

Ésta es la primera lección del curso de fundamentos de **HTML**.

Para comenzar, se presenta formalmente qué es la web y cuáles son sus principales componentes. A continuación, se introduce las principales tecnologías para desarrollar sitios webs. Y finalmente, se resume el plan de estudio del curso.

Al finalizar la lección, el estudiante sabrá:

- Qué es la **web**.
- Qué es **HTML**.
- Para qué sirve **HTML**.
- Cuál es el programa del curso.

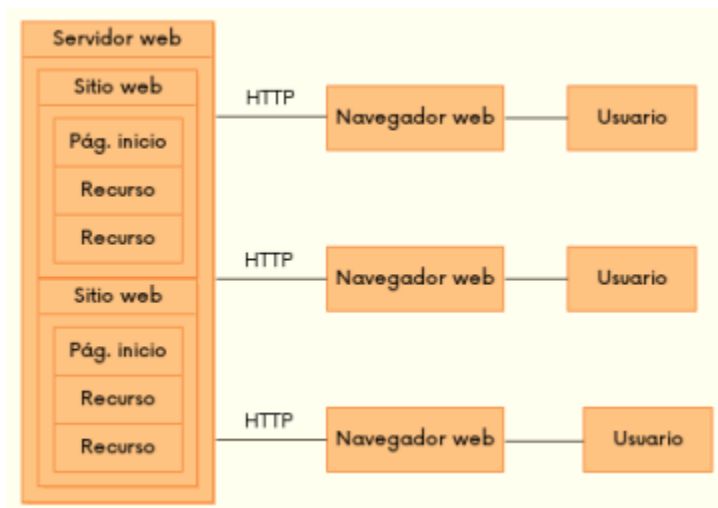
Introducción

La **web**, también conocida formalmente como **world wide web**, es un sistema de información a través del cual compartir recursos como, por ejemplo, documentos **HTML**, **PDFs**, imágenes, audio y video. Fue creada en 1989 por **Tim Berners-Lee**. Un **sistema de información** (*information system*) no es más que un grupo de componentes trabajando juntos en un objetivo común, en nuestro caso, el intercambio de recursos.

Las principales características de la web son:

- Es multiplataforma, es decir, es accesible desde cualquier tipo de dispositivo bajo cualquier sistema operativo.
- Es dinámico, esto es, su contenido puede cambiar en cualquier momento, rápidamente.
- Es interactivo, o sea, podemos movernos a través de la web fácil y, generalmente, rápidamente.

Los principales componentes de la web son los recursos webs, los sitios webs, los servidores webs y los navegadores webs.



Un **recurso web** (*web resource*) es un elemento que puede intercambiarse como, por ejemplo, un archivo. Tienen un **URL**, un identificador que actúa como dirección única que permite su acceso.

Un **sitio web** (*website*) es un conjunto de recursos webs accesibles mediante la web. Estos sitios webs tienen lo que se conoce como **página de inicio** (*home page*) o **página índice** (*index page*), un documento **HTML** que actúa como punto de entrada del sitio.

Los **servidores webs** (*web servers*) son los ordenadores que publican sitios webs, poniéndolos así a disposición de los usuarios de la web. Un mismo servidor web puede publicar varios sitios webs. Y la web está formado por millones de servidores webs. Las organizaciones que operan estos servidores, se conocen como **entidades de alojamiento** (*hosting organizations*). Entre otras encontramos **GitHub**, **GoDaddy** y **OVH**.

Por su parte, un **navegador web** (*web browser*) son los programas principales a través de los cuales los usuarios pueden acceder a los sitios webs y que se encargan de presentar los recursos accedidos. Los navegadores más conocidos son **Chrome**, **Edge**, **Firefox** y **Safari**.

Para que los programas que hacen posible la web puedan entenderse, es necesario algún tipo de mecanismo que conozcan y permita el acceso. En este punto, encontramos algunos protocolos como **HTTP**, **TCP** e **IP**. Básicamente, mediante los protocolos **TCP** e **IP** se puede identificar y acceder a los dispositivos, tanto servidores como clientes. Son independientes de la web. Se utilizan en muchos sistemas de información. En cambio, **HTTP** es específico de la web y es el que permite el intercambio de recursos webs.

Sitios webs

El objetivo de este curso es presentar uno de los lenguajes claves para el desarrollo de sitios webs: **HTML**. Para poder programar la web, desde el lado cliente, no hay que ser un ávido conocedor de **TCP** e **IP**. **HTTP** tampoco, pero es muy recomendable. Ahora bien, sí hay que serlo de **HTML**, **CSS** y **JavaScript**, principalmente. También es muy útil conocer otras tecnologías adicionales que permiten crear sitios webs profesionales como, por ejemplo, **Sass**, **Bootstrap** y **React**.

HTML

Un **lenguaje** (*language*) es un sistema de comunicación; un conjunto de signos y reglas que permite la comunicación. A través de él, las personas manifestamos lo que pensamos o sentimos. Un **lenguaje de programación** (*programming language*) es aquel que permite la comunicación con el ordenador. Se utiliza para crear programas o aplicaciones. Entre otros lenguajes de programación, podemos encontrar **C/C++**, **Dart**, **Java**, **JavaScript** o **Python**.

Además también tenemos los lenguajes de marcas. Un **lenguaje de marca** (*markup language*) es aquel que se utiliza para escribir documentos de texto mediante un conjunto de etiquetas o marcas. Siendo **HTML** y **XML** los más utilizados y reconocidos actualmente.

Así pues, **HTML** (*Hyper Text Markup Language*) es un lenguaje de marcas, el principal lenguaje para escribir sitios webs. Está desarrollado y estandarizado por el **W3C** (*World Wide Web Consortium*) y el **WHATWG** (*Web Hypertext Application Technology Working Group*). Fue desarrollado inicialmente en 1989 por **Tim Berners-Lee**, el creador de la web, en el **CERN** para que los investigadores del centro pudieran compartir sus documentos. A finales de la década de 1990, se delegó en el **W3C** el mantenimiento, la evolución y la estandarización del lenguaje **HTML**. El estándar ha sufrido varias revisiones a lo largo de los años, pero todas ellas muy estables y duraderas. Actualmente, la última versión estandarizada y recomendada para nuevos proyectos es **HTML5**, que se presenta a lo largo del curso, siendo la más potente e interesante versión del estándar vista hasta la fecha.

Es soportado por todos los navegadores como, por ejemplo, **Chrome**, **Edge**, **Firefox**, **Opera** y **Safari**.

HTML se utiliza principalmente para componer o proporcionar el contenido de la página o documento web. También puede darle formato, aunque esta tarea se delega principalmente en **CSS**. ¿Qué quiere decir *componer o proporcionar el contenido*? Sencillo. Se habla del texto que se presentará a los usuarios repartido en párrafos, secciones, artículos, etc. De su estilo, esto es, colores, tipo de letra, etc., aunque se puede hacer también con **HTML**, se delega en **CSS**.

CSS

CSS (*Cascading Style Sheets*) es el lenguaje principal para definir el formato de un documento web, esto es, los colores, los tipos de letra, la ubicación de las áreas de texto, etc. Al igual que el lenguaje **HTML**, se encuentra estandarizado por el **W3C**.

Es importante tener clara la separación del contenido del documento de su presentación. Esta separación permite que un mismo documento pueda ser presentado al usuario de distintas formas de

una manera muy sencilla. Vamos a insistir en ello. El contenido de un documento hace referencia al texto a presentar a los usuarios, sin importar el formato en que se hará. En cambio, la presentación de un documento define cómo debe visualizarse el contenido al usuario, haciendo hincapié en los colores, los tipos de letra, la separación de párrafos, los márgenes, los bordes, el relleno, etc. Ambas cosas se pueden realizar únicamente con **HTML**, pero se recomienda encarecidamente utilizar **HTML** para el contenido y **CSS** para la presentación. Y así lo haremos a lo largo del curso. Vamos a exponer el lenguaje de marcas **HTML**, como medio para proporcionar el contenido de un sitio web. Dejando su formateo para un curso específico de **CSS** y **Sass**.

JavaScript

JavaScript es un lenguaje de programación muy utilizado en páginas webs. Se utiliza principalmente para procesar o atender las interacciones de los usuarios con los documentos o páginas webs. Y para proporcionar contenido dinámicamente. Se encuentra estandarizado por la **ECMA** (*European Computer Manufacturers Association*).

Para convertirse en un buen programador web, además de conocer **HTML** y **CSS**, el estudiante también debe ser hábil en este lenguaje de programación.

Información del curso

Este curso es la primera parte de una especialización de cursos dedicada al desarrollo de aplicaciones webs y móviles.

Tiene como objetivo presentar los fundamentos del desarrollo de aplicaciones webs con **HTML**. Dejándose para otros cursos especializados otras tecnologías como **CSS** y **Sass**, **Bootstrap**, **JavaScript**, **Node**, **React** y **React Native**.

Al finalizarlo, el estudiante sabrá:

- Qué es **HTML**.
- Cómo proporcionar contenido mediante documentos **HTML**.
- Para qué se utiliza **HTML**.

Este curso se recomienda a aquellas personas que deseen desarrollar su carrera como desarrolladores webs, tanto *front-end* como *back-end*.

Conocimientos previos

El estudiante no requiere ningún conocimiento especial.

Plan de estudio

El curso tiene una duración aproximada de 4 horas. Se divide en 8 lecciones, cada una de ellas con una parte de teoría y generalmente una de práctica. Al finalizar el curso, puede realizar un examen de tipo test, con el que evaluar los conocimientos adquiridos.

El enfoque a seguir es muy sencillo: ir lección a lección; primero hay que leer la teoría y, después, realizar la práctica. Se recomienda encarecidamente que el estudiante realice cada lección, tanto teoría como práctica, en el mismo día, con el menor número de interrupciones a lo largo de su estudio. Al finalizar el curso, se recomienda encarecidamente realizar el examen.

A continuación, se enumera las distintas lecciones y el tiempo estimado para su estudio:

Lección	Teoría	Práctica	Descripción
1 Introducción	10min	-	Esta lección.
2 Documentos HTML	20min	10min	Descripción de qué es un documento o página HTML y cómo se estructura.
3 Texto en HTML	20min	20min	Cómo dividir el contenido del documento HTML en secciones, artículos o partes claramente definidas.
4 Listas y tablas en HTML	15min	20min	Cómo crear listas y tablas de contenido.

5 Enlaces en HTML	15min	20min	Cómo enlazar con otros recursos webs e incrustar imágenes en un documento HTML .
6 Audio y video en HTML	10min	15min	Cómo incrustar archivos de audio y video en documentos HTML .
7 Formularios HTML	20min	20min	Cómo crear formularios HTML a través de los cuales los usuarios pueden introducir datos.
8 Validación de formularios HTML	5min	10min	Cómo validar los datos introducidos por los usuarios mediante formularios HTML .
Examen	60min		Evaluación de los conocimientos adquiridos por el estudiante.

Información de publicación

Título **Fundamentos de HTML**

Autor **Raúl G. González - raulggonzalez@nodemy.com**

Primera edición **Octubre de 2016**

Versión actual **1.0.0**

Versión de HTML **HTML5**

Contacto **hola@nodemy.com**