque ellos no tengan cuenta en Dropbox. Podrás también eliminar los enlaces que hayas creado.



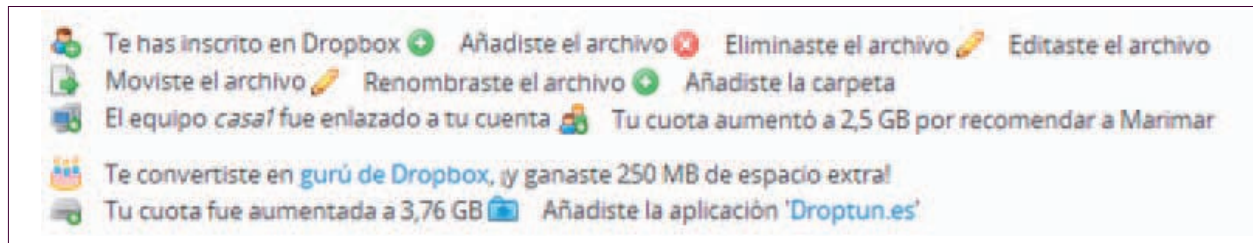
La opción *Actividades* del menú principal es como el historial de los documentos y las acciones realizadas en tu cuenta de Dropbox. Podrás recuperar versiones anteriores de los documentos.



Se registran todas las actividades realizadas en tu cuenta desde que te inscribes. Se registra actividad si se añade, elimina, edita, mueve o renombra un archivo en tu espacio de disco virtual de Dropbox, y también cuando se añade una carpeta.

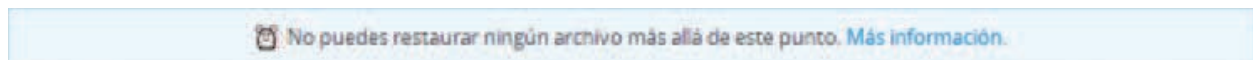
Esta cronología de actividades ofrece **seguridad**. Si instalas la aplicación de escritorio de Dropbox en un equipo y enlazas a ella tu cuenta de usuario, podrás comprobar si alguien usa tu cuenta sin permiso en otra máquina, y también si se realizan actividades que afecten al contenido de tus documentos, a través de la Web. Además, en la opción actividades se registran tus aumentos de cuota por recomendaciones, así como las aplicaciones que añades.

Accede a tu historial de actividades en Dropbox, busca y localiza algunos de estos mensajes:



Cuando tengas la necesidad de recuperar alguna versión anterior de tus ficheros subidos a Dropbox, puedes hacerlo a través de este registro de actividades, pero debes tener en cuenta que no podrás acceder a todas las versiones anteriores; está limitado a un tiempo máximo anterior, establecido por Dropbox.

Intenta localizar el siguiente mensaje; sólo lo visualizarás cuando tengas actividades realizadas hace más de 30 días.



Dropbox permite contratar cuentas de mayor capacidad y otras funcionalidades; por ejemplo, ofrece una función añadida, denominada *Packrat*, que permite guardar un histórico de todas las versiones anteriores de los ficheros.

Actividades propuestas

1. Entra en Ubuntu One (<https://one.ubuntu.com/>) regístrate, configúralo en tu ordenador y compáralo con Dropbox.



Streaming

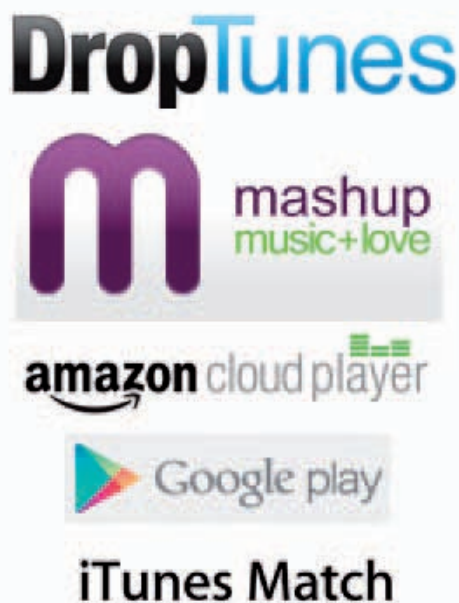
El *streaming* permite distribuir archivos multimedia, normalmente audio o vídeo, a través de Internet o de una red local. Así los usuarios de dicha red pueden escuchar una canción o visualizar un vídeo casi al mismo tiempo que se produce la descarga del archivo multimedia. La palabra *streaming* expresa algo que no para de fluir; en este caso, los paquetes de datos que contienen el archivo multimedia. Mediante un sistema de almacenamiento intermedio en el ordenador del usuario, los datos se van almacenando e inmediatamente se van reproduciendo, sin necesidad de descargar el archivo completo para poder comenzar a reproducirlos.

2.2 > Almacena y escucha música en la nube

Existen diversas posibilidades para poder escuchar la música que almacenes en la nube; hay servicios dedicados a la música *online* y otros que permiten complementar a los servicios de almacenamiento, para que los usuarios puedan escuchar en *streaming* los archivos de música que previamente almacenaron en ellos.

Algunas posibilidades son:

- **DropTunes**, una extensión web de Dropbox que permite a los usuarios escuchar la música almacenada en sus cuentas a través de un reproductor web, sin necesidad de descargar los contenidos; solo necesitan acceder a DropTunes con sus cuentas de Dropbox, y el sistema automáticamente les mostrará la lista de canciones existentes en su cuenta. Esta es una de las mejores opciones gratuitas para hacer *streaming* de tu música.
- **Mashup**, la empresa Mougg ofrece a los usuarios un lugar para almacenar (hasta 2 GB de forma gratuita), mostrar y compartir sus colecciones de música, vídeos y conciertos favoritos en línea, a través de la Web y a través de dispositivos móviles con sistemas operativos de Apple, Android, Windows Phone y BlackBerry. También dispone de la tienda en línea donde los usuarios pueden comprar canciones.
- **Cloud Player** permite reproducir en cualquier ordenador, tablet o smartphone, ya sea Mac, Android o Windows, los contenidos musicales almacenados previamente en Amazon Cloud Drive (el disco duro virtual en el que podemos almacenar música, vídeos, fotos y otros documentos que permanecen accesibles desde la cuenta del usuario a través de cualquier navegador web). Los contenidos musicales puedes subirlos desde tu propio ordenador o bien comprarlos a través de la tienda *online* de MP3.
- **Play** es un servicio con el que podrás consumir contenidos de todo tipo, música, películas, e-books, aplicaciones o juegos; es una tienda *online* y al mismo tiempo un sistema de almacenamiento. Este servicio en constante desarrollo unificó varias tiendas, Google Music, eBookstore y el antiguo Android Market, con el fin de alojar todo en el mismo lugar. Play permite sincronizar archivos entre diferentes dispositivos; puedes comprar un contenido en tu teléfono y luego verlo desde donde quieras, en cualquier equipo, usando tu cuenta de Google en su interfaz web y también a través de la aplicación de música de Google para Android.
- **iTunes Match** permite que la música que compras en el iTunes Store aparezca de forma automática en los dispositivos de Apple, iPhone, iPad, iPod touch, Mac, PC o Apple TV, aunque la música haya sido comprada fuera de iTunes. Con esta prestación, integrada en el ordenador y en la aplicación de música de los dispositivos iOS, puedes almacenar una colección de música en iCloud, incluso las canciones importadas desde un CD.



Casos prácticos

2

Uso de la aplicación DropTunes

•• Accede a la aplicación web de DropTunes y escucha música almacenada previamente en Dropbox.

Solución ••

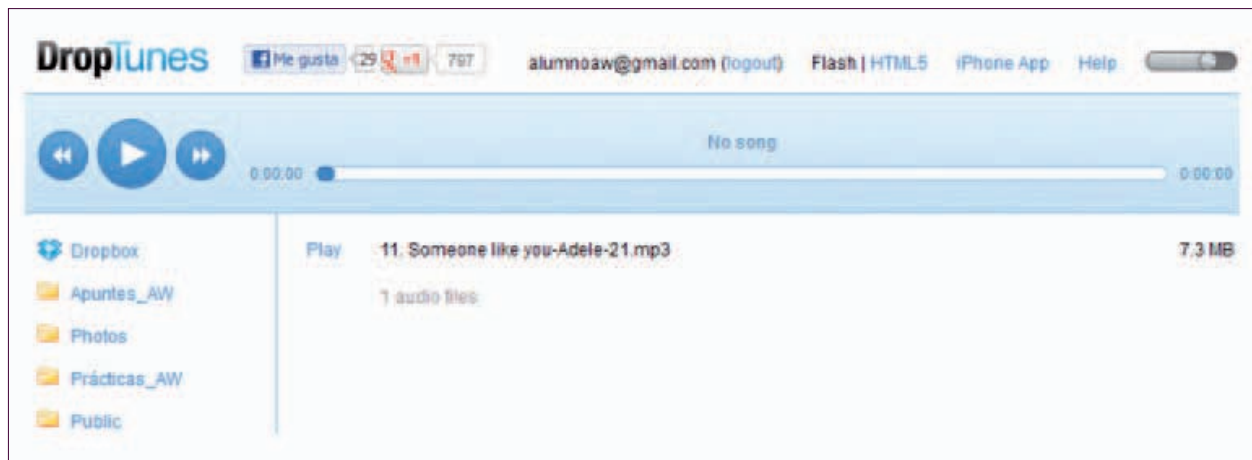
Entra en la página web de DropTunes; para ello, accede con el navegador a la dirección www.droptun.es. Introduce tu nombre de usuario y contraseña, elegidos en el caso práctico anterior al crear tu cuenta de Dropbox. Una vez que has validado tus datos, el entorno de DropTunes te permitirá acceder a todos los ficheros de audio que tengas almacenados en la nube de Dropbox.



Si previamente no has subido ficheros de música a tu cuenta de Dropbox, sube ahora una canción a tu espacio de almacenamiento.

Cuando tengas almacenado algún fichero de audio intenta localizarlo, desde el sitio de DropTunes, accediendo a la carpeta donde esté almacenado.

Una vez localizada la carpeta donde guardaste la música, elige la canción que quieres reproducir y pulsa el enlace *Play* que aparece en color azul al lado de la canción.



Actividades propuestas

- 2•• Busca en Internet información acerca de los servicios de almacenamiento en la nube y realiza un listado de los que encuentres, resumiendo sus características principales, capacidad de almacenamiento que ofrecen, etc.
- 3•• Resume las principales ventajas del *cloud computing* respecto al almacenamiento de información.
- 4•• Busca la página web de Amazon Cloud Drive, y descarga e instala la aplicación.
- 5•• Realiza los pasos necesarios para subir y sincronizar archivos en la nube a través de Cloud Drive.

2.3 > Gestión de fotos en la nube

Hoy en día, se obtienen enormes cantidades de fotos utilizando cámaras digitales y smartphones. Los sistemas de almacenamiento en la nube permiten que tengamos copias de seguridad de dichas fotos. Por ejemplo, si dispones de un iPhone, puedes utilizar **iCloud** para tener allí copias de tus fotos y sincronizarlas con el resto de dispositivos.

Similares prestaciones se ofrece a los usuarios de Windows Phone a través de las funciones disponibles en el sistema de almacenamiento en la nube **SkyDrive**.

También **Google+** facilita a los dispositivos Android un sistema de envío automático de fotos a su red social, permitiendo su almacenamiento y su posterior publicación.

El sistema de almacenamiento **Dropbox** dispone igualmente de una función que permite la carga automática de fotos a través de su aplicación disponible para Android. Otro ejemplo es **Ubuntu One**, con el que podemos realizar cargas automáticas de fotos si usamos Android o iOS.

Existen muchas alternativas a la hora de subir imágenes y álbumes completos a Internet. Picasa y Flickr siguen siendo las dos opciones preferidas por los usuarios para compartir fotografías en la Red, tanto si se trata de aficionados como en el caso de profesionales del sector de la fotografía y el diseño.

- **Picasa**: es el servicio de almacenamiento de fotos de Google, servicio de gran potencial que pronto cambiará de nombre pasando a denominarse simplemente Google Fotos, integrado en Google+. El concepto del álbum web de Picasa se basa en colocar en una misma carpeta muchas fotos, que pueden mostrarse como diapositivas, descargarse o compartirse, por correo o como un enlace accesible desde Internet. Mediante la aplicación para móviles de Google+, se puede subir automáticamente a Picasa cualquier foto recién hecha, evitando desperdigar las fotos en diferentes plataformas; de esta forma podemos compartir las fotos que permanecen guardadas en Picasa.
- **Flickr**: es quizás el servicio de alojamiento de imágenes más conocido. Permite compartir todo tipo de fotos y vídeos de manera pública o privada. En Flickr, el concepto de álbum adquiere un sentido más social y participativo, ya que una de las grandes bazas de este servicio es su potente comunidad de usuarios, aficionados y profesionales de la fotografía, que comparten imágenes, comentan sobre ellas y recomiendan álbumes o usuarios. Posee una de las imágenes de marca más modernas y sofisticadas del mundo 2.0. Por ello, los diseñadores, web máster y fotógrafos profesionales, suelen tener una cuenta en Flickr.

Ambas plataformas ofrecen unas prestaciones similares. Tanto Picasa como Flickr cuentan con una herramienta para poder subir fotos que se puede instalar en el ordenador, además de poder hacerlo desde su página web. La versión de pago de Flickr incluye un completo módulo de estadísticas que permite conocer las fotos preferidas de los visitantes. En Picasa también se ofrece información sobre las visitas realizadas a una foto.

Servicios de fotografías en la nube



www.picasaweb.google.com



<http://www.flickr.com>

Casos prácticos

3

Gestión de imágenes en la nube con la aplicación web de Picasa

•• Accede a la aplicación web de Picasa y gestiona álbumes web con Picasa.

Solución ••

Accede con un navegador web a la dirección www.picasaweb.google.com. Introduce tu nombre de usuario y contraseña de Google; puedes utilizar la cuenta que creaste en la unidad didáctica anterior al crear un blog con Blogger.



Una vez validada tu cuenta, fíjate en el menú de opciones de Picasa, con el que gestionarás tus álbumes de fotos:



Crea tu primer álbum de fotos en Picasa y añádele fotos.

Para crear un álbum nuevo, o añadir fotos a uno ya existente, pulsa en el botón *Subir*. En la siguiente pantalla, indica un nombre para tu álbum de fotos, selecciona y añade fotos al álbum y, finalmente, para terminar pulsa en el botón *Aceptar*.

Pulsa en la pestaña *Mis fotos*, y verás todos tus álbumes que hay creados, en este caso solo uno.

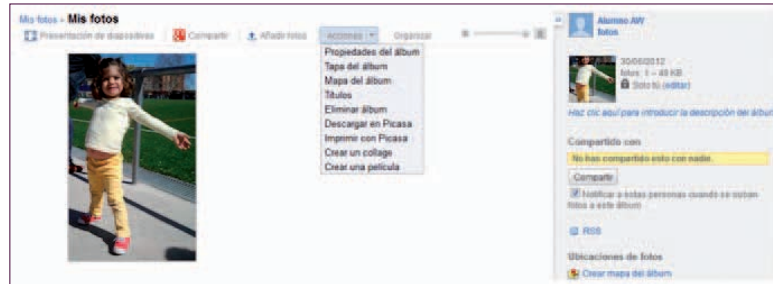
A continuación, pulsa en el enlace *Modificar visibilidad* para seleccionar quién quieres que tenga acceso a tu álbum de fotos.



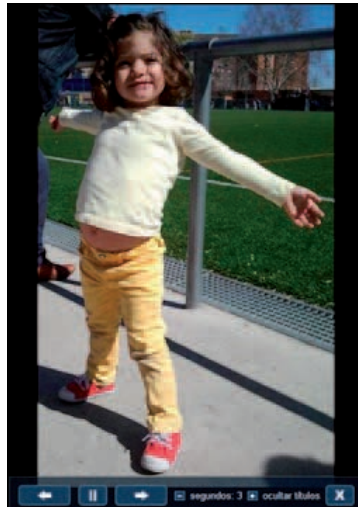
Luego, selecciona el último álbum que has creado, pulsando sobre él, y podrás visualizar todas las fotos que contiene.





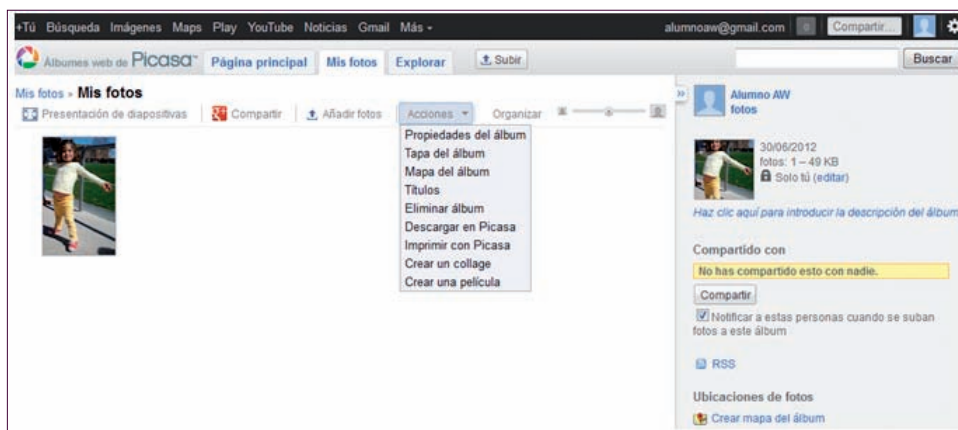
Accede a las opciones del menú *Acciones*, configura el álbum de fotos, propiedades, título, fecha, descripción, ubicación en el mapa. Selecciona la foto para la tapa del álbum. Añade títulos a cada una de las fotos.



Visualiza el álbum como una *Presentación de diapositivas* y compártelo con tu compañero.



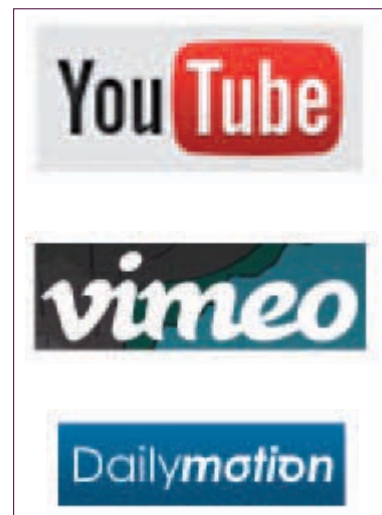
Con los dos iconos  , que aparecen a la derecha del usuario con el que has compartido el álbum, puedes permitir que tu compañero añada fotos a tu álbum, o bien dejar de compartirlo con él. Permite que tu compañero añada fotos a tu álbum y pídele que añada una foto a tu álbum.



2.4 > Gestión de vídeos en la nube

Hay varios servicios web que permiten el almacenamiento y gestión de vídeos en la nube; entre los más populares tenemos:

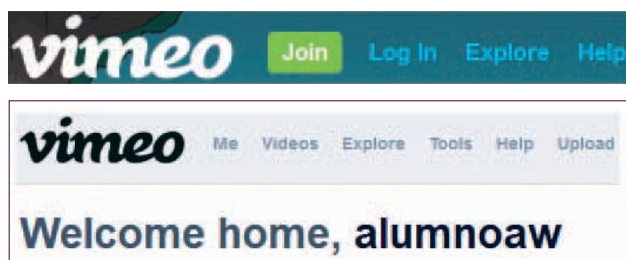
- **YouTube**, de Google, es probablemente la mayor comunidad de vídeos existente. Tiene numerosas funcionalidades, tales como buscar vídeos, subir y compartir vídeos propios en HD con el resto del mundo, comentar y valorar vídeos ajenos y crear listas de reproducción de tus vídeos favoritos para verlos más tarde. Además, en YouTube se pueden alquilar películas completas para reproducirlas en tu ordenador o smartphone.
- **Vimeo** permite cargar vídeos de manera sencilla, admite muchos tipos de formatos de archivo y ofrece herramientas para que los vídeos se vean perfectamente, incluso en HD y 3D, controles de privacidad para que puedas decidir quién puede ver tus vídeos, exploración de vídeos de interés a través de categorías, grupos, canales y temas, así como una escuela sobre creación de vídeos.
- **Dailymotion** ofrece ilimitadas subidas de vídeos HD y 3D, seguir vídeos de gente que te gusta y compartir opiniones en foros. Es posible acceder a Dailymotion desde (casi) cualquier dispositivo, guardar vídeos en tu lista de favoritos y asistir a espectáculos en vivo.



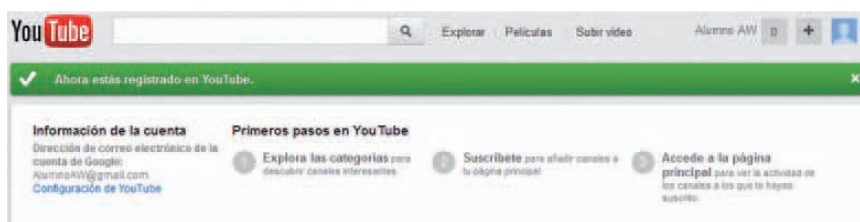
Ejemplos

Subir un vídeo en Vimeo y en YouTube

Para **subir un vídeo a Vimeo**, primero debes registrarte. Para ello accede a la página de Vimeo (<http://vimeo.com/>) pulsa en *Join*, y regístrate con una cuenta gratuita *Basic*. Una vez registrado, recibirás un correo electrónico de bienvenida para que verifiques tu dirección de correo; hazlo para poder subir vídeos. Haz clic en *Upload* y podrás cargar vídeos en Vimeo.

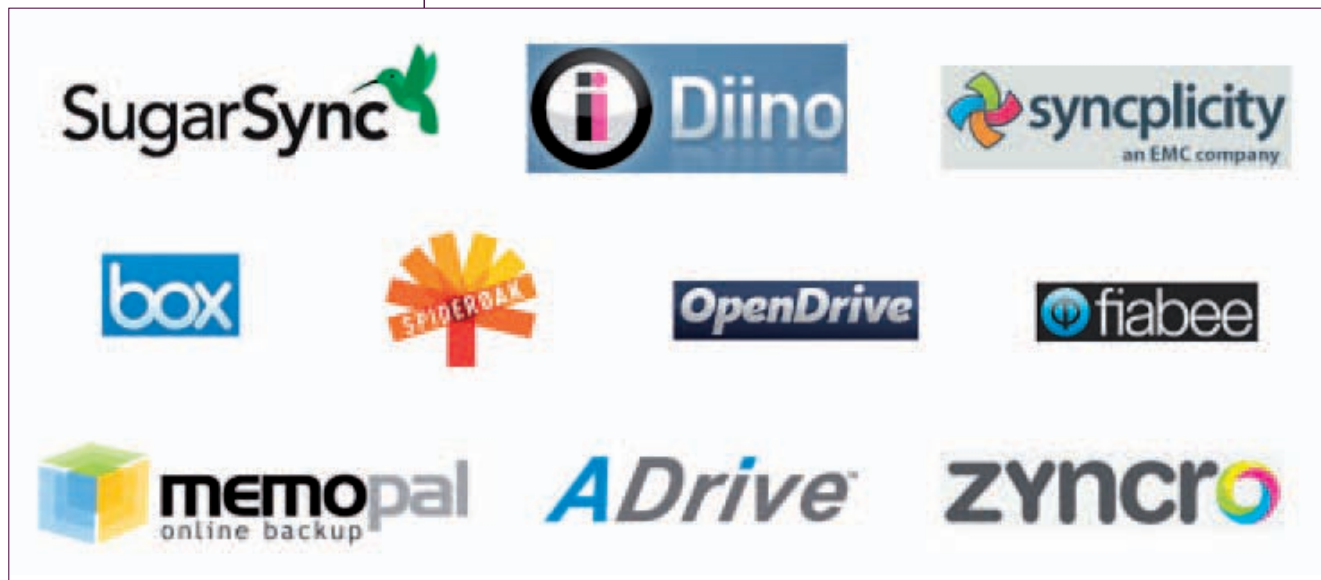


Para **subir vídeos a YouTube**, accede a la dirección (<http://www.youtube.com/>) e inicia sesión con tu cuenta de usuario de Google. Podrás subir vídeos elaborados por ti, explorar vídeos por categorías y alquilar los últimos estrenos de películas.



3 >> Integración de servicios en la nube

Existen otros muchos sitios que ofrecen almacenamiento virtual. Se diferencian en la capacidad de almacenamiento, en la cuota para cuentas de mayor capacidad y privilegios, en la capacidad de compartir la información, en la velocidad de acceso a la información, en ofrecer software en línea para editar documentos, en los sistemas de seguridad, así como en otras funcionalidades. En la figura se recogen solo algunos de los servicios de gestión de archivos disponibles en la nube.



WebDAV

Web-based Distributed Authoring and Versioning define un protocolo de un servicio que permite compartir, editar y manejar archivos personales que residen en un servidor remoto, vía HTTPS.

Para conectarse a través de WebDav es necesario utilizar un programa cliente específico. Existen varios en el mercado para Windows 7 y versiones anteriores; para Ubuntu y para MacOS hay programas clientes con interfaz gráfica. También existe algún programa cliente por consola, como la aplicación denominada *Cadaver*.

En cualquier caso, puede resultar complicado gestionar todos los archivos, cuando los tengamos repartidos por distintos servicios de almacenamiento.

Con las actualizaciones multiplataforma de servicios ya consagrados de almacenamiento *online*, como Dropbox entre otras muchas, y el *boom* por la llegada de aplicaciones populares, como Google Drive, conviene tener en cuenta la existencia de alguna aplicación que permite integrar estos servicios.

Así, **Otixo** es una aplicación web que puede resultarnos muy útil, dado que a veces es difícil recordar dónde hemos guardado ese trabajo tan importante que hay que entregar para el instituto, o al jefe, y nos pasamos un buen rato buscando entre las carpetas de Dropbox, SkyDrive, Box, etc. Este problema se soluciona con esta aplicación que permite mover o copiar archivos entre varios servicios de almacenamiento en la nube.



De esta forma, puedes gestionar todas tus cuentas de Dropbox, Box, SkyDrive, SugarSync, Picasa, etc, desde un mismo lugar, Otixo. Además, podrás agregar tus propios servidores FTP o WebDav, para tenerlo todo absolutamente controlado y poder gestionar accesos a la información desde diferentes tipos de aplicaciones clientes con diferentes protocolos.

Ejemplos

Integración de servicios de almacenamiento en la nube con Otixo

Es una herramienta que te ayuda a conectar tus aplicaciones favoritas en la nube, es decir, te permite acceder a todos los archivos que tienes en la Web, aunque estén distribuidos entre varios servicios en la nube (como puedan ser Dropbox, Google Drive, SkyDrive, y cualquier otro servicio), pero solo desde esta aplicación. Con Otixo puedes colaborar sin fronteras, conectar los equipos y proyectos de trabajo, mejorar tu organización y trabajar más rápido.



Para entrar en Otixo, accede a la dirección (<http://otixo.com/>), haz clic en *Sign Up for Free!* Después de rellenar tus datos, tienes que hacer clic en *Create Account* para crear tu cuenta en Otixo. Recibirás un mensaje que te pedirá consultar el correo electrónico indicado para confirmar tu registro en Otixo. Una vez que confirmes tu registro, podrás acceder a Otixo para empezar a utilizar el servicio. Este es el escritorio que encontrarás la primera vez que accedas al servicio.

