

### 1.1

**Internet** es una inmensa red de computadoras alrededor de todo el mundo conectadas entre sí. En cambio, **la web** (la World Wide Web) es una enorme colección de páginas que se asienta sobre la red de computadoras. Así que cuando navegas a través de tu teléfono u ordenador usas internet para acceder a la web.

### 1.2

Estos son los principales servicios de internet: World Wide Web, Correo electrónico, Grupos de noticias, Listas de distribución, Foros Web, Weblogs, blogs o bitácoras, Transferencias de archivos FTP, Intercambio de archivos P2P, Archie, Wikis, Sindicatos de contenidos, Chats o IRC...

El servicio más extendido es la Web o World Wide Web.

### 1.3

La World Wide Web es básicamente un conjunto de documentos electrónicos enlazados unos con otros, gracias al hipertexto. El primer detalle que tenemos que tener en cuenta, es como se localiza un documento. Esto se realiza mediante las URL (Localizadores Uniformes de Recursos) que son un subconjunto de las URIs (Identificadores Universales de Recursos). Las URIs que hagan referencia a documentos u objetos accediendo mediante protocolos existentes los denominaremos como URLs. En el curso cuando queramos indicar la dirección y la forma de acceso a un determinado recurso utilizaremos su URL.

Los URL contienen la forma de acceder a un recurso: el protocolo de acceso (http, ftp, news, etc.), el servidor en el que se encuentra el recurso (y el puerto) y el camino o ruta dónde se encuentra el recurso.

El protocolo de transferencia de hipertexto (*HTTP, HyperText Transfer Protocol*) es el protocolo que permite la comunicación entre servidores y clientes en la WWW. Básicamente consiste en una conexión del cliente con el servidor, un envío de petición del cliente al servidor, una respuesta por parte del servidor (un documento u objeto, o un error) y el cierre de la conexión.

HTML es un lenguaje de marcado componente básico de la Web con el que se estructuran el contenido de la Web.

#### 1.4

El Hipertexto es una tecnología que te permite enlazar una información con otros textos relacionados, pulsando con el mouse o el teclado en ciertas zonas sensibles y destacadas del texto, llamadas hipervínculos o hot spots.

La Hipermedia, por otra parte, surge de la fusión entre los conceptos 'hipertexto' y 'multimedia'. Es un conjunto de métodos o procedimientos que permiten crear, manipular, examinar, diseñar o componer contenidos que incluyan texto, video, audio e imágenes, que además permiten la posibilidad de interactuar con los usuarios.

#### 1.5

navegadores: es un software que permite el acceso a Internet, interpretando la información de archivos y sitios web para que éstos puedan ser leídos.

Algunos ejemplos de los más utilizados son: Google Chrome, Safari, Firefox, Opera, Eminent, Internet Explore.

La función principal de un navegador es solicitar al servidor los recursos web que elija el usuario y mostrarlos en una ventana. El recurso suele ser un documento HTML, pero también puede ser un archivo PDF, una imagen o un objeto de otro tipo. El usuario especifica la ubicación del recurso mediante el uso de una URI (siglas de Uniform Resource Identifier, identificador uniforme de recurso).

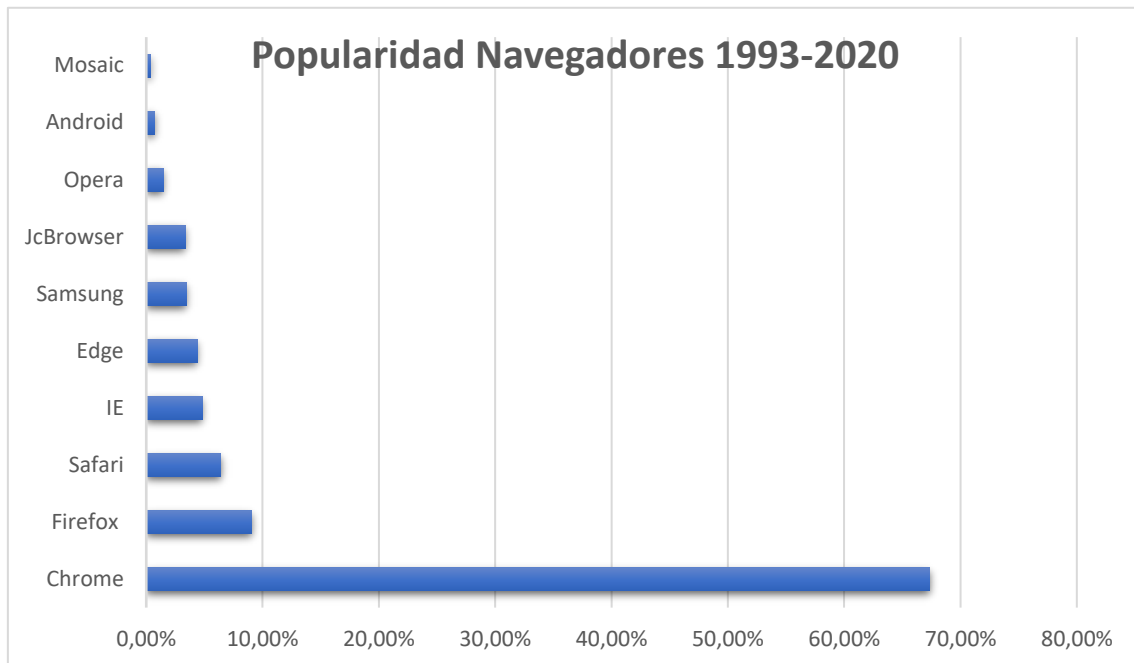
#### 1.6

Los navegadores web (en inglés, *web browsers*) son los programas que permite acceder a la web. Desde su aparición en 1991, los navegadores han evolucionado considerablemente, a la vez que la propia web.

Para muchos usuarios y durante muchos años, decir navegador web fue decir Internet Explorer de Microsoft. Esos tiempos ya pasaron y actualmente, decir navegador web es decir Google Chrome.

Antes de la aparición del primer navegador *Mosaic* las Web eran simplemente texto, luego apareció un Netscape, con la intención de ser un programa comercial, aunque con versiones gratuitas.

La tercera aparición fue la de Internet Explorer junto con el nacimiento de Windows 95. Tras el boom de Internet Explorer nació Opera, otro navegador de pago. Tras la salida de Opera, fue acusado de monopolio por incluir Internet Explorer gratis en Windows siendo el navegador más utilizado con un 92,5% de usuarios. En 2002 nace la primera beta de Firefox llamada Phoenix y luego Firebird. En 2003 Apple lanza Safari 1.0 y un año después sale la primera versión oficial de Firefox 1.0. El 2 de septiembre de 2008 Google lanza Chrome 1.0 y en 2012 adelanta a Internet Explorer y se convierte en el navegador más usado hasta el día de hoy. En 2015 Microsoft anuncia que Edge reemplazará a Internet Explorer quedando la gráfica de popularidad de este modo:



Dejo un video de las estadísticas de la guerra entre navegadores más usados de internet desde 1993 hasta 2020 y los cambios más importantes

<https://www.youtube.com/watch?v=1BV1HvUIUcQ&t=62s>

## 1.7

Un sitio web es el conjunto de archivos electrónicos y páginas web referentes a un tema en particular, que incluye una página inicial de bienvenida, generalmente denominada *home page*, con un nombre de dominio y dirección en Internet específicos. Un sitio web es un gran espacio documental organizado que la mayoría de las veces está típicamente dedicado a algún tema particular o propósito específico.

En resumen, una página web se aloja en un sitio web.

## 1.8

Como ya hemos comentado varias veces... HTML es un lenguaje de marcada que se utiliza para dar forma entre otras cosas a las páginas web. El lenguaje HTML es permisivo en su sintaxis, por lo que es posible escribir sus etiquetas y atributos de muchas formas diferentes. Esto es un aspecto positivo dependiendo del objetivo para el que se ha escrito la página web, pero en otras ocasiones el resultado final puede ser una página con un código HTML desordenado, difícil de mantener y manipular.

XHTML soluciona estos problemas añadiendo ciertas normas en la forma de escribir las etiquetas y atributos. La principal norma de XHTML que debes tener en cuenta es que todas las etiquetas deben tener su etiqueta de inicio y de fin.

La mayoría de etiquetas HTML encierran un contenido de texto entre la etiqueta de inicio y la etiqueta de fin. En esos casos HTML y XHTML se comportan igual. Sin embargo, algunas etiquetas especiales llamadas "etiquetas vacías" o "etiquetas sin contenido" no necesitan

encerrar ningún texto. Por ejemplo, la etiqueta `<br>`

Pues bien, XHTML obliga a cerrar siempre todas las etiquetas abiertas y por tanto dicho etiqueta se escribiría así: `<br></br>`

Una segunda norma es que antes de acceder al valor de un atributo, se deben eliminar todos los espacios en blanco que se encuentran antes y después del valor. Es decir, se eliminan todos salvo un único espacio en blanco utilizado para separar las diferentes palabras.

La tercera norma hace referencia a la etiqueta `<script>`. En esta etiqueta, el código JavaScript debe encerrarse entre unas etiquetas especiales (`<![CDATA[ y ]]>`) para que el navegador no interprete de forma errónea caracteres como `&` y `<`

Aunque muchas de las etiquetas de marcado permanecen sin cambios en respecto a HTML 4.01, varias de ellas se han simplificado, por lo que es mucho más fácil y rápido escribir código. Se basa en un estándar completamente nuevo, y su análisis sintáctico también es completamente diferente.

A diferencia de las versiones anteriores de HTML, que permitían crear sitios principalmente estáticos que necesitaban ser aderezados con CSS y JavaScript, HTML5 es mucho más dinámico e incluye elementos multimedia. Soporta de forma nativa el vídeo y el audio, e incluso puedes crear juegos o animaciones con él.

En otras palabras, HTML5 es totalmente capaz de hacer cosas que antes había que hacer con herramientas antiguas como JavaScript, Flash o Silverlight. Esto significa que tu sitio es más seguro y está menos abierto a los exploits de los atacantes que intentan entrar.

Y aunque CSS y JavaScript siguen siendo definitivamente necesarios para crear un sitio web completo, ya no es necesario depender de ellos para hacer nada dinámico en absoluto.

HTML5 ya no es sólo un constructor de sitios web. Es todo un constructor de aplicaciones por sí mismo.

En lugar de que la próxima actualización salga como «HTML6», HTML5 se actualiza continuamente como un estándar vivo que evoluciona según las necesidades de la web. Es la última y mejor versión de esta tecnología de larga duración, y va a estar aquí durante mucho tiempo.

## 1.10

Algunos tipos de herramientas que se utilizan a la hora de crear páginas web son: editores de código, entornos de desarrollo, editores de imagen, Cliente FTP, servidor local...

Ejemplos de herramientas que nos ayudan a la hora de crear páginas web: Adobe Color CC, Google Fonts, WhatFont, MockFlow, Canva, SublimeText, VScode, Bloom, Wordpress...

### 1.11

Según la W3C:

-Un **elemento en HTML** se refiere al conjunto de la etiqueta, su atributo (si tuviese), y el contenido que lo conforma. En el se encuentran:

-**Las etiquetas** son la estructura básica de HTML, siempre van entre signos de mayor que y menor que < >

-**Los atributos** van dentro de las etiquetas de inicio de un elemento y nos indica las propiedades que pueden llevar asociadas los elementos.

No todas las etiquetas HTML son de la misma estructura. Si bien la mayoría de los elementos requieren una etiqueta de apertura, una etiqueta de cierre y contenido, algunos elementos, conocidos como elementos vacíos, solo requieren una etiqueta de apertura ya que ellos mismos no contienen ningún elemento. Un elemento vacío no puede tener ningún contenido, pero puede tener atributos. Los elementos vacíos se cierran automáticamente, por lo que no deben tener una etiqueta de cierre.

```
<area shape="" coords="" href="" alt="">
<base href="">
<br>
<col>
<embed src="" type="">
<hr>
<img src="" alt="">
<input type="">
<keygen>
<link rel="stylesheet" href="">
<meta>
<param name="" value="">
<source>
<track>
<wbr>
```

### 1.13

Un servidor web (es un ordenador de gran potencia que se encarga de “prestar el servicio” de transmitir la información pedida por sus clientes (otros ordenadores, dispositivos móviles, impresoras, personas, etc.)

Los servidores web son un componente de los servidores que tienen como principal función almacenar, en web hosting, todos los archivos propios de una página web (imágenes, textos, videos, etc.) y transmitirlos a los usuarios a través de los navegadores mediante el protocolo HTTP. El proceso es un ejemplo del modelo cliente / servidor.

Un cliente es un ordenador o software que accede a un servidor y recupera servicios especiales o datos de él. ... Un cliente no ejecuta tareas de servidor, sino que es simplemente un elemento intermedio. Los clientes típicos son los navegadores web o los clientes de correo electrónico.

#### 1.14

No, existen las arquitecturas redes de pares y Cliente-cola-Cliente

Si bien la clásica arquitectura C/S requiere uno de los puntos terminales de comunicación para actuar como un servidor, que puede ser algo más difícil de aplicar, la arquitectura **Cliente-Cola-Cliente** habilita a todos los nodos para actuar como clientes simples, mientras que el servidor actúa como una cola que va capturando las peticiones de los clientes (un proceso que debe pasar sus peticiones a otro, lo hace a través de una cola, por ejemplo, una consulta a una base de datos, entonces, el segundo proceso conecta con la base de datos, elabora la petición, la pasa a la base de datos, etc.).

La arquitectura **par a par** no posee clientes ni servidores, en la arquitectura p2p pura, se trabaja sin la dependencia de un servidor central, los usuarios no están limitados por las reglas convencionales a seguir que definen los servidores y no utilizan Dispositivos de Almacenaje.

#### 1.15

**JavaScript** es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas y JavaScript del lado del servidor (*Server-side JavaScript* o *SSJS*). Su uso en aplicaciones externas a la web, por ejemplo, en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

**PHP** es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web. PHP originalmente significaba *Personal Home Page* (Página personal), pero ahora significa el inicialismo recursivo *PHP: Hypertext Preprocessor*.

El código PHP suele ser procesado en un servidor web por un intérprete PHP implementado como un módulo, un daemon o como un ejecutable de interfaz de entrada común (CGI).

**Flash** es una aplicación informática englobada en la categoría de reproductor multimedia.

Permitía reproducir archivos en formato SWF, creados con la herramienta Animate CC o Flex de la propia Adobe, o con otras herramientas de terceros. Estos archivos se reproducían en un entorno determinado. En un sistema operativo tenía el formato de aplicación del sistema, mientras que, si el entorno era un navegador, su formato era el de un complemento u objeto ActiveX. Flash Player tenía soporte para un lenguaje de programación interpretado conocido como ActionScript basado en el estándar ECMAScript.

**JavaServer Pages** (JSP) es una tecnología que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML y XML, entre otros tipos de documentos. JSP es similar a PHP, pero usa el lenguaje de programación Java.

**Active Server Pages** , también conocido como **ASP clásico**, es una tecnología de Microsoft del lado del servidor para páginas web generadas dinámicamente, que ha sido comercializada como un anexo a Internet Information Services (IIS).

Lo interesante de este modelo tecnológico es poder utilizar diversos componentes ya desarrollados como algunos controles ActiveX así como componentes del lado del servidor, tales como CDONTS, por ejemplo, envío de correo utilizando CDONTS que permite la interacción de los scripts con el servidor SMTP que integra IIS.

**AJAX**, acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones web asíncronas. Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible interactuar con el servidor sin necesidad de recargar la página web, mejorando la interactividad, velocidad y usabilidad en las aplicaciones.

2.

Pulsando ctrl+u se puede **acceder al código fuente** de la página en Google, Firefox y Opera.

La herramienta de **inspeccionar elementos** , en Google se acciona con la tecla F12, en Opera ctrl+shift+i y en Firefox es una utilidad que permite ver el código fuente subyacente de cualquier página web. Además, puedes usarlo para realizar cambios temporales y ver los resultados en tiempo real, dejando intacto el código fuente original.

Esto es extremadamente útil si necesitas probar un cambio o diagnosticar un problema. También puede ser de gran ayuda si encuentras un sitio con una característica que te guste y tienes curiosidad por ver cómo se implementó.

La ventana de Firefox **Información de la página** te da los detalles técnicos de la página en la que te encuentras y te permite cambiar los permisos de esa página. Para abrirla: Haz clic con el botón secundario en una zona en blanco de la página web y, en el menú contextual, selecciona la opción Ver información de la página.

3.

```
<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>CurriculumVitaeBruceWayne</title>
</head>

<body>
  <h1>Curriculum Vitae de Bruce Wayne</h1>
  <h3>Datos personales</h3>
  <ul>
    <li>Nombre completo: <b>Bruce Wayne</b></li>
    <li>Fecha de nacimiento: <b>1/5/1939</b> </li>
    <li>Lugar de nacimiento: <b>Gotham City</b> </li>
  </ul>

  <h3>Formación académica</h3>
  <ul>
    <li>1956-1961: <b>Universidad del Espantapajaros</b></li>
    <li>1952-1956: <b>Instituto de Dos Caras</b> </li>
    <li>1944-1952: <b>Escuela Primaria del Joker</b> </li>
  </ul>

  <h3>Experiencia laboral</h3>
  <ul>
    <li>1975-1985: <b>En el paro</b></li>
    <li>1965-1975: <b>Cazavillanos y demás chusma</b> </li>
    <li>1962-1965: <b>Aprendiz de superhéroe </b> </li>
  </ul>
</body>
</html>
```



## 4.

### 4.1

```
<acronym></acronym>
```

-> Se utilizó en HTML4 para abreviaturas y se cambió por `<abbr></abbr>`

```
<applet></applet>
```

-> se utilizó en HTML 4 para definir un complemento  
-> Los complementos son programas informáticos que amplían la funcionalidad estándar del navegador.  
-> La mayoría de los navegadores ya no son compatibles con los complementos y subprogramas de Java.  
Los controles ActiveX ya no son compatibles con ningún navegador.  
La compatibilidad con Shockwave Flash también se ha desactivado en los navegadores modernos.

```
<basefont>
```

-> La etiqueta se utilizó en HTML 4 para especificar un color de texto, un tamaño de fuente o una familia de fuentes predeterminados para todo el texto de un documento HTML. En su lugar se usan hojas de estilos CSS:  
`font-family: font-size: color:...`

```
<big></big>
```

-> La etiqueta se usó en HTML 4 para definir un texto más grande.  
-> En su lugar se usan hojas de estilos: `font-size: 50px;`

```
<center></center>
```

-> La etiqueta se usó en HTML 4 para centrar el texto a la página.  
-> En su lugar se usan hojas de estilos: `text-align:center;`

```
<dir></dir>
```

-> La etiqueta se usó en HTML 4 para definir un texto más grande.  
-> En su lugar se usan hojas de estilos: `font-size: 50px;`

```
<font></font>
```

-> La etiqueta se usó en HTML 4 para especificar el tipo de fuente, el tamaño de fuente y el color del texto.  
-> En su lugar se usan hojas de estilos: `font-family: font-size:...`

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

-> La etiqueta se utilizó en HTML 4 para definir un conjunto de marcos.

```
<frame>
```

-> La etiqueta se usó en HTML 4 para definir una ventana en particular (marco) dentro de un `<frameset>`.

```
<noframes></noframes>
```

-> La etiqueta se usó en HTML 4 para actuar como una etiqueta de respaldo para los navegadores que no admitían marcos.

### 4.2

```
<!DOCTYPE HTML>
```

```
<audio></audio>
```

```
<video src=""></video>
```

Nuevas APIs, Comunicación entre varias páginas

#### 4.3

Las etiquetas `<tr>` (del inglés "table row") definen cada fila de la tabla y encierran todas las columnas. Por último, la etiqueta `<td>` (del inglés "table data cell") define cada una de las columnas de las filas, aunque realmente HTML no define columnas sino celdas de datos.

`<tr>` acepta los atributos `abbr`, `colspan`, `headers`, `rowspan` y `scope`

`<th>` acepta los atributos `colspan`, `headers` y `rowspan`

#### 4.4

La etiqueta `<time>` es soportada en la versión 62.0 de Google, 18.0 de Edge, 22.0 de Firefox, 7.0 de Safari y 49.0 de Opera

### 5

Surge en 1980.

El primer navegador capaz de soportar HTML fue Mosaic.

NetScape se anuncia el 13 de octubre de 1994 y desaparece el 1 de enero de 2008 y Internet Explorer fue su sucesor.

Java Script surge el 4 de diciembre de 1995 y Java surgen en el año 1996. **JavaScript** y **Java** son lenguajes totalmente diferentes. Pero, ¿cuáles son las diferencias más destacadas entre ambos lenguajes? Aquí podéis ver algunas diferencias:

**JavaScript** ha sido un lenguaje interpretado, y **Java** compilado. Los programas de **JavaScript** son archivos de texto que se integra directamente en las páginas HTML y es interpretado (sin estar compilado) por el cliente (navegador), mientras que en **Java** se compilan a un archivo especial para que ser optimizados a un lenguaje intermedio llamado bytecode, y leído posteriormente en un ordenador que lo ejecute.

**Java** es un lenguaje de programación orientado a objetos puros (OOP), mientras que **JavaScript** está basado en prototipos y, puede emular la programación orientada a objetos.

**JavaScript** es gestionado por ECMAScript, una organización sin ánimo de lucro. En **Java**, el control lo tiene Oracle, una empresa privada que lo gestiona en función de sus intereses.

**JavaScript** se depura en una fase y Java en dos. JavaScript hace que el código trabaje informando de los errores que se producen a medida que se ejecutando. **Java**, primero hace la fase de compilación y el compilador nos indica los posibles errores de sintaxis presentes en nuestro código. Después, se ejecuta el programa, donde pueden surgir errores para ser depurados.

**Java** tiene variables definidas que no se pueden cambiar y es más complejo, **JavaScript** puede ser cambiante, dándole flexibilidad y es más sencillo.

**JavaScript** es débilmente tipado, una misma variable puede contener primero un texto, luego un número, un array o un objeto. Esto, que parece una ventaja, también es fuente de posibles errores si no controlamos correctamente lo que ocurre en nuestro código. **Java** es fuertemente tipado, todas las variables tienen un tipo determinado y, una vez definidas, no se pueden cambiar.

Flash surge en 1996 y qué es ya lo he respondido esto en el mismo documento como muchas otras preguntas.

HTML5 se presentó el 27 de octubre de 2014 y se convirtió en una recomendación oficial de la W3C el 29 de octubre de 2014.

Chrome es el navegador mas joven del mercado