



## TEMA 2: UTILIZACIÓN DE LENGUAJES DE MARCAS

---

# 1. DEL HTML AL XHTML

HTML es el lenguaje utilizado para crear la mayor parte de las páginas web. Es un estándar reconocido en todos los navegadores, por lo tanto, todos ellos visualizan una página HTML de forma muy similar.

El origen de HTML fue un sistema de hipertexto para compartir documentos electrónicos en 1980. La primera propuesta oficial para convertirlo en un estándar se realizó en 1993.

HTML 2.0 fue la primera versión oficial de HTML. El IETF (Grupo de trabajo en Ingeniería de Internet) lo publicó en septiembre de 1995.

---

- 
- HTML 3.2 se publicó el 14 de Enero de 1997 por el W3C. Incorpora los applets de Java y texto alrededor de las imágenes.
  - HTML 4.0 se publicó el 24 de Abril de 1998. Las novedades que presentaba fueron las hojas de estilos de CSS y la posibilidad de incluir pequeños programas en las páginas web.
  - HTML 4.01 se publicó el 24 de Diciembre de 1999. En este momento el W3C detuvo la actividad de estandarización de HTML hasta Marzo de 2007, momento en que se retoma y comienzan a publicarse los primeros borradores de HTML 5.0
  - Debido a la incompatibilidad con algunas herramientas basadas en XML se crea el lenguaje XHTML como combinación de ambos.
-

---

- HTML 5

Es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML:

Un “clásico” HTML (text/html), la variante conocida como HTML5 y una variante XHTML conocida como sintaxis XHTML5 que deberá ser servida como XML.

Debido a sus avances, mejoras y ventajas su uso por múltiples desarrolladores ha ido en aumento. Lo recomendable es usar la última versión para aprovechar el potencial de HTML5.

---

---

## 2. ESTRUCTURA DE UN DOCUMENTO HTML

La estructura de una página HTML ha de ser coherente para cualquier archivo XML. Por tanto, para mejor organización tendrá:

- **Prólogo:** en él indicamos el tipo de documento que se va a iniciar y, además, la versión de HTML utilizada para la realización del código, permitiendo que lo interprete correctamente.

- **Ejemplar:** es la parte del documento que está delimitado por las etiquetas `<html>` y `</html>`. El ejemplar se puede dividir en dos partes:

  - La cabecera, que está delimitada por las etiquetas `<head>` y `</head>`.

  - El cuerpo, contiene la información que se va a presentar en pantalla y todo el contenido se encuentra delimitado por las etiquetas `<body>` y `</body>`.

---

---

## 3. IDENTIFICACIÓN DE ETIQUETAS Y ATRIBUTOS DE HTML.

- Un documento HTML está formado por etiquetas y atributos.
  - Como en XML las etiquetas pueden ser de apertura, o de cierre. La principal diferencia es que en HTML las etiquetas están limitadas
  - Aunque HTML tiene gran cantidad de etiquetas, estas no son suficientes para crear páginas complejas. Como no se pueden crear etiquetas para cada elemento se hace uso de los atributos
  - Para cada uno de los atributos hay definido un conjunto de valores que se puede asignar, si el valor del atributo no es válido, el navegador lo ignora.
-

---

## 3.1. CLASIFICACIÓN DE LOS ATRIBUTOS COMUNES SEGÚN SU FUNCIONABILIDAD

**Atributos básicos:** Se pueden usar en casi todas las etiquetas HTML.

Atributo	Descripción
<b>name = "texto"</b>	Permite asignar el nombre "texto" a un objeto HTML
<b>title = "texto"</b>	Asigna un título a un elemento HTML, mejorando así la accesibilidad. Dicho título es mostrado por los navegadores cuando el usuario pasa el ratón por encima del elemento. Es especialmente útil con los elementos: a, link, img, object, abbr y acronym
<b>id = "texto"</b>	Permite identificar al elemento HTML sobre el que se aplica de forma única mediante el identificador "texto". Sólo es útil cuando se trabaja con CSS y con Javascript. No pueden empezar por números y sólo puede contener letras, números, guiones medios y/o guiones bajos.
<b>style = "texto"</b>	Permite aplicar al elemento HTML el estilo "texto" directamente.
<b>class = "texto"</b>	Permite aplicar al elemento HTML el estilo "texto" definido en las CSS. No pueden empezar por números y sólo puede contener letras, números, guiones medios y/o guiones bajos.

---

**Atributos para internacionalización:** Los utilizan aquellas páginas que muestran sus contenidos en varios idiomas o aquellas que quieren indicar de forma explícita el idioma de sus contenidos.

Atributo	Descripción																
<b>dir</b>	Indica la dirección del texto por lo que sólo puede tomar dos valores: ltr (left to right) de izquierda a derecha. Es el valor por defecto. rtl (right to left) de derecha a izquierda.																
<b>lang = "codigo"</b>	Especifica el idioma del elemento mediante un código predefinido. Los posibles valores de este atributo se encuentran en el documento RFC 1766, algunos de los valores posibles son: <table><tr><th>Código</th><th>Idioma</th><th>Código</th><th>Idioma</th></tr><tr><td>en</td><td>Inglés (Gran Bretaña)</td><td>es</td><td>Español</td></tr><tr><td>en-US</td><td>Inglés americano</td><td>fr</td><td>Francés</td></tr><tr><td>ja</td><td>Japones</td><td>fr-CA</td><td>Francés de Canada</td></tr></table>	Código	Idioma	Código	Idioma	en	Inglés (Gran Bretaña)	es	Español	en-US	Inglés americano	fr	Francés	ja	Japones	fr-CA	Francés de Canada
Código	Idioma	Código	Idioma														
en	Inglés (Gran Bretaña)	es	Español														
en-US	Inglés americano	fr	Francés														
ja	Japones	fr-CA	Francés de Canada														
<b>xml:lang = "codigo"</b>	Especifica el idioma del elemento mediante un código definido según la recomendación RFC 1766.																



---

## 3.2. ELEMENTOS HTML

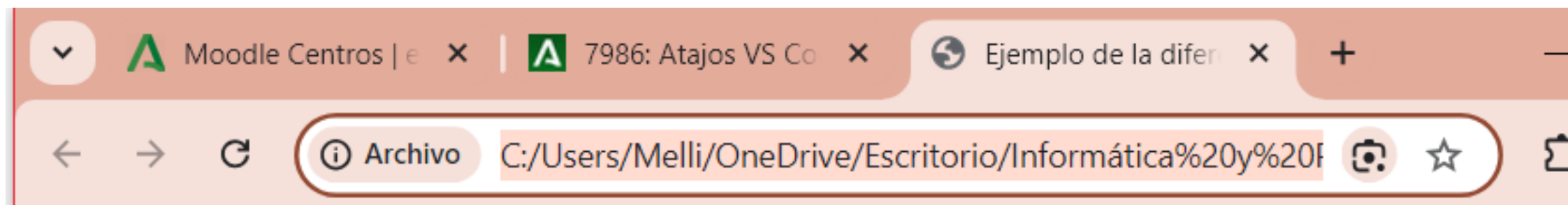
Un elemento HTML está formado por:

- Una etiqueta de apertura.
- Cero o más atributos.
- Texto encerrado por la etiqueta, aunque es opcional.
- Una etiqueta de cierre.

Según el modo en que ocupen el espacio disponible en la página los elementos pueden ser de dos tipos:

- Elementos en línea.
  - Elementos en bloque
-

```
1  <!DOCTYPE html>
2  <html lang="es">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=
7      , initial-scale=1.0">
8      <title>Ejemplo de la diferencia entre los elementos en línea y los elementos de bloque</title>
9  </head>
10
11 <body>
12     <h1>los encabezados son los elementos de bloque.</h1>
13     <p>Y los párrafos también.</p>
14     <a href="http://www.infoalisal.com">Los enlaces son elementos de línea</a>
15     <p>Incluso si está definido dentro de un párrafo, <b>un texto en negrita</b> sigue siendo un elemento en línea</p>
16 </body>
17
18 </html>
```



**los encabezados son los elementos de bloque.**

Y los párrafos también.

Los enlaces son elementos de línea

Incluso si está definido dentro de un párrafo, **un texto en negrita** sigue siendo un elemento en línea

---

---

## 3.2.1 ELEMENTOS DE LA ESTRUCTURA BÁSICA DEL DOCUMENTO

La estructura básica de un documento viene determinada por las siguientes etiquetas:

Elemento	Descripción
<b>html</b>	Documento HTML.
<b>head</b>	Cabecera del documento.
<b>body</b>	Cuerpo del documento. Permite definir formatos que se aplican a los elementos de la página de manera global, como son el color del fondo del texto, los márgenes, el color de los enlaces, ...

---

---

## 3.2.2. ELEMENTOS DE LA SECCIÓN DE CABECERA

Elementos contenedores:

- Title: Título del documento.
- Script: Script incrustado. Su contenido ha de ir situado entre las marcas de comentarios ya que no ha de ser interpretado.
- Style: Estilo aplicado al documento utilizando CSS. Su contenido ha de ir situado entre las marcas de comentarios ya que no ha de ser interpretado.

Elementos no contenedores:

- Base: URI base del documento.
  - Isindex: Prompt de entrada de datos
  - Link: enlaces a documentos externos de librerías.
  - Meta: Información que agiliza la búsqueda del documento en buscadores.
-

---

## 3.2.3 ELEMENTOS QUE DAN FORMATO AL TEXTO DE UN PÁRRAFO

Los distintos elementos que podemos utilizar para dar formato a nuestro texto son:

Los distintos elementos que podemos utilizar para dar formato a nuestro texto son:	
Elemento	Descripción
<b>p</b>	Delimita los párrafos
<b>h<i>i</i></b>	Encabezado de nivel <i>i</i> , donde <i>i</i> es un número entero entre 1 y 6, ambos inclusive. El tamaño de la letra del encabezado es mayor cuanto menor sea el valor de <i>i</i> . No deben de usarse estas etiquetas para formatear texto. Sólo estarán bien usadas para designar títulos de párrafos.
<b>b</b>	Indica que el texto que está en ese elemento se le pondrá en negrita.
<b>i</b>	Indica que el texto que está en ese elemento se le pondrá en itálica ó cursiva.
<b>u</b>	Indica que el texto que está en ese elemento se le pondrá subrayado.
<b>sup</b>	Indica que el texto que está en ese elemento es un supraíndice.
<b>sub</b>	Indica que el texto que está en ese elemento es un subíndice.
<b>strong</b>	Indica que el texto que está en ese elemento estará resaltado. Habitualmente los navegadores resaltan el texto poniéndolo en negrita aunque podría haber algún navegador que resaltase el texto poniéndolo en cursiva y en naranja.

---

---

# EJEMPLO 1:

- Crear una página web básica con la estructura mínima de HTML.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi primera página HTML</title>
7  </head>
8  <body>
9      <p>¡Hola, Mundo! Esta es mi primera página en HTML.</p>
10 </body>
11 </html>
```

---

# EJEMPLO 2: ENCABEZADOS Y PÁRRAFOS

- Crea una página que tenga todos los niveles de encabezados de <h1> a <h6>.
- Agrega algunos párrafos (<p>) para simular el contenido.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Encabezados y Párrafos</title>
7  </head>
8  <body>
9      <h1>Encabezado 1</h1>
10     <h2>Encabezado 2</h2>
11     <h3>Encabezado 3</h3>
12     <h4>Encabezado 4</h4>
13     <h5>Encabezado 5</h5>
14     <h6>Encabezado 6</h6>
15     <p>Este es un párrafo de ejemplo para practicar HTML.</p>
16     <p>Otro párrafo con texto adicional.</p>
17 </body>
18 </html>
19
```

---



---

## 3.2.4.- ELEMENTOS DE LISTAS

Hay tres tipos de listas: ordenadas, desordenadas y listas de definición

Elemento	Descripción
<b>ul</b>	Delimita los elementos que forman una lista desordenada
<b>ol</b>	Delimita los elementos que forman una lista ordenada
<b>li</b>	Indica cada uno de los elementos de una lista
<b>dl</b>	Delimita los elementos que forman una lista de definición
<b>dt</b>	Cada uno de los términos que se definen de una lista de definición.
<b>dd</b>	Cada una de las definiciones de una lista de definición.

---

---

## EJEMPLO 3: LISTAS ORDENADAS Y DESORDENADAS

Crear listas ordenadas y desordenadas.

- Crea una lista desordenada (<ul>) de, al menos, tres elementos.
- Crea una lista ordenada (<ol>) de, al menos, tres elementos.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-s<
6      <title>Listas en HTML</title>
7  </head>
8  <body>
9      <h2>Lista Desordenada</h2>
10     <ul>
11         <li>Elemento 1</li>
12         <li>Elemento 2</li>
13         <li>Elemento 3</li>
14     </ul>
15
16     <h2>Lista Ordenada</h2>
17     <ol>
18         <li>Primer paso</li>
19         <li>Segundo paso</li>
20         <li>Tercer paso</li>
21     </ol>
22 </body>
23 </html>
```

---

## 3.2.5. ELEMENTOS DE TABLAS

Los elementos para definir una tabla son los siguientes:

Elemento	Descripción
<b>table</b>	Delimita el contenido de una tabla.
<b>tr</b>	Delimita cada una de las líneas de la tabla.
<b>td</b>	Delimita el contenido de cada celda de la tabla.
<b>colgroup</b>	Permite agrupar columnas.
<b>tbody</b>	Permite agrupar líneas de la tabla.
<b>thead</b>	Define la línea cabecera de la tabla.
<b>th</b>	Delimita cada una de las celdas de la cabecera
<b>tfoot</b>	Define la fila pie de la tabla.

---

---

## EJEMPLO 4: TABLAS SIMPLES

Crear una tabla básica con filas y columnas.

- Crea una tabla con 3 filas y 2 columnas.
- Asegúrate de usar las etiquetas `<table>`, `<tr>`, `<th>` y `<td>`.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1">
6      <title>Tablas en HTML</title>
7  </head>
8  <body>
9      <h2>Tabla de ejemplo</h2>
10     <table border="1">
11         <tr>
12             <th>Nombre</th>
13             <th>Edad</th>
14         </tr>
15         <tr>
16             <td>Juan</td>
17             <td>25</td>
18         </tr>
19         <tr>
20             <td>Ana</td>
21             <td>30</td>
22         </tr>
23     </table>
24 </body>
25 </html>
```

---

## 3.2.6.- ELEMENTOS DE FORMULARIOS

Los elementos que pueden contener un formulario son los siguientes:

Elemento	Descripción
<b>form</b>	Delimita el contenido del formulario
<b>input</b>	Caja de texto para texto corto. Dependiendo del valor que tome el atributo type de este elemento podemos estar ante un texto sin más, un campo de texto donde al escribir no se visualice el contenido si no que escriba asteriscos, un botón de radio que el usuario podrá elegir, una opción que el usuario podrá activar, botones, ...
<b>textarea</b>	Caja de texto para texto largo.
<b>select</b>	Crea un menú desplegable que permite elegir de una lista de opciones que contiene el elemento.
<b>option</b>	Delimita cada una de las opciones de un menú desplegable que le contiene.
<b>button</b>	Permite definir un botón. Su principal ventaja frente a los botones hechos con input es que este elemento permite introducir en el botón cualquier otro elemento de HTML, como por ejemplo imágenes.
<b>fieldset</b>	Permite agrupar elementos de un un formulario.
<b>legend</b>	Permite poner un título al fieldset.
<b>label</b>	Etiqueta de un campo del formulario.

---

---

# EJEMPLO 5: FORMULARIOS BÁSICOS

Crear un formulario con diferentes tipos de campos de entrada.

- Crea un formulario que incluya campos de texto, contraseña, correo electrónico y un botón de envío.

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

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Formulario Básico</title>
</head>

<body>

  <h1>Formulario de Registro</h1>

  <form action="/submit" method="POST">
    <!-- Campo para el nombre -->
    <label for="nombre">Nombre:</label>
    <input type="text" id="nombre" name="nombre" required>
    <br><br>

    <!-- Campo para el correo electrónico -->
    <label for="email">Correo Electrónico:</label>
    <input type="email" id="email" name="email" required>
    <br><br>

    <!-- Campo para la contraseña -->
    <label for="password">Contraseña:</label>
    <input type="password" id="password" name="password" required>
    <br><br>

    <!-- Botón de envío -->
    <button type="submit">Enviar</button>
  </form>

</body>

</html>
```

# Formulario de Registro

Nombre:

Correo Electrónico:

Contraseña:

Enviar

---

# EXPLICACIÓN DETALLADA DE CADA UNA DE LAS ETIQUETAS DE FORMULARIOS

Los **formularios** en HTML son fundamentales para recolectar datos del usuario. Se componen de varias etiquetas que permiten crear campos de entrada, menús desplegables, botones, entre otros.

---



---

## 1.- Etiqueta <form>

La etiqueta <form> define el contenedor del formulario. Es donde comienza el formulario y contiene todos los campos de entrada y botones de envío.

### **Atributos importantes:**

- action: Define la URL a la que se enviarán los datos del formulario.
  - method: Especifica el método HTTP para el envío de datos. Los más comunes son:
    - GET: Los datos se envían en la URL como parámetros (menos seguro y limitado en tamaño).
    - POST: Los datos se envían en el cuerpo de la solicitud (más seguro y sin limitación de tamaño visible).
-

---

Ejemplo de form

```
<form action="/submit-form" method="post">
```

```
<!-- Campos del formulario aquí -->
```

```
</form>
```

- `action="/submit-form"`: Especifica la URL donde se enviarán los datos del formulario. En este caso, es una URL ficticia `/submit-form`. Si fuera una aplicación real, aquí se colocaría la URL de procesamiento del servidor.
  - `method="post"`: Define el método HTTP usado para enviar los datos. El método POST es más seguro y no muestra los datos en la barra de direcciones del navegador. Si utilizáramos GET, los datos se enviarían como parte de la URL.
-

---

## 2.-Etiqueta <input>

Esta es la etiqueta más versátil para crear diferentes tipos de campos de entrada. El tipo de campo se define con el atributo type.

### Atributos importantes:

- type: Define el tipo de campo. Hay muchos tipos disponibles, como text, password, email, file, checkbox, radio, submit, entre otros.
  - name: Especifica el nombre del campo que se envía al servidor.
  - value: Define el valor inicial del campo.
  - placeholder: Texto que aparece como sugerencia dentro del campo (cuando está vacío).
  - required: Indica si el campo es obligatorio.
  - disabled: Desactiva el campo.
  - readonly: Hace el campo solo de lectura.
  - maxlength: Establece el número máximo de caracteres que el usuario puede ingresar.
-

---

# EJEMPLOS DE TIPOS COMUNES DE <INPUT>:

- **Texto:**

`<input type="text" name="nombre" placeholder="Escribe tu nombre" required>`

- **Contraseña:**

`<input type="password" name="clave" placeholder="Escribe tu contraseña">`

- **Email:**

`<input type="email" name="correo" placeholder="Escribe tu correo" required>`

---

---

- **Número**

`<input type="number" name="edad" min="0" max="100">`

- **Checkbox (casilla de verificación)**

`<input type="checkbox" name="aceptar" value="si" required> Acepto los términos y condiciones`

- **Radio (botones de opción)**

`<input type="radio" name="genero" value="hombre"> Hombre`

`<input type="radio" name="genero" value="mujer"> Mujer`

- **Archivo**

`<input type="file" name="archivo">`

- **Botón de enviar**

`<input type="submit" value="Enviar formulario">`

---

---

### 3.- Etiqueta <label>

Asocia un texto descriptivo con un campo de entrada. Mejora la accesibilidad y la experiencia del usuario, ya que el texto del <label> está vinculado al campo de entrada correspondiente.

#### **Atributo importante:**

- for: Relaciona el campo de entrada con el id de un elemento <input>.

---

# EJEMPLO ETIQUETA <LABEL>

Ejemplo:

```
<label for="nombre">Nombre:</label>
```

```
<input type="text" id="nombre" name="nombre" placeholder="Escribe tu nombre" required>
```

- `<label for="nombre">Nombre:</label>`: El elemento `<label>` muestra un texto explicativo que acompaña al campo de entrada. El atributo `for="nombre"` conecta este `<label>` con el campo `<input>` cuyo atributo `id` también es `nombre`. Esto es útil para la accesibilidad, ya que el usuario puede hacer clic en la etiqueta para activar el campo de texto.
  - `<input type="text">`: Define un campo de entrada de texto simple.
    - `id="nombre"`: Identificador único del campo. Se usa con el `<label>` para asociar ambos.
    - `name="nombre"`: Nombre que el servidor utilizará para identificar el campo al recibir los datos del formulario.
    - `placeholder="Escribe tu nombre"`: Texto que aparece dentro del campo como sugerencia cuando el campo está vacío.
    - `required`: Indica que el campo es obligatorio. El formulario no se puede enviar si este campo está vacío.
-

---

## 4.- Etiqueta <textarea>

Se utiliza para crear un área de texto donde el usuario puede escribir varias líneas, ideal para comentarios o descripciones largas.

### Atributos importantes:

- Name: Nombre del campo enviado.
- Rows y cols: Definen el tamaño del área de texto.
- Placeholder: Texto que aparece cuando el área de texto está vacía.

Ejemplo:

```
<textarea name="comentarios" rows="5" cols="40" placeholder="Escribe tus  
comentarios"></textarea>
```



---

## 5.- Etiqueta <select> y <option>

Crea un menú desplegable con opciones predefinidas.

### Atributos importantes:

- name: Nombre del campo que se envía.
- Dentro de <select>, usamos etiquetas <option> para definir cada opción del menú.
- Atributo selected en <option> indica la opción seleccionada por defecto.

Ejemplo:

```
<label for="pais">País:</label>
<select name="pais" id="pais">
    <option value="mx">México</option>
    <option value="es">España</option>
    <option value="us">Estados Unidos</option>
</select>
```

---

---

## 6.- Etiqueta <button>

Define un botón interactivo dentro del formulario.

Atributos importantes:

- Type: Define el tipo de botón:
  - Submit: Envía el formulario.
  - Reset: Restablece los campos del formulario a sus valores iniciales.
  - Button: botón normal que no realiza acciones por defecto.

Ejemplo:

```
<button type="submit">Enviar</button>
```

---

## 7. Etiqueta `<fieldset>` y `<legend>`

Agrupar campos del formulario para organizar mejor los elementos. `<legend>` define un título para el grupo.

- `<fieldset>`: Agrupa varios campos del formulario dentro de una "sección". Visualmente, esto crea un cuadro alrededor de los campos agrupados.
- `<legend>`: Es el título del grupo. Aquí indica que esta sección contiene información personal del usuario.

Ejemplo:

```
<fieldset>
  <legend>Información Personal</legend>
  <label for="nombre">Nombre:</label>
  <input type="text" id="nombre" name="nombre">
  <label for="correo">Correo electrónico:</label>
  <input type="email" id="correo" name="correo">
</fieldset>
```

---

## Ejemplo de formulario: Matriculación en el ASIR

Nombre:  Apellidos:  Sexo: ☐ M ☐ H

Fecha de nacimiento:

Escoge los módulos en los que te matriculas:

- |   |  |
|---|--|
| <input type="checkbox"/> Fundamentos de Hardware                                    | <input type="checkbox"/> Servicios de Red e Internet                           |
| <input type="checkbox"/> Gestión de Bases de Datos                                  | <input type="checkbox"/> Administración de sistemas Operativos                 |
| <input type="checkbox"/> Implantación de Sistemas Operativos                        | <input type="checkbox"/> Implantación de Aplicaciones Web                      |
| <input type="checkbox"/> Planificación y Administración de Redes                    | <input type="checkbox"/> Administración de Sistemas Gestores de Bases de Datos |
| <input type="checkbox"/> Lenguajes de Marcas y Sistemas de Gestión de Información   | <input type="checkbox"/> Seguridad y Alta Disponibilidad                       |
| <input type="checkbox"/> Formación y Orientación Laboral                            | <input type="checkbox"/> Empresa e Iniciativa Emprendedora                     |
| <input type="checkbox"/> Proyecto de Administración de Sistemas Informáticos en Red | <input type="checkbox"/> Formación en Centros de Trabajo                       |

Estudios previos:

Enviar

Restablecer

Pulsa el botón de "enviar" para formalizar la matrícula o el boton de "restablecer" para limpiar el formulario.

---

## 3.2.7 ELEMENTOS DE FRAMES

Los frames se utilizan para dividir una página web en secciones que pueden cargar documentos HTML independientes. Antiguamente se utilizaban los siguientes elementos:

Elemento	Descripción
<code>frameset</code>	Define la partición de la ventana del navegador en marcos. Sólo puede partirse en filas o en columnas. Para partir la ventana del navegador en filas y columnas hay que anidar frames.
<code>Frame</code>	Define un marco que contiene información

El uso de `<frameset>` y `<frame>` se introdujo en versiones anteriores de HTML para dividir una ventana del navegador en varias secciones, cada una con su propio documento.

---

---

# EJEMPLO DE FRAMESET Y FRAME

```
1  <!DOCTYPE html>
2  <html lang="es">
3
4  <head>
5      <meta charset="UTF-8">
6      <title>Ejemplo de frameset</title>
7  </head>
8  <frameset rows="30%, 70%">
9      <!-- El primer frame mostrará la página de ejemplo 1 -->
10     <frame src="https://www.example.com" name="frame1">
11         <!-- El segundo frame mostrará otra página -->
12         <frame src="https://www.wikipedia.org" name="frame2">
13 </frameset>
14
15 </html>
```

---

---

# IFRAMES

Con la llegada de HTML5 se dejó de utilizar `<frameset>` y `<frame>` y se utilizó un nuevo elemento que es `<iframe>`.

Con `<iframe>` realizamos lo mismo que con `<frameset>` y `<frame>`, es decir, se utiliza para incrustar otra página web dentro de la página actual.

```
1  <!DOCTYPE html>
2  <html lang="es">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Ejemplo de iframe</title>
8      <style>
9          body {
10              font-family: Arial, sans-serif;
11          }
12
13          iframe {
14              border: 2px solid #db9834;
15              border-radius: 8px;
16              width: 100%;
17              height: 500px;
18          }
19      </style>
20  </head>
21
22  <body>
23
24      <h1>Ejemplo de iframe</h1>
25      <p>A continuación se muestra una página web incrustada con un iframe:</p>
26
27      <!-- Incrustando Moodle Centros en un iframe -->
28      <iframe src="https://www.juntadeandalucia.es/educacion/eaprendizaje/moodle-centros/"
29          title="Incrustación de moodle-centros"></iframe>
30
31  </body>
32
33  </html>
```

---

## 3.2.8. CAPAS

- Las **capas** se pueden definir como páginas que se pueden incrustar dentro de otras, es decir, que son bloques con contenido HTML que pueden situarse en la página de manera dinámica y organizables en 3D.
  - Los **atributos de una capa** (posición, visibilidad, etc..), como los de cualquier otro elemento HTML, pueden y deben definirse dentro de una hoja de estilo. Realmente tienen sentido cuando se les aplican estilos CSS, pues es cuando sus ventajas se pueden aprovechar al 100%. Es quizás la mejor ilustración de lo que significa separar contenido de presentación. Gracias a las capas o bloques es posible hacer que una página web se adapte al dispositivo desde el que se visualiza.
  - Su **contenido**, en cambio, siempre deberá ser especificado dentro de la parte principal de la página. Como se puede ver, se está siguiendo la filosofía de separar el contenido y la forma de representarlo.
-



- 
- Sin duda, lo más importante de las capas es la posibilidad que presentan de ser movidas y modificadas desde un lenguaje de *script*. Desgraciadamente, las implementaciones de los distintos navegadores pueden ser incompatibles entre sí, por lo que resulta complicado escribir código que funcione de la misma forma en distintas plataformas, de ahí que un buen diseñador y programador web verifique su web en las distintas plataformas existentes y se asegure de que los resultados sean similares.
  - La única manera común de definir capas para distintos navegadores es mediante **hojas de estilo**. En realidad, las capas no se definen completamente mediante el lenguaje HTML, necesitan del lenguaje de definición de estilos CSS. Entre ambos podemos colocar en nuestras páginas estos elementos movibles, ocultables y en general manipulables de forma dinámica.
-

Elemento	Descripción
div	Permite crear una sección dentro del documento XHTML.
id	Permite asignar un identificador único al elemento <i>div</i> .
style	Permite dar un estilo al contenido del elemento empleando las h
class	Permite dar un estilo al contenido del elemento empleando las h

---

## 3.2.8. OTROS ELEMENTOS

Elemento	Descripción
<b>a</b>	Permite definir un enlace a una página web, un archivo o una dirección de correo.
<b>img</b>	Permite insertar una imagen en una página web. Es obligatorio utilizar el atributo src para determinar el path del fichero de imagen que queremos insertar.
<b>abbr</b>	Indica una forma abreviada.
<b>acronym</b>	Indica un acrónimo.
<b>blockquote</b>	Contiene un bloque de texto con sangría.
<b>q</b>	Contiene una cita por lo que el navegador le añade las marcas de citación
<b>br</b>	Inserta una línea en blanco. No tiene etiqueta de cierre, sólo se abre.

---

---

# EJEMPLO 6: IMÁGENES

- 1.- Descarga una imagen de ejemplo o utiliza una url de una imagen en línea.
- 2.- Inserta la imagen en una página utilizando la etiqueta <img>.
- 3.- Agrega un texto alternativo (atributo alt) a la imagen.

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Imágenes en HTML</title>
7  </head>
8  <body>
9      <h2>Imagen de ejemplo</h2>
10     
11 </body>
12 </html>
13
```

---

---

# EJEMPLO 7: ENLACES

- 1.- Crea un enlace que lleve a cualquier sitio web.
- 2.- Asegúrate de que el enlace abra una nueva pestaña utilizando el atributo target="\_blank".

```
1  <!DOCTYPE html>
2  <html lang="es">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Enlaces en HTML</title>
8  </head>
9
10 <body>
11     <p>Visita <a href="https://www.juntadeandalucia.es/educacion/eaprendizaje/moodle-centros/" target="_blank">
12         Moodle Centros</a> para entrar en el modulo de la asignatura.</p>
13 </body>
14
15 </html>
```

## Ejemplo de uso de enlaces

[Clickeame](#)

## Ejemplo de uso de imagenes

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi primera página HTML</title>
7  </head>
8  <body>
9      <p>¡Hola, Mundo! Esta es mi primera página en HTML.</p>
10 </body>
11 </html>
```

## Ejemplo de uso de abreviaturas

El [ASIR](#) comenzó a impartirse en Septiembre de 2010.

## Ejemplo de uso de acrónimos

El protocolo web más usado es el [HTTP](#).

## Ejemplo de uso de citas

La frase “No hay nada repartido de modo más equitativo que la razón: todo el mundo está convencido de tener suficiente.” fué dicha por René Descartes (1596-1650) Filósofo y matemático francés.

## Ejemplo de uso de imágenes con enlaces

```
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Mi primera página HTML</title>
7  </head>
8  <body>
9      <p>¡Hola, Mundo! Esta es mi primera página en HTML.</p>
10 </body>
11 </html>
```

Pulsa en la imagen



---

## 4.- XHTML FRENTE A HTML

- El lenguaje XHTML es muy similar a HTML. No es más que una adaptación de HTML al lenguaje XML.
  - El lenguaje HTML tiene una sintaxis muy permisiva, por lo que es posible escribir sus etiquetas y atributos de muchas formas diferentes. Las etiquetas, por ejemplo, podían escribirse en mayúsculas, en minúsculas e incluso combinando mayúsculas y minúsculas. El valor de los atributos de las etiquetas se puede indicar con o sin comillas. Además, el orden en el que se abrían y cerraban las etiquetas no era importante.
  - La flexibilidad de HTML da lugar a páginas con un código desordenado, difícil de mantener y muy poco profesional.
  - XHTML soluciona estos problemas añadiendo ciertas normas en la forma de escribir las etiquetas y atributos.
-

---

## 4.1.- XHTML: DIFERENCIAS SINTÁCTICAS Y ESTRUCTURALES CON HTML.

El esquema básico del documento, para considerarse conforme a la especificación deberá cumplir las siguientes condiciones:

El elemento raíz del documento debe indicar el espacio nominal XHTML usando el atributo xmlns. El espacio nominal para XHTML es <http://www.w3.org/1999/xhtml>

Debe haber una declaración DOCTYPE en el prólogo del documento. El identificador público incluido en la declaración DOCTYPE debe hacer referencia a alguna de las tres DTD definidas por el W3C usando el Identificador Formal Público correspondiente:

---



---

## 4.1.1. ESTRUCTURA BÁSICA DE UN DOCUMENTO XHTML

- La estructura básica de cualquier página o documento XHTML válido incluye al menos las siguientes etiquetas:
    - `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
    - `<html xmlns="http://www.w3.org/1999/xhtml" lang="es" xml:lang="es">`
    - `<head>`
      - `<title>...</title>`
      - ...
      - `</head>`
      - `<body>`
        - ...
        - `</body>`
        - `</html>`
  - EL DOCTYPE también puede ser: `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
-

---

## 4.2.- VENTAJAS E INCONVENIENTES DE XHTML SOBRE HTML

### **Ventajas:**

- Compatibilidad parcial con navegadores antiguos: la información se visualiza, aunque sin formato.
  - Un mismo documento puede adoptar diseños radicalmente distintos en diferentes apartados.
  - Sencillez a la hora de editar y mantener el código.
  - Es compatible con los estándares que está desarrollando el W3C como recomendación para futuros agentes de usuario o navegadores.
  - Los documentos escritos conforme a XHTML 1.0 presentan mejor rendimiento en las actuales herramientas web que aquellos escritos conforme a HTML.
  - La separación de los contenidos y su presentación hace que los documentos XHTML se adapten mejor a las diferentes plataformas: pantallas de ordenador, pantallas de dispositivos móviles, ...
  - Como es XML se pueden utilizar fácilmente herramientas creadas para procesar documentos XML genéricos (editores, XSLT, etc.).
-

---

## 5.- HERRAMIENTAS DE DISEÑO WEB

El universo de las herramientas de diseño web es cada vez más amplio. Sobre todo, por los avances tecnológicos que propician el surgimiento de nuevas plataformas con innovadoras funciones. Hay algunas que destacan por las facilidades, optimización y el ahorro de tiempo que brindan.

Sin embargo, por esa misma razón, a veces es complicado saber con exactitud cuál es la mejor herramienta de diseño web, o por lo menos, cuál es la que necesitas en función de tu nivel de conocimientos y la página que deseas crear.

Herramientas de edición de código, software de diseño web y recursos que son muy útiles en la actualidad

---

---

**Adobe Color CC:** La aplicación de Adobe Color CC mejora el color de tus fotografías, ya que analiza el tono de las ilustraciones y busca soluciones armónicas. Así, ayuda a la combinación de paletas cromáticas de forma armoniosa para cada uno de tus proyectos. No importa qué tipo de sitio web diseñes: Adobe Color CC logrará hacerlo ver atractivo con imágenes perfectamente equilibradas.

Esta herramienta te permite realizar ajustes de color, tono y saturación, curvas y máscaras de luminosidad. Si no tienes claro cómo combinar los colores, aquí tendrás una gama de tonalidades a partir de una base.

Es gratuita y fácil de usar. Está disponible en dos aplicaciones de escritorio de Creative Cloud: Adobe InDesign y Adobe After Effects; también puedes acceder a través de Photoshop. Si requieres una aplicación móvil, la encuentras como Adobe Capture CC.

---

- 
- **Google Fonts:** Una página web completa, además de ser agradable y estar optimizada, debe contar con diversos elementos que la hagan resaltar. Uno de los más importantes es la tipografía. Google Fonts te garantiza que tu fuente será original y adecuada para la imagen corporativa o el estilo que imaginas, en tanto que aquí encuentras un catálogo gratuito con más de 800 opciones. Y si necesitas un poco de inspiración, también puedes visitar su sección Featured donde conocerás muchas colecciones temáticas creadas por la misma herramienta.
  - Es muy intuitiva y visual. Dentro de su aplicación encontrarás filtros de búsquedas que puedes determinar por categorías y opciones de prueba para elegir la que mejor se adapte a tu proyecto.
-

---

**Whatfontis:** Muchas fuentes pueden ser muy conocidas, pero entre tantas que se crean día con día (hay unas 500 mil registradas en el mundo), cada vez es más complicado identificarlas. Por fortuna, en la actualidad ya no tienes que quedarte con la duda sobre cuál fuente viste, ya que con a WhatFontIs podrás saberlo.

Esta herramienta es la más potente del mercado gracias a su sencillez, rapidez y capacidad de reconocimiento. Su sistema es muy simple, ya que solo basta con que subas una imagen o URL a su sitio. Si la imagen que guardaste no tiene una resolución baja, podrás editarla rápidamente y obtener mejores resultados.

Reconoce hasta 460.000 tipografías diferentes. Además, dentro de su sitio encontrarás una comunidad de usuarios a la que podrás acudir cuando quieras.

---

- 
- **Mockflow:** La herramienta Mockflow es perfecta para construir esquemas de diseño de forma más rápida. Lo único que tienes que hacer es arrastrar y soltar para colocar los elementos con iconos preconfigurados. Te apoyará en la creación de una página web funcional y en la planificación de interfaces de usuario.

Cuenta con una tienda de plantillas para que encuentres inspiración por medio del trabajo de otros diseñadores. Puedes utilizarla también desde la aplicación móvil.

- **Tinypng:** Tinypng es la mejor forma de optimizar tus imágenes: su función es esencial pues comprime tus contenidos gráficos. Esto es necesario para que, al momento de navegar en una página web, las imágenes no hagan más lento el tiempo de carga.

Su uso es fácil y optimiza las imágenes de forma automática.

---

- 
- **Notepad++:** Si prefieres diseñar sin un sistema de gestión de contenidos o CMS, entonces esta se convertirá en tu herramienta de cabecera. Con Notepad++ podrás despedirte de tu bloc de notas, ya que en esta aplicación editarás los códigos fuente. Identifica con su ayuda las expresiones de lenguaje utilizado y resalta las combinaciones de colores para facilitar la lectura. Optimiza las rutinas y mantiene un uso sencillo.
  - **Canva:** Sus funciones son más sencillas en comparación a otras herramientas de diseño web, pero sin duda, igual de efectivas. Canva es la plataforma estrella en la creación de imágenes, cabeceras para blogs y páginas web. Considerando que es lo primero que verán tus usuarios, se ha vuelto una herramienta imprescindible a la hora de armonizar el diseño de tu web.

Además, también se ha vuelto una plataforma aliada a la hora de crear publicaciones para las redes sociales. Y el plus de esta herramienta es que no tienes que ser un experto en diseño gráfico, su interfaz es muy sencilla, gratuita y online.

---