

# Usabilidad y Accesibilidad

## Práctica 1: HTML (1)

### 1. Objetivos

- Aprender el lenguaje de marcado HTML y la importancia de escribir código correcto y conforme a la especificación oficial.
- Aprender a estructurar una página web con HTML5.
- Conocer algunas herramientas que ayudan a escribir código HTML correcto y compatible con distintos navegadores.
- Aprender algunos consejos que se deben aplicar para crear páginas web usables.
- Aprender algunas características que se deben emplear para crear páginas web accesibles.

### 2. Recursos

¿Qué etiquetas existen en HTML? ¿Qué atributos existen y qué valores pueden tomar? ¿Cómo se emplean las etiquetas?

- **W3Schools**<sup>1</sup>: cursos de aprendizaje y guías de referencia de diversas tecnologías empleadas en la programación web.
- **W3C**<sup>2</sup>: organismo internacional que desarrolla las especificaciones de las principales tecnologías que se emplean en la programación web.

¿Cómo puedo saber que el código HTML que he escrito es correcto?

- **W3C Markup Validation Service**<sup>3</sup>: servicio de validación para diferentes lenguajes de marcado (HTML, XHTML, SVG, y otros).
- **Html Validator**<sup>4</sup>: complemento para Mozilla Firefox que permite comprobar el número de errores que posee una página web que se visualiza.
- **Validity**<sup>5</sup>: complemento para Google Chrome que permite comprobar el número de errores que posee una página web que se visualiza.

¿Cómo puedo comprobar que la página web se visualiza correctamente en diferentes navegadores?

- **Website Responsive Testing Tool**<sup>6</sup>: permite comprobar cómo se visualiza una página web en diferentes dispositivos con diferentes tamaños de pantalla.

¿Existe alguna herramienta que me pueda ayudar a escribir el código HTML?

- **Notepad++**<sup>7</sup>: editor gratuito de código fuente que soporta varios lenguajes de programación. Entre otras características, posee sintaxis coloreada, envoltura de sintaxis y autocompletado.

---

<sup>1</sup><https://www.w3schools.com>

<sup>2</sup><https://www.w3.org>

<sup>3</sup><https://validator.w3.org>

<sup>4</sup><https://addons.mozilla.org/es-ES/firefox/addon/249>

<sup>5</sup><https://chrome.google.com/webstore/detail/validity/bbicmjbohdfglopkidebfccilipgeif?hl=es>

<sup>6</sup><https://responsivetesttool.com/>

<sup>7</sup><https://notepad-plus-plus.org/>

- **Sublime Text**<sup>8</sup>: excelente editor compatible con múltiples lenguajes de programación y disponible para Windows, OS X y Linux; se puede probar de forma gratuita, pero para un uso continuo hay que adquirir una licencia.
- **Real-time HTML Editor**<sup>9</sup>: editor WYSIWYG de HTML en página web.
- **Online-HTML-Editor.org**<sup>10</sup>: editor WYSIWYG de HTML en página web que permite abrir y guardar una página web.
- **Tidy HTML Online**<sup>11</sup>: limpia y organiza el código HTML.

### 3. ¿Qué tengo que hacer?

MedlinePlus<sup>12</sup> es:

[...] el sitio web de los Institutos Nacionales de la Salud para pacientes, familiares y amigos. Producida por la Biblioteca Nacional de Medicina de los Estados Unidos, la biblioteca médica más grande del mundo, MedlinePlus le brinda información sobre enfermedades, afecciones y bienestar en un lenguaje fácil de leer. MedlinePlus le ofrece información confiable y actualizada en todo momento, en cualquier lugar y de forma gratuita.

MedlinePlus incluye la Enciclopedia Ilustrada de Salud<sup>13</sup> (*Health Illustrated Encyclopedia*) de A.D.A.M. incluye más de 4.000 artículos acerca de enfermedades, exámenes, síntomas, lesiones y procedimientos quirúrgicos. En la Figura 1 se muestra una captura de pantalla de la página principal de la enciclopedia, en la que se distingue un buscador, una barra de navegación horizontal en la parte superior y una barra de navegación compuesta de las letras del alfabeto. Además existen algunos enlaces a las redes sociales de la organización y enlaces a otras páginas.



Figura 1: Enciclopedia Ilustrada de Salud de MedlinePlus

<sup>8</sup><https://www.sublimetext.com/>

<sup>9</sup><https://htmledit.squarefree.com/>

<sup>10</sup><https://online-html-editor.org/>

<sup>11</sup><https://tidy.js.org/html/>

<sup>12</sup><https://medlineplus.gov/spanish/>

<sup>13</sup><https://medlineplus.gov/spanish/encyclopedia.html>

Mayo Clinic<sup>14</sup> es “un líder mundial de atención médica, investigación y educación para todo tipo de personas, sin fines de lucro”.

Mayo Clinic incluye diferentes glosarios como, por ejemplo, Enfermedades y afecciones<sup>15</sup>. En la Figura 2 se muestra una captura de pantalla de la página principal de Enfermedades y afecciones, en la que se distingue un buscador, una barra de navegación horizontal en la parte superior y una barra de navegación compuesta de las letras del alfabeto. Además, tal como se puede ver en la Figura 3, en esta página también se muestra una sección llamada “Temas publicados” en la que se destaca algunos de los artículos del glosario. Además existen algunos enlaces a las redes sociales de la organización y enlaces a otras páginas.



Figura 2: Enfermedades y afecciones de Mayo Clinic

En esta práctica debes construir un sitio web que sea una mini Enciclopedia Ilustrada de Salud, similar a las que existen en MedlinePlus o Mayo Clinic.

Las páginas web que debe contener son:

**Página principal** Contiene una explicación del sitio web, un índice alfabético de temas (una barra de navegación A-Z con un enlace por cada letra del alfabeto) y un panel “Temas destacados” en el que se muestra el resumen de algunos de los artículos de la enciclopedia y un enlace para acceder al artículo completo. El enlace puede ser un enlace a un artículo del propio sitio web o de otro sitio web cualquiera.

**Página índice de letra** Contiene un índice de los artículos correspondientes a la letra seleccionada. No es necesario crear esta página para todas las letras, solo para aquellas letras que tengan algún artículo. **Crea una página de error** (una sola página), del estilo “Error 404 - Página no encontrada”, para las letras que no tengan un artículo<sup>16</sup>.

**Página de artículo** Contiene una pequeña descripción, las causas, los síntomas, las pruebas y exámenes, el tratamiento y las expectativas. **El sitio web debe incluir al menos cuatro páginas de artículo de al menos dos letras distintas** (al menos cuatro páginas en total, no significa cuatro páginas por cada letra).

En concreto, tienes que crear las páginas que se indican en la Figura 4.

En esta práctica se tiene que hacer uso de los siguientes elementos de HTML:

<sup>14</sup><https://www.mayoclinic.org/es-es/about-mayo-clinic>

<sup>15</sup><https://www.mayoclinic.org/es-es/diseases-conditions>

<sup>16</sup>Es decir, todos los enlaces a páginas que no existan deben dirigir a la misma página de error.

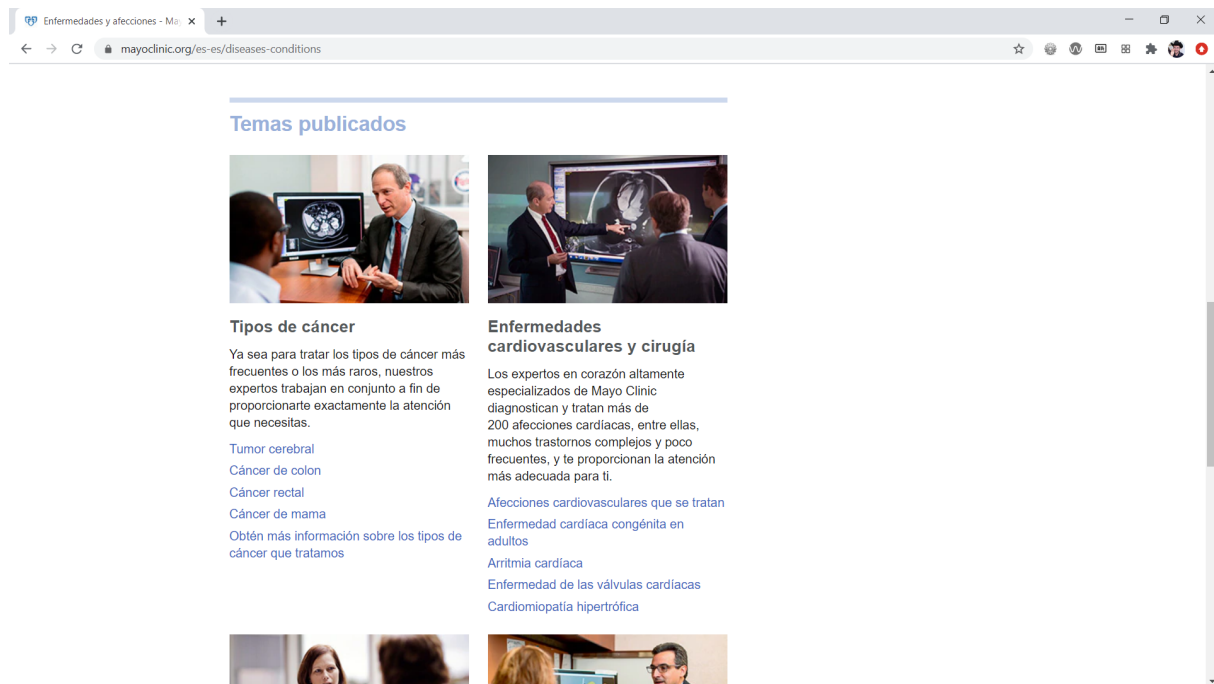


Figura 3: Panel con artículos destacados en Enfermedades y afecciones de Mayo Clinic

- Encabezados.
- Párrafos.
- Listas.
- Enlaces.
- Imágenes.
- Elementos semánticos como citas, direcciones, palabras destacadas, etc.

Además, incluye también al menos dos vídeos de YouTube, Vimeo o algún sitio similar mediante el mecanismo de “Insertar vídeo” que suelen proporcionar.

**Muy importante: el objetivo de esta asignatura es que el sitio web sea usable y accesible, este requisito lo debes tener presente desde el principio.** Sin embargo, como los conocimientos necesarios para lograrlo los irás adquiriendo poco a poco, el sitio web lo irás mejorando práctica a práctica, hasta lograr que sea totalmente usable y accesible en la última práctica.

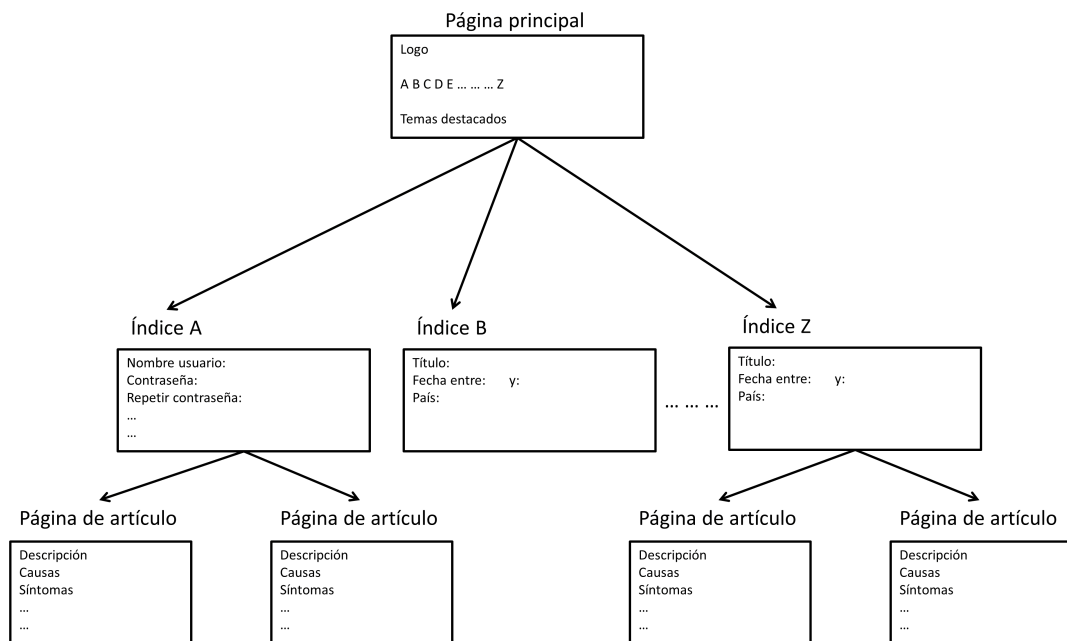


Figura 4: Diagrama de páginas que componen el sitio web

## 4. ¿Cómo lo hago?

En esta práctica ya tienes que ser capaz de evitar 4 de los 10 errores de accesibilidad más comunes que se explican en “Top Ten Most Common Web Accessibility Issues”<sup>a</sup>:

- **Image alt text errors**

Not including alternative text for images can result in a confusing and negative experience for screen reader users. Providing descriptive text via the alt attribute for an image element is an area where we see many issues.

- **Non-descriptive text for hyperlinks**

Many times, the link text will not provide enough detail to allow for the understanding of the link’s purpose out of context. For example, links may be present on the page that use the text ‘click me’ or ‘click here’.

- **Improper use of heading elements**

All too often, a page with considerable content will lack heading elements that help screen reader users understand the relationship of the page content, as well as provide a means to easily navigate through the page.

- **Same descriptive text for different resources**

Using the same descriptive text for links that point to different resources. When focused on creating excellent user experiences, it’s best to make your descriptive text relevant and unique.

<sup>a</sup><https://www.tpgi.com/ten-common-web-accessibility-issues/>

### 4.1. Escribir una página web

Antes de ponerte a escribir el código HTML, realiza un diagrama de las páginas web que componen tu sitio web. Este diagrama puede tener tanto detalle como tú quieras, pero hay que encontrar un equilibrio entre utilidad y legibilidad: un diagrama muy simple y con poco contenido será poco útil, mientras que

un diagrama muy complejo y con mucha información será difícil de entender y no supondrá una ayuda.

Todas las páginas, incluida la página de error, deben mostrar una identidad visual que permita identificar que la página pertenece a un mismo sitio web. Entre otras cosas, esto se logra manteniendo la misma cabecera y pie de página en todas las páginas. Por tanto, crea una página web “esqueleto” a partir de la cual puedas crear todas las páginas de tu sitio web. En esta página incluye todos los elementos que se repitan en todas las páginas (cabecera y pie de página, barra de navegación). Asegúrate que el código de esta página web es correcto.

En el pie de página incluye los nombres de los autores de la práctica, un aviso de copyright con el año y alguna información más.

## 4.2. Estructura de una página web

En HTML5 existen unas etiquetas semánticas para definir regiones especiales que tienen la mayoría de las páginas web: cabecera, menú principal, pie de página, etc. Todas las páginas del sitio web deben estar estructuradas siguiendo las etiquetas semánticas de HTML5 que definen las regiones más importantes en una página:

- La cabecera se etiqueta con `<header>`.
- La barra o menú de navegación con `<nav>`.
- Un panel con información adicional con `<aside>`.
- La zona de contenido principal con `<main>`.
- Las entradas o artículos con `<article>`.
- Las secciones de un artículo con `<section>`.
- Una imagen con `<figure>` y `<figcaption>`.
- Y el pie de página con `<footer>`.

**Es una buena práctica que todo el contenido de una página web se encuentre dentro de alguna de estas etiquetas.**

En la Figura 5 se muestra un ejemplo de una página web con las regiones identificadas<sup>17</sup>.

## 4.3. Características básicas de accesibilidad

Presta atención a algunas características de accesibilidad que ya has aprendido:

- El título de las páginas debe incluir el título de la página web y el título del sitio web. Existen diferentes formas de hacerlo, por ejemplo:
  - Grado en ingeniería biomédica - Universidad de Alicante
  - Grado en ingeniería biomédica | Universidad de Alicante
  - Grado en ingeniería biomédica :: Universidad de Alicante
- Se debe incluir un enlace para volver a la página principal en todas las páginas. Este enlace suele estar situado en el título o logo del sitio web, pero también puede aparecer como un enlace adicional que ponga “Inicio” o “Página principal” (es decir, habría dos enlaces). También puede aparecer un enlace más al final de la página, para que el usuario no tenga que volver al principio de la página a buscar el enlace.
- Se debe evitar que el usuario tenga que pulsar el botón “Atrás” en el navegador. Para ello, se debe poner enlaces que vuelvan “Atrás” en la propia página.
- Utiliza los encabezados de HTML solo para definir encabezados de la página, no los utilices para mostrar un texto más grande o en negrita, para eso utiliza CSS que aprenderás más adelante.

---

<sup>17</sup>Es muy probable que el código de la página web no siga realmente esta propuesta, es simplemente un ejemplo, no es una explicación de la página original.



Figura 5: Estructura típica de una página web

- Emplea las etiquetas de lista para crear listas, no definas listas hechas “a mano”. Cuando un contenido es una lista, se debe etiquetar como una lista. Por ejemplo, la barra de navegación A-Z es una lista, una lista de enlaces, por lo que se debe etiquetar como una lista (lista no numerada o numerada). Su apariencia visual como lista se podrá modificar en el futuro mediante CSS para que no parezca una lista (se podrá eliminar el símbolo de viñeta y se podrá poner en horizontal).
- Los textos de los enlaces tienen que ser significativos y se deben entender fuera del contexto en el que se encuentran. Además, cada enlace debe tener un texto único en la página, a no ser que sean enlaces que conduzcan a un mismo destino.
- Las imágenes deben tener un texto alternativo que describa la imagen o identifique la función que realiza la imagen. Cuando una imagen tenga únicamente una función decorativa, emplea un texto alternativo nulo o vacío.
- Los vídeos deben tener subtítulos. YouTube ofrece la opción de “subtítulos generados automáticamente”, pero en algunos casos la calidad de los subtítulos no es muy buena. Los vídeos que incluyas en la página web deben tener subtítulos añadidos por el propietario del vídeo.

#### 4.4. Sobre las etiquetas `article` y `section`

Las etiquetas `<article>` y `<section>` ayudan a estructurar el contenido de una página web. En el estándar oficial, `<article>`<sup>18</sup> está definido como:

The article element represents a complete, or self-contained, composition in a document, page, application, or site and that is, in principle, independently distributable or reusable, e.g. in syndication. This could be a forum post, a magazine or newspaper article, a blog entry, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

Y también en el estándar oficial, `<section>`<sup>19</sup> está definido como:

<sup>18</sup><https://html.spec.whatwg.org/#the-article-element>

<sup>19</sup><https://html.spec.whatwg.org/#the-section-element>

The section element represents a generic section of a document or application. A section, in this context, is a thematic grouping of content, typically with a heading.

Examples of sections would be chapters, the various tabbed pages in a tabbed dialog box, or the numbered sections of a thesis. A web site's home page could be split into sections for an introduction, news items, and contact information.

Es normal que exista confusión en el uso de estas dos etiquetas, ya que el estándar proporciona una definición muy abierta a la interpretación. Una explicación más concreta puede ser:

**<article>** Esta etiqueta se emplea de la siguiente forma:

- Propósito: Representa un contenido independiente y autocontenido.
- Cuándo usarla:
  - Entradas de blog.
  - Noticias o artículos de revista.
  - Publicaciones en un foro.
  - Comentarios de usuarios.
- Características clave:
  - El contenido tiene sentido por sí mismo fuera del contexto de la página.
  - El contenido puede ser reutilizado o redistribuido (por ejemplo, en un feed RSS).
  - Normalmente incluye un encabezado propio (**<h1>** - **<h6>**).

**<section>** Esta etiqueta se emplea de la siguiente forma:

- Propósito: Representa una sección temática o agrupación de contenido relacionado dentro de un documento.
- Cuándo usarla:
  - Para dividir un artículo o página en subapartados temáticos.
  - Para organizar un capítulo de un informe.
  - Dentro de una página de inicio con secciones como “Quiénes somos”, “Servicios”, “Contacto”.
- Características clave:
  - No siempre tiene sentido por sí sola fuera del documento.
  - Debe estar acompañada por un encabezado que indique su tema.
  - Sirve para dar estructura semántica jerárquica.

En “**<article>** vs. **<section>**: How To Choose The Right One”<sup>20</sup> se analiza el uso de estas dos etiquetas y se incluye la perspectiva de la accesibilidad en su uso, pero el análisis es muy complejo para una introducción a su uso.

## 4.5. Sobre la presentación visual de la página web

En esta práctica sólo tienes que definir la estructura y contenido de las páginas web, el estilo visual lo definiremos en prácticas posteriores (utilizando únicamente HTML es imposible cambiar el tipo de letra o el color de un texto en una página web). Emplea las etiquetas de HTML que poseen contenido semántico, como **<h1>**, **<p>**, **<ul>**, etc. Fíjate que las etiquetas **<center>** o **<font>** no existen ni en XHTML 1.0 ni en HTML5, aunque las puedes encontrar en muchas páginas web “mal hechas”.

**No utilices tablas para maquetar una página web.** Para todo ello emplearemos la tecnología CSS que utilizaremos en una próxima práctica.

Por tanto, recuerda que **en esta práctica no te tienes que preocupar por la presentación visual de las páginas web, solo tienes que utilizar HTML. Las presentación visual la definirás en las próximas prácticas con la tecnología CSS.** Por tanto, no te preocupes si la página se ve “fea”, por ahora lo importante es que escribas código HTML correcto.

<sup>20</sup><https://www.smashingmagazine.com/2022/07/article-section-elements-accessibility/>



## 4.6. Consejos básicos de usabilidad

En “Usability Guidelines for Accessible Web Design”<sup>21</sup> se proporcionan algunos consejos de usabilidad que debes aplicar en tu sitio web:

### ■ GRAPHICS AND MULTIMEDIA

2. Minimize the use of graphics.
3. Give all graphics (even advertising banners) names that are understandable and that thoroughly convey what the graphic is and does. Use ALT text to briefly describe images, and the LONGDESC attribute to thoroughly describe them.
4. Never blur pictures to indicate unavailability.
5. When graphics contain useful information, also provide the information in text.
6. Refer users to alternate ways for getting information contained in any graphics they encounter.
7. Do not shrink down a picture of an actual page on your site and use it as a graphic (or button) on another page.
8. When you do use graphics, always choose crisp and clear images.
9. Make it easy for users to skip any multimedia and Flash demos.
10. Do not automatically create a text-only version of your site.

### ■ LINKS AND BUTTONS

17. Limit the number of links on a page.
18. Avoid very small buttons and tiny text for links.
19. Leave space between links and buttons.
20. Avoid using images as the only method for linking to something.
21. Ensure that important commands appear as their own unique links.
22. Underline all links.
23. Create links within text when it makes sense. Use additional buttons only when it's necessary.

### ■ PAGE ORGANIZATION

24. Immediately confirm the company name once the homepage has loaded.
25. Immediately confirm what the page is once it has loaded.
26. Do not associate the word homepage with your company logo if you plan to reuse the same graphic on all pages.
27. Minimize the need for scrolling.
28. When users must make a choice, keep all possibilities in the same vicinity.
29. When users must make a choice, warn them that the choice is coming, and tell them how many options they have.
30. Design pages consistently.
31. Carefully consider using “Skip Links” so users can skip links or navigational elements.
32. Choose a simple, informative Web address for your site, and keep that URL in the address field after the page loads. (Make whatever appears in the address field logical.)

El consejo “10. Do not automatically create a text-only version of your site” está directamente relacionado con la accesibilidad de un sitio web. No se debe crear y proporcionar una versión “solo texto” de un sitio web como solución para garantizar la accesibilidad del sitio web, no se debe hacer ni de forma automática ni de forma manual.

Los consejos que dependen de la presentación de la página todavía no los puedes usar. Por ejemplo, los consejos “18. Avoid very small buttons and tiny text for links” y “19. Leave space between links and buttons” los podrás aplicar en las prácticas en las que tendrás que definir la presentación de tu sitio web mediante CSS.

---

<sup>21</sup>[https://media.nngroup.com/media/reports/free/Usability\\_Guidelines\\_for\\_Accessible\\_Web\\_Design.pdf](https://media.nngroup.com/media/reports/free/Usability_Guidelines_for_Accessible_Web_Design.pdf)

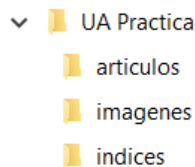


Figura 6: Posible organización de los ficheros del sitio web

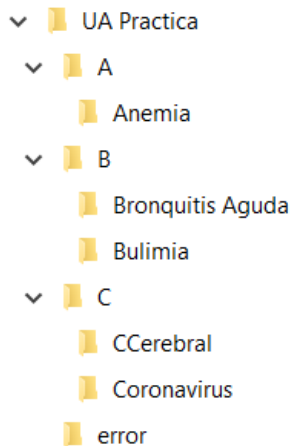


Figura 7: Otra posible organización de los ficheros del sitio web

#### 4.7. Organización de los ficheros del sitio web

Poner todos los ficheros (páginas web, imágenes, otros ficheros) de un sitio web en un mismo directorio puede estar bien para un sitio web pequeño, pero en cuanto el sitio web crezca, puede darte muchos problemas o incluso llegar a ser inmanejable.

En las prácticas de esta asignatura, debes establecer una organización de los ficheros que componen tu sitio web. La organización puede ser cualquiera que elijas, pero debe existir algún tipo de organización.

Por ejemplo, en la Figura 6 se muestra una posible organización en la que existe un directorio para los artículos, otro directorio para todas las imágenes del sitio web y otro directorio para las páginas web que son los índices (el índice para la letra A, el índice para la letra B, etc.). Además, en el directorio raíz se encuentra la página principal `index.html`.

En la Figura 7 se muestra otra posible organización en la que existe un directorio para cada letra del alfabeto y dentro de cada directorio existe un directorio por cada artículo que contiene todos los ficheros de ese artículo (la página web y las imágenes).

Si quieres aprender más sobre la organización de los ficheros en un sitio web, consulta la página web “Dealing with files”<sup>22</sup>.

### 5. Recomendaciones

El lenguaje HTML es muy sencillo de emplear: no es un lenguaje de programación, como mucha gente cree, sino un lenguaje de marcado compuesto de etiquetas (o elementos) que pueden contener atributos que poseen un valor. Las etiquetas no se pueden escribir de cualquier forma, sino que existen una serie de reglas que definen un orden entre las etiquetas.

Si escribes mal el código HTML, *normalmente* la página web se visualizará más o menos bien en la mayoría de los navegadores porque están preparados para admitir errores, pero en algunos casos la presentación de la página puede variar enormemente de un navegador a otro. Por tanto, **para asegurar que una página web se visualiza correctamente en la mayoría de los navegadores es imprescindible escribir un código correcto**. Puedes emplear The W3C Markup Validation Service<sup>23</sup> para

<sup>22</sup>[https://developer.mozilla.org/en-US/docs/Learn/Getting\\_started\\_with\\_the\\_web/Dealing\\_with\\_files](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Dealing_with_files)

<sup>23</sup><https://validator.w3.org/>

verificar que tu código sea correcto.

En la actualidad existen diferentes versiones de HTML: HTML 4.01, XHTML 1.0 y HTML5. Conviene que las conozcas todas: para un desarrollo nuevo, lo más conveniente es utilizar la última versión, pero es muy probable que tengas que trabajar también con páginas web desarrolladas con versiones antiguas de HTML. **Para realizar la práctica se recomienda utilizar HTML5.**

Para empezar a aprender HTML consulta algún curso de aprendizaje, como por ejemplo los disponibles en W3Schools:

- **HTML**<sup>24</sup>.
- **HTML - XHTML**<sup>25</sup>.

En HTML5 puedes emplear dos sintaxis<sup>26</sup>:

- HTML: la recomendada para la mayoría de los autores de páginas web.
- XHTML: una aplicación de XML, la sintaxis es más estricta que la anterior. Se recomienda si se quiere hacer uso de software para XML.

¿De dónde puedes copiar la estructura base de una página web y que sea correcta? Mira el código de la página del W3C<sup>27</sup> y copia lo mínimo necesario para crear una página web correcta. El siguiente código lo puedes utilizar para crear páginas web basadas en XHTML5 (sintaxis XML):

```
<!DOCTYPE html>
<html lang="es">
<head> <!-- La cabecera -->
<meta charset="utf-8" />
<meta name="generator" content="Bloc de notas" />
<meta name="author" content="Sergio Luján Mora" />
<meta name="keywords" content="HTML5, web" />
<meta name="description" content="Plantilla base de una página creada con HTML5" />
<title>Plantilla base de HTML5</title>
</head>
<body> <!-- El cuerpo -->
<p>
Bla, bla, bla.
</p>
</body>
</html>
```

Fíjate que pone `<meta charset="utf-8" />`, se recomienda usar el juego de caracteres UTF-8.

Aunque HTML sea muy flexible y permita a veces omitir algunas etiquetas, es mejor que escribas el código tal como lo suele hacer la mayoría de los desarrolladores. Por ejemplo, el siguiente código es válido:

```
<!DOCTYPE html>
<meta charset="utf-8">
<title>Hola HTML</title>
Hola Mundo
```

pero no es una buena idea omitir las etiquetas `<head>` y `<body>`.

Una vez hayas aprendido los conceptos básicos, consulta la guía de referencia de W3Schools (**HTML Reference**<sup>28</sup>) para conocer todas las etiquetas y atributos. Te vendrá muy bien recordar esta página web cuando estés escribiendo código HTML y te surja una duda.

Si quieres conocer más de HTML5, consulta la especificación oficial del W3C (**HTML5: A vocabulary and associated APIs for HTML and XHTML**<sup>29</sup>).

---

<sup>24</sup><https://www.w3schools.com/html/default.asp>

<sup>25</sup>[https://www.w3schools.com/html/html\\_xhtml.asp](https://www.w3schools.com/html/html_xhtml.asp)

<sup>26</sup>HTML Standard - 1.8 HTML vs XHTML: <https://html.spec.whatwg.org/multipage/introduction.html#html-vs-xhtml>

<sup>27</sup><https://www.w3c.es/>

<sup>28</sup><https://www.w3schools.com/tags/default.asp>

<sup>29</sup><https://html.spec.whatwg.org/multipage/>

Cuando escribas el código HTML a mano, utiliza la herramienta **W3C Markup Validation Service**<sup>30</sup> para comprobar que el código es correcto y la herramienta **Tidy HTML Online**<sup>31</sup> para obtener una mejor apariencia visual del código fuente.

Cuando se empieza a trabajar con una nueva tecnología es muy normal cometer muchos errores. ¿Cuáles son los errores más comunes? El artículo **Errores comunes del desarrollo web**<sup>32</sup> te lo explica.

A partir de ahora, almacena cada práctica en un directorio independiente; al final de la asignatura debes tener un directorio por cada práctica semanal. Cuando comiences una nueva práctica, copia y renombra el directorio de la práctica anterior.

**Importante:** el fichero de la página principal de un sitio web suele tener el nombre **index.html**. Es muy importante que lo nombres ya así, para evitar problemas cuando publiques tu sitio web en Internet en las siguientes prácticas.

**Por último, ten cuidado con los nombres de los ficheros:** utiliza únicamente letras del alfabeto inglés y números (no utilices caracteres no ASCII), no uses espacios en blanco y emplea únicamente minúsculas.

## 6. ¿Qué debo entregar?

- Todos los ficheros que componen el sitio web. Los vídeos que insertes no los debes entregar, los vídeos residen en Internet y tu página web sólo debe contener el código HTML necesario para incrustarlos.

---

<sup>30</sup><https://validator.w3.org>

<sup>31</sup><https://tidy.js.org/html/>

<sup>32</sup>[https://www.456bereastreet.com/archive/200408/web\\_development\\_mistakes\\_redux/](https://www.456bereastreet.com/archive/200408/web_development_mistakes_redux/)