

Fundamentos de Ingeniería Informática

Módulo V – Unidad 1 Introducción al desarrollo web

Profesor: Héctor Molina García

¿Qué es una página web?

Introducción a la programación web

Una **página web** es un documento de hipertexto en la World Wide Web. Las páginas web son entregadas por un servidor web al usuario y se muestran en un navegador web.

World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#), [Policy](#), November's [W3 news](#), [Frequently Asked Questions](#).

[What's out there?](#)

Pointers to the world's online information, [subjects](#), [W3 servers](#), etc.

[Help](#)

on the browser you are using

[Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#), [X11 Viola](#), [NeXTStep](#), [Servers](#), [Tools](#), [Mail robot](#), [Library](#))

[Technical](#)

Details of protocols, formats, program internals etc

[Bibliography](#)

Paper documentation on W3 and references.

[People](#)

A list of some people involved in the project.

[History](#)

A summary of the history of the project.

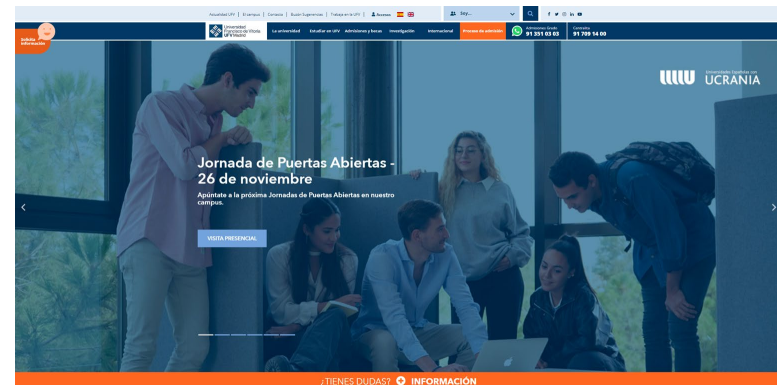
[How can I help?](#)

If you would like to support the web..

[Getting code](#)

Getting the code by [anonymous FTP](#), etc.

<http://info.cern.ch/hypertext/WWW/TheProject.html>



<https://www.ufv.es/>

La primera página web fue creada en el CERN por [Tim Berners-Lee](#) el 6 de agosto de 1991.

Introducción a la programación web

¿Cuál es la diferencia entre un sitio web y una página web?

Un sitio web se refiere a una ubicación central que contiene más de una página web. Por ejemplo, UFV se considera un sitio web, que incluye docenas de páginas web diferentes.

<https://www.ufv.es/la-universidad/sobre-ufv/>

				
Protocol	Subdomain	domain	directories	web page

La página web es siempre la última parte de la URL. Actualmente los sitios web ocultan el nombre del archivo porque el contenido generalmente se genera dinámicamente.

Introducción a la programación web

¿Cuál es la diferencia entre un sitio web y una aplicación web?

Un **sitio web** solo muestra información estática, y los usuarios solo pueden desplazarse, mirar el contenido y seguir los enlaces.

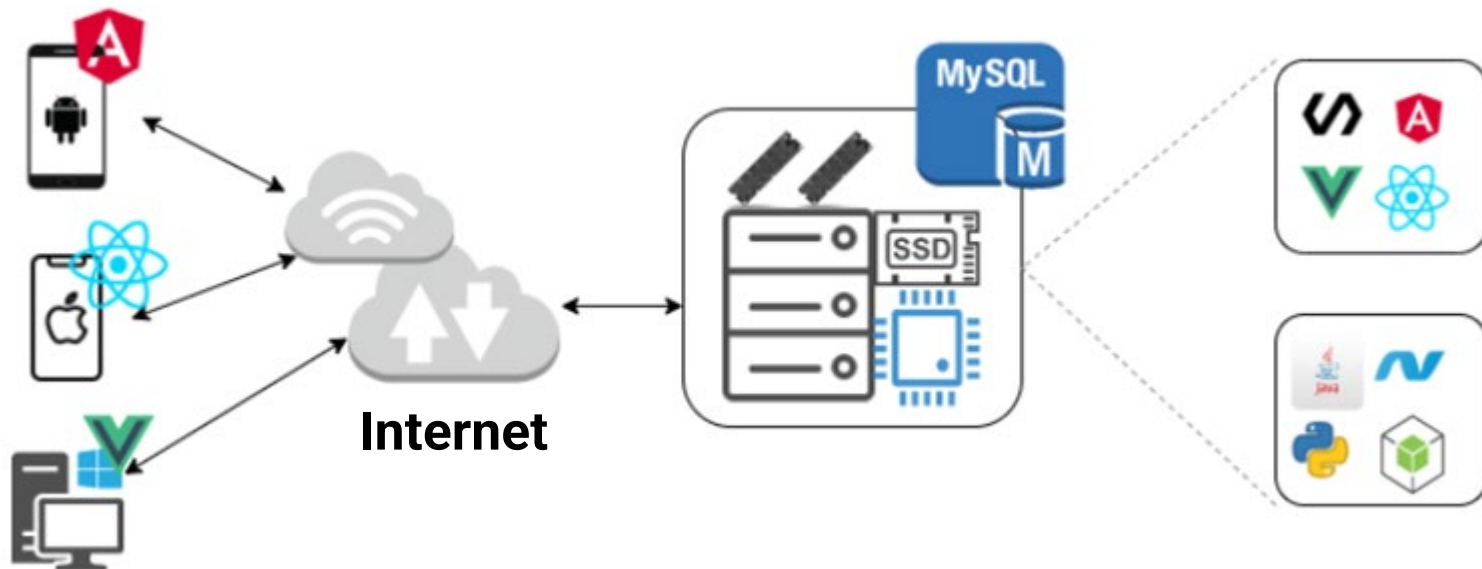
Una aplicación web (web app) es un producto más complejo que permite muchos tipos de interacciones y puede usar API para dar a los usuarios acceso a servicios de terceros. A través de una aplicación web, los usuarios pueden completar tareas como realizar y pagar pedidos, cargar documentos y acceder a análisis.



Introducción a la programación web

Arquitectura cliente-servidor

La arquitectura cliente-servidor se refiere a un sistema que aloja, entrega y administra la mayoría de los recursos y servicios que el cliente solicita.

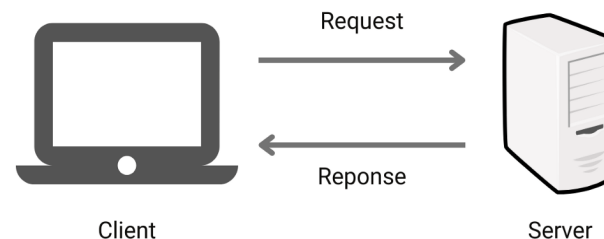


Introducción a la programación web

Arquitectura cliente-servidor

Una arquitectura cliente-servidor operativa ejecuta las siguientes operaciones:

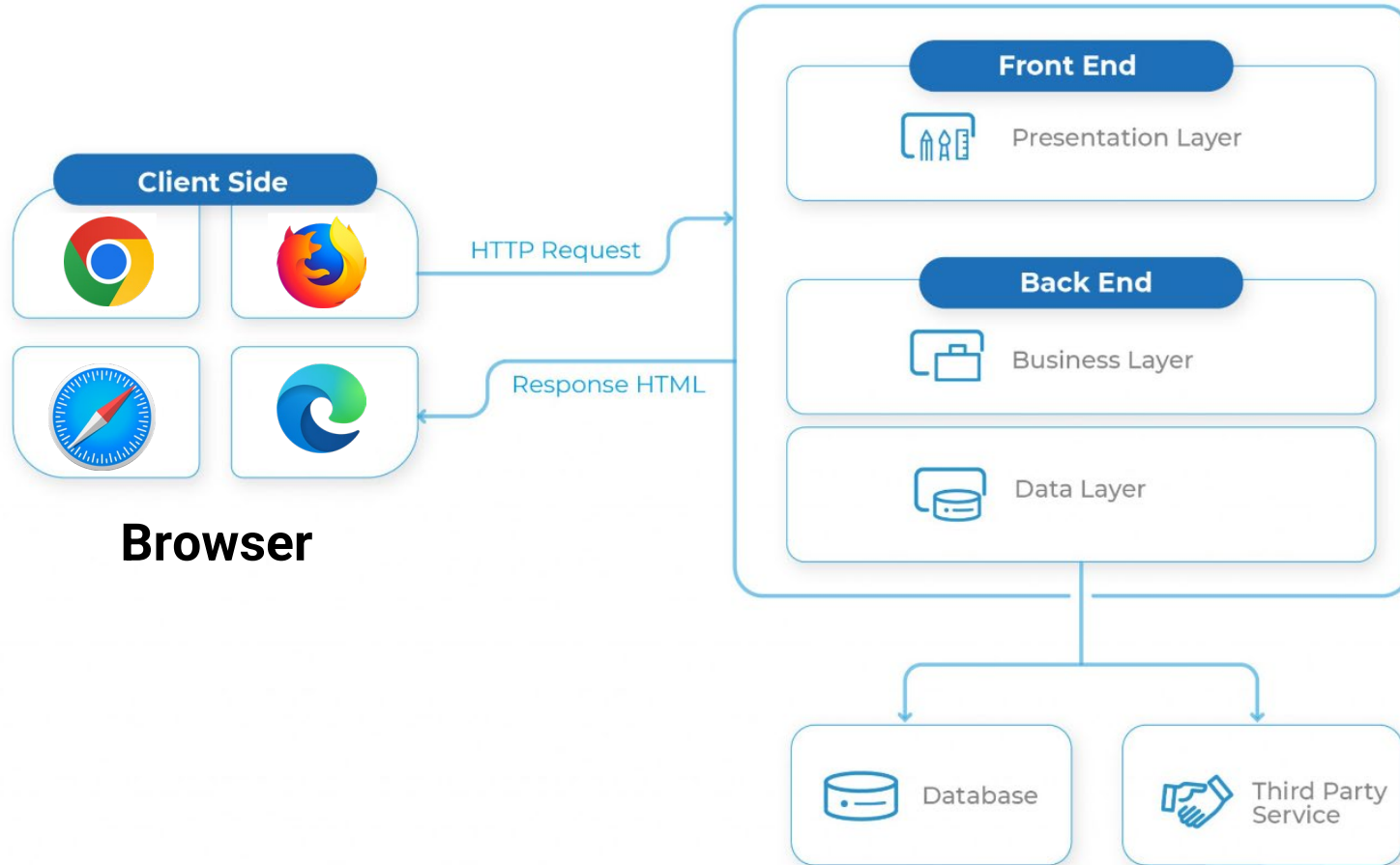
- El cliente solicita una página web a través de un dispositivo habilitado para la red.
- El servidor web recibe y procesa la solicitud.
- El servidor web envía una respuesta al cliente.



¿Qué podemos construir una aplicación web?

Introducción a la programación web

¿Cómo podemos construir una aplicación web?



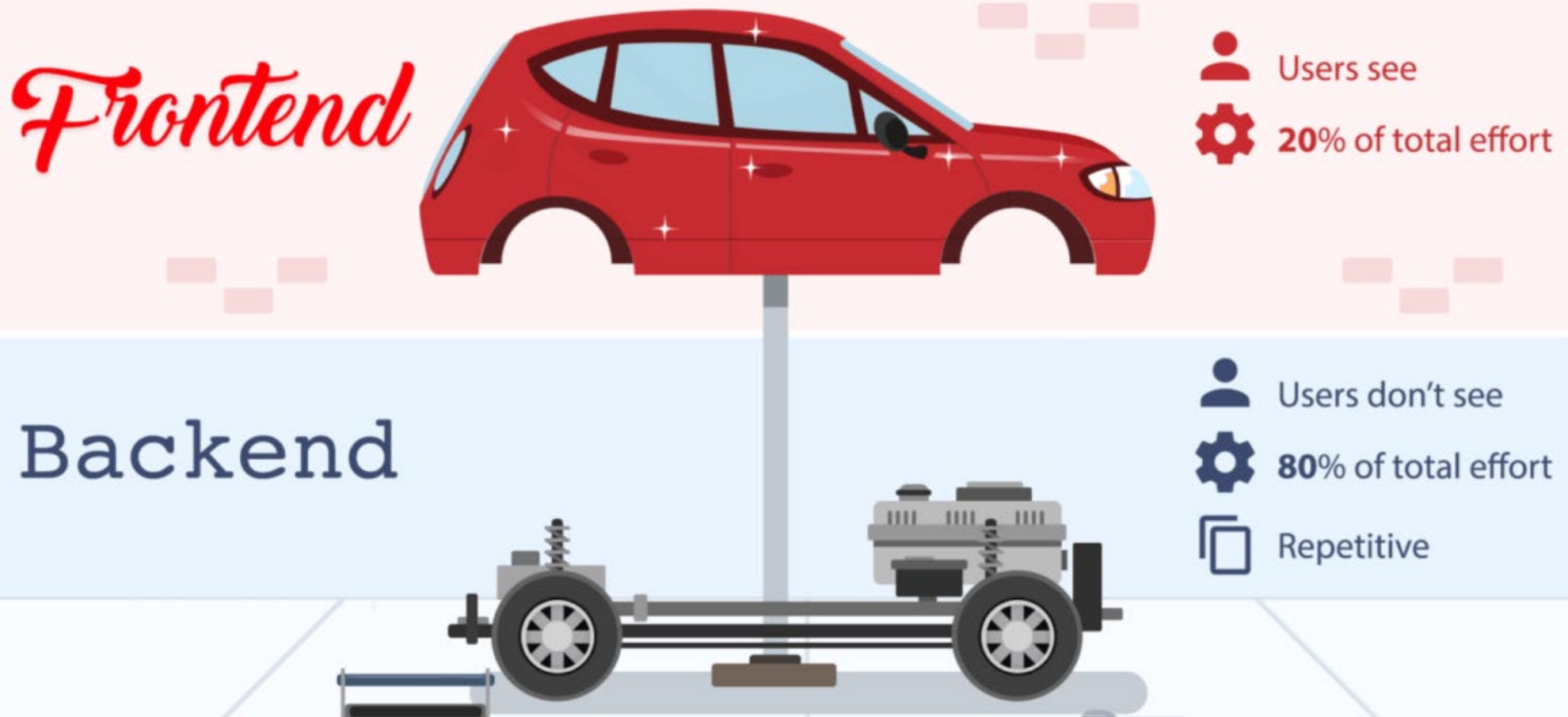
Introducción a la programación web

¿Cómo podemos construir una aplicación web?



Introducción a la programación web

¿Cómo podemos construir una aplicación web?



Introducción a la programación web

¿Cómo podemos construir una aplicación web?

Backend Vs Frontend Who's The Boss?



Introducción a la programación web

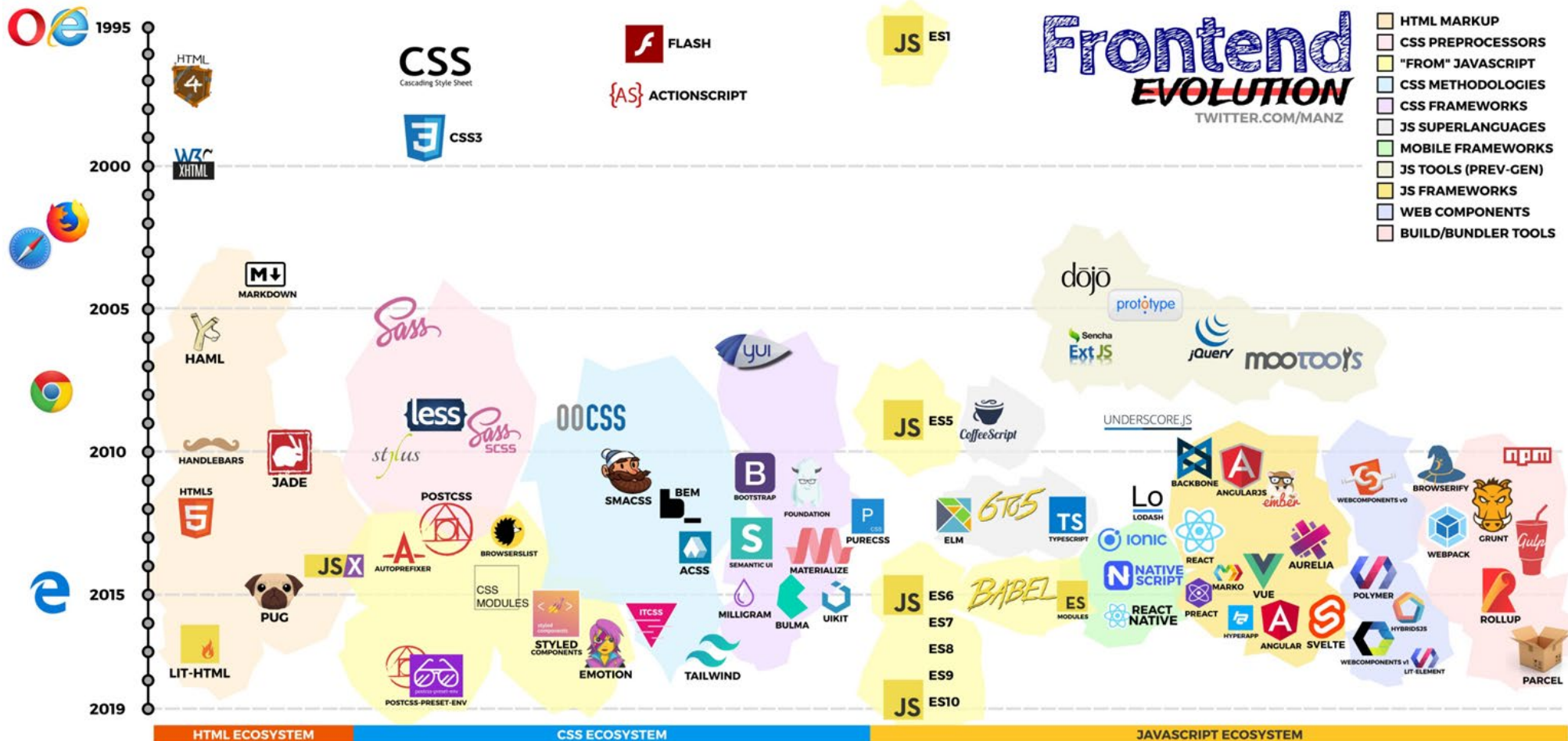
Desarrollo Front End



Las tecnologías frontend abarcan los lenguajes de programación interpretados por el navegador. Esto significa que los archivos se procesan en el si (cliente) del usuario. Los sitios web que solo tienen una capa frontend se denominan aplicaciones web estáticas.

Introducción a la programación web

Desarrollo Front End



HTML (HyperText Markup Language)

HTML es el lenguaje de marcado estándar para documentos diseñados para mostrarse en un navegador web. Puede ser asistido por tecnologías como hojas de estilo en cascada y lenguajes de scripting como JavaScript.

- **HyperText:** HyperText simplemente significa "Texto dentro del texto". Un texto tiene un enlace dentro de él, es un hipertexto. El hipertexto es una forma de vincular dos o más páginas web (documentos HTML) entre sí..
- **Markup Language:** es un lenguaje informático que se utiliza para aplicar convenciones de diseño y formato a un documento de texto.

CSS (hojas de estilo en cascada)

Cascading Style Sheets (CSS) es un lenguaje de hojas de estilo utilizado para describir la presentación de un documento escrito en HTML o XML (incluidos dialectos XML como SVG, MathML o XHTML).

- CSS proporciona una vista mucho mejor de nuestra página HTML en comparación con los atributos HTML.
- Los estilos se aplican como reglas en cascada a cada aparición de ese elemento. Por lo tanto, el código corto significa tiempos de descarga de alta velocidad.
- Los cambios se actualizan automáticamente en todas las páginas web. No necesitamos cambiar html.
- CSS permite contenido que funciona en todos los navegadores web.

Introducción a la programación web

JavaScript (JS)

JavaScript, a menudo llamado JS, es un lenguaje de programación imperativo, ligero, interpretado (use una máquina virtual) o compilado justo a tiempo con funciones orientadas a objetos y de primera clase. Es una de las tecnologías centrales de la World Wide Web, junto con HTML y CSS..

- El 98% de los sitios web actuales utilizan JS en el lado del cliente, a menudo incorporando bibliotecas de terceros.
- Todos los principales navegadores web tienen un motor JS dedicado para ejecutar el código en el lado de los usuarios..

Aunque Java y JavaScript son similares en nombre, sintaxis y respectivas bibliotecas estándar, ambos lenguajes son distintos y difieren enormemente en diseño.

¿Y ahora qué?

¿Y ahora qué?

El siguiente bloque de la asignatura consiste en::

- Conceptos básicos sobre desarrollo web.
- HTML - Cómo funciona.
- CSS - Cómo funciona.

Trabajo individual

- Diseño e implementación de un sitio web

Fundamentos de Ingeniería Informática

Módulo V – Unidad 2 HTML

Profesor: Héctor Molina García

Conceptos básicos de HTML

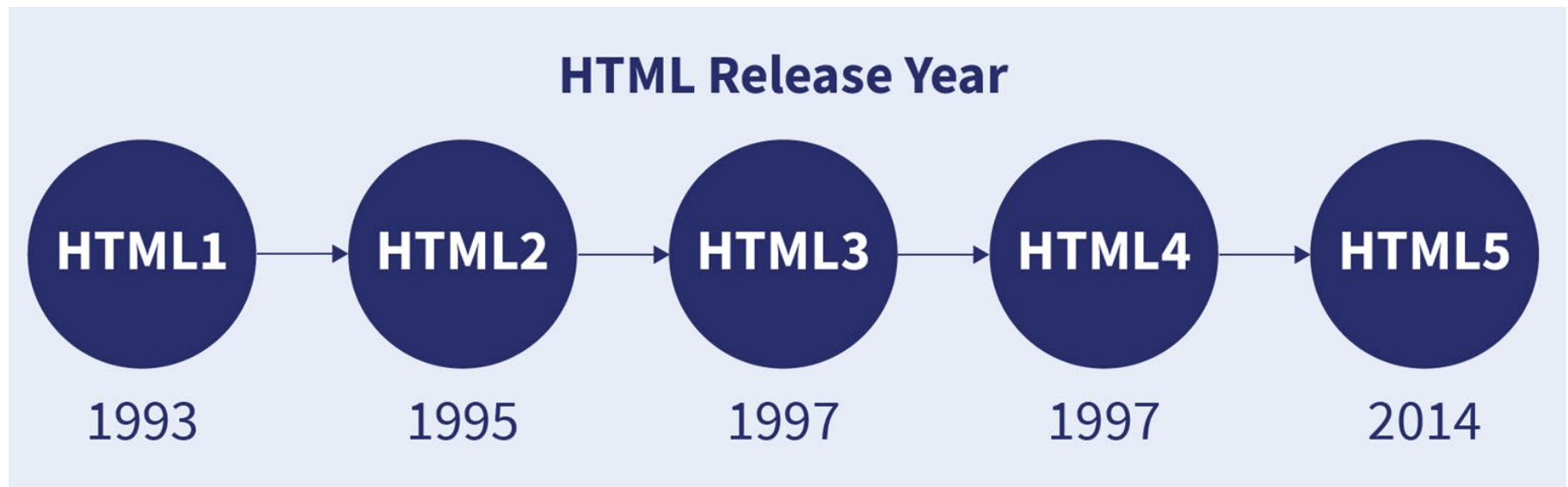
Conceptos básicos de HTML

The **HyperText Markup Language** or HTML es el lenguaje estándar para documentos diseñados para mostrarse en un navegador web. Puede ser asistido por tecnologías como las hojas de estilo en cascada y lenguajes de scripting como JavaScript.

- **HyperText:** HyperText simplemente significa "Texto dentro del texto". Un texto tiene un enlace dentro de él, es un hipertexto. El hipertexto es una forma de vincular dos o más páginas web (documentos HTML) entre sí.
- **Markup Language:** A markup language es un lenguaje informático que se utiliza para aplicar convenciones de diseño y formato a un documento de texto.

Conceptos básicos de HTML

HTML es un lenguaje de marcado utilizado por el navegador para manipular texto, imágenes y otro contenido, con el fin de mostrarlo en el formato requerido. HTML fue creado por Tim Berners-Lee en 1991.

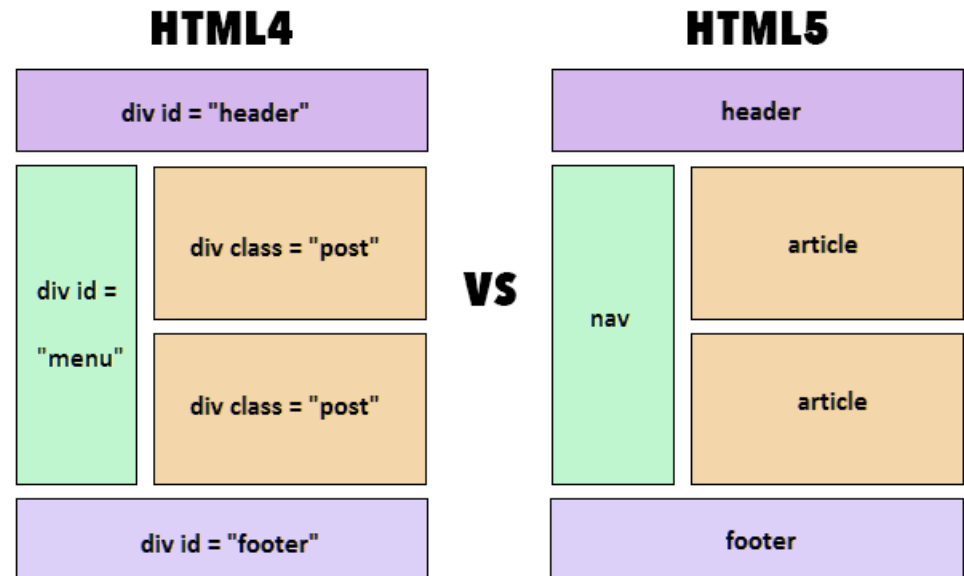


Conceptos básicos de HTML

¿Cuál es la diferencia entre HTML5 y HTML4?

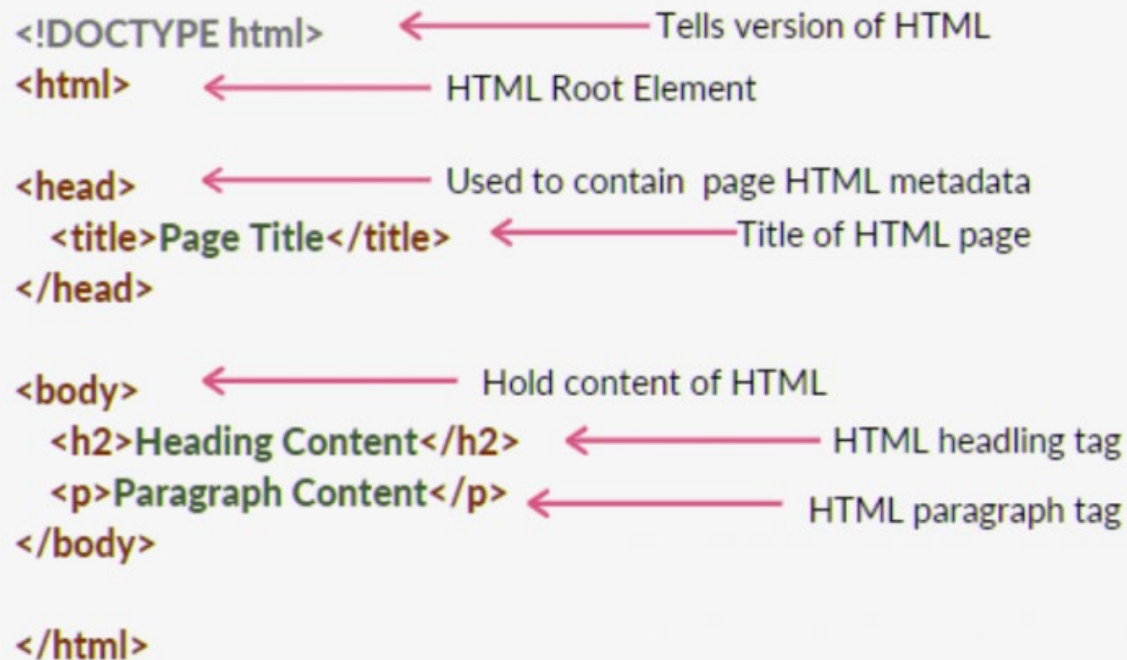
HTML significa HyperText Markup Language y HTML5 es la última versión de HTML Language. Es un marcado más limpio y un código mejorado para los desarrolladores.

- Nuevos elementos semánticos: `<header>`, `<footer>`, `<main>`, `<section>`, `<article>`, `<figure>`, `<aside>` etc `<nav>`.
- Nuevos atributos de los elementos del formulario: número, fecha, hora, calendario y rango, etc.
- Nuevos elementos de formulario: `<datalist>` y `<output>`
- Nuevos elementos gráficos: `<svg>` y `<canvas>`
- Nuevos elementos multimedia: `<audio>`, `<source>`, `<track>`, y `<video>`



Conceptos básicos de HTML

La estructura básica de una página HTML se compone de diferentes bloques (es decir, declaración doctype, HTML, head, title y elementos del cuerpo) que definen la estructura de la página web.



The diagram illustrates the basic structure of an HTML document with the following components and annotations:

- `<!DOCTYPE html>` ← Tells version of HTML
- `<html>` ← HTML Root Element
- `<head>` ← Used to contain page HTML metadata
 - `<title>Page Title</title>` ← Title of HTML page
- `</head>`
- `<body>` ← Hold content of HTML
 - `<h2>Heading Content</h2>` ← HTML heading tag
 - `<p>Paragraph Content</p>` ← HTML paragraph tag
- `</body>`
- `</html>`

Conceptos básicos de HTML

Hay cuatro etiquetas principales que debemos incluir en cualquier página web:

- **DOCTYPE html** (opcional): Esta es la declaración de tipo de documento (técnicamente no es una etiqueta). Declara un documento como un documento HTML.
- **html**: Esta etiqueta se denomina elemento raíz HTML. Todos los demás elementos de la página web están contenidos en ella.
- **head**: Esta etiqueta define el encabezado que contiene los elementos "detrás de escena" para una página web. Los elementos dentro de la cabeza no son visibles en el front-end de una página web.
- **body**: Esta etiqueta define el contenido que se utiliza para encerrar todo el contenido visible de una página web. En otras palabras, el contenido del cuerpo es lo que el navegador mostrará en el front-end.

Conceptos básicos de HTML

El `<head>` es un contenedor de metadatos (datos sobre datos) y se coloca entre la etiqueta `<html>` y la `<body>` etiqueta. Los metadatos son datos que no se muestran sobre el documento HTML. Los siguientes elementos pueden ir dentro del `<head>` elemento:

- `<title>` (required in every HTML document)
- `<style>`
- `<base>`
- `<link>`
- `<meta>`
- `<script>`
- `<noscript>`

Conceptos básicos de HTML

La etiqueta **<body>** define el cuerpo del documento. El elemento body contiene todo el contenido de un documento HTML, como encabezados, párrafos, imágenes, hipervínculos, tablas, listas, etc.

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- The above meta characteristics make a website compatible with different devices. -->
  <title>Demo Web Page</title>
</head>

<body>

  <h1>First title</h1>
  <p>Hello world!!!</p>

</body>
</html>
```

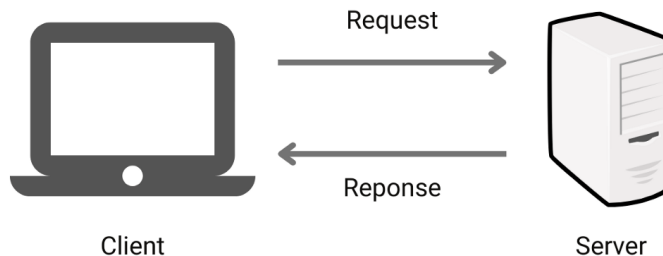
Importante: Solo puede haber un <body> elemento en un documento HTML.

Conceptos básicos de HTML

Arquitectura cliente-servidor

Una arquitectura cliente-servidor operativa ejecuta las siguientes operaciones:

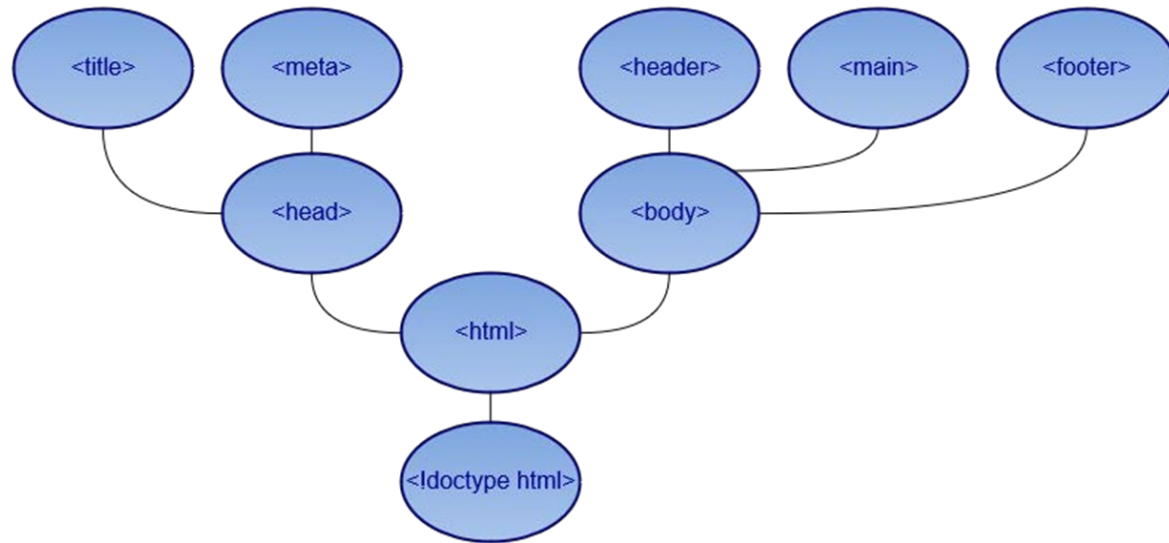
- El cliente solicita una página web a través de un dispositivo habilitado para la red.
- El servidor web recibe y procesa la solicitud.
- El servidor web envía una respuesta (código html) al cliente para mostrarla en el navegador.



El modelo de objetos de documento DOM

DOM

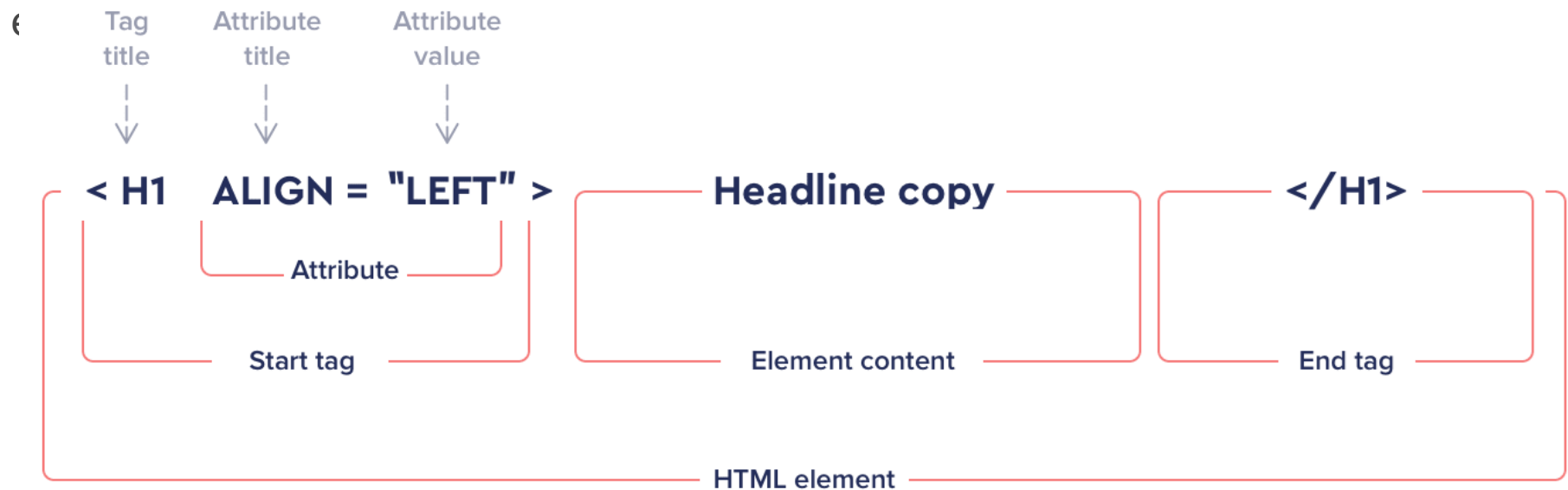
El Document Object Model (DOM) es una interfaz multiplataforma e independiente del lenguaje que trata un documento XML o HTML como una estructura de árbol en la que cada nodo es un objeto que representa una parte del documento..



El DOM representa un documento html con un árbol lógico. Cada rama del árbol termina en un nodo, y cada nodo contiene objetos.

Etiquetas HTML

Las etiquetas HTML son como palabras clave que definen cómo el navegador web formateará y mostrará el contenido. Las etiquetas HTML contienen tres partes principales: etiqueta de apertura, contenido y etiqueta de cierre. Pero algunas



Textos

Existen diferentes etiquetas para incluir texto y manipuladas. Estos son algunos ejemplos:

Un párrafo está marcado de la siguiente manera:

`<p>Este es un texto en un párrafo.</p>`

Podemos manipular palabras usando algunas etiquetas:

`` Etiqueta en negrita ``

`<i>` Etiqueta en cursiva `</i>`

`<u>` Subrayar etiqueta `</u>`

Podemos insertar saltos de línea individuales en un texto:

`<p>Para forzar
 saltos
 de línea en un texto,
 utilice el elemento br
.</p>`

Listas

Las listas HTML permiten a los desarrolladores web agrupar un conjunto de elementos relacionados en listas. Hay 2 tipos de listas: (1) sin ordenar; y (2) ordenado.

Listas

Las listas HTML permiten a los desarrolladores web agrupar un conjunto de elementos relacionados en listas. Hay 2 tipos de listas: (1) sin ordenar; y (2) ordenado.

Una lista desordenada comienza con la `` etiqueta. Cada elemento de lista comienza con la `` etiqueta.

Una lista HTML desordenada:

- Café
- Té
- Leche

```
<p>Una lista HTML desordenada:</p>
```

```
<ul>
```

```
<li>café</li>
```

```
<li>té</li>
```

```
<li>leche</li>
```

```
</ul>
```

Listas

Las listas HTML permiten a los desarrolladores web agrupar un conjunto de elementos relacionados en listas. Hay 2 tipos de listas: (1) sin ordenar; y (2) ordenado.

Una lista ordenada comienza con la `` etiqueta. Cada elemento de lista comienza con la `` etiqueta.

Una lista HTML ordenada:

1. Café
2. Té
3. Leche

```
<p>Una lista HTML ordenada:</p>  
<ol>  
    <li>café</li>  
    <li>té</li>  
    <li>leche</li>  
</ol>
```

Tablas

Las tablas HTML permiten a los desarrolladores web organizar los datos en filas y columnas.

```
<table>
  <tr>
    <th>estudiante</th>
    <th>grado</th>
    <th>país</th>
  </tr>
  <tr>
    <td>Diego Romero</td>
    <td>informática</td>
    <td>Alemania</td>
  </tr>
  <tr>
    <td>Marta Cifuentes</td>
    <td>Negocios y Tecnología</td>
    <td>España</td>
  </tr>
</table>
```

Student	Degree	Country
Diego romero	Computer Science	Germany
Marta Cifuentes	Francisco Chang	Spain

Links

La etiqueta `<a>` define un hipervínculo, que se utiliza para enlazar de una página a otra. El atributo más importante del `<a>` elemento es el atributo `href`, que indica el destino del enlace. Los enlaces tienen un estilo general que define cómo se muestran en todos los navegadores:

- Un enlace no visitado está subrayado y azul
- Un enlace visitado está subrayado y morado
- Un enlace activo está subrayado y en rojo

```
<a href="https://www.ufv.es">Visit Universidad Francisco de Vitoria</a>
```

Imágenes

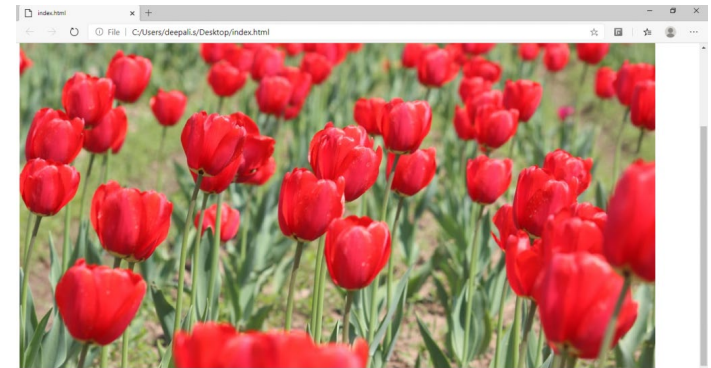
La etiqueta `` se utiliza para añadir una imagen en una página HTML. Esta etiqueta tiene un atributo especial llamado `src` que especifica la ruta de la imagen.

```

```

Podemos especificar la URL en el atributo `src` de las dos maneras siguientes:

- URL absoluta: ruta de una imagen externa que está alojada en otro sitio web.
- URL relativa: ruta de la imagen que está alojada dentro del sitio web.



La etiqueta `` también contiene los atributos `width` y `height`. Como su nombre indica, estos atributos especifican el ancho y el alto de la imagen en píxeles.).

Atributos HTML

Los atributos HTML son palabras especiales que proporcionan información adicional sobre los elementos o atributos que son el modificador del elemento HTML. Cada elemento o etiqueta puede tener atributos (par id:value).



Atributo HTML

Los atributos más utilizados que se pueden utilizar en la mayoría de las etiquetas HTML son:

- Id se utiliza para identificar de forma única cualquier elemento dentro de una página HTML para aplicar estilos o comportamientos.
- El título se utiliza para incluir un título en el atributo. A menudo se muestra como información sobre herramientas cuando el cursor pasa por encima del elemento o mientras se carga el elemento.
- La clase se utiliza para asociar un elemento con una hoja de estilos.
- El estilo se utiliza para especificar reglas de hoja de estilos en cascada (CSS) dentro del elemento.

Formularios HTML

Formularios HTML

Un formulario HTML (o formulario web) es un elemento HTML donde los usuarios pueden insertar datos o información personal para ejecutar un proceso en la página web. Por ejemplo, los usuarios pueden compartir su nombre y dirección de correo electrónico para registrarse en una página web con el fin de acceder a algunos servicios. Los formularios web se utilizan de varias maneras, como:

- Recopilar información del usuario para el registro.
- Recopile información de envío como direcciones u horarios.
- Encuesta a tus clientes.
- Pago de comercio electrónico.

Formularios HTML

Un formulario HTML utiliza las etiquetas de formulario y tiene algunas propiedades importantes que deben configurarse:

- El atributo **Action** define la acción que se realizará cuando se envíe el formulario.
- El atributo **target** especifica dónde mostrar la respuesta que se recibe después de enviar el formulario. El valor más común es `_blank` lo que significa que la respuesta se muestra en una nueva ventana o pestaña.
- El atributo **method** especifica el método HTTP que se utilizará al enviar los datos del formulario. Hay dos opciones: (1) obtener; o (2) publicar.

Hay más atributos como `enctype` o `accept-charset`. Más [información](#).

Formularios HTML

Un formulario HTML utiliza las etiquetas de formulario y varias etiquetas de entrada para definir la estructura del formulario en la página web..

```
<form>
  <label for="name">Name:</label><br>
  <input type="text" id="name" name="fname"><br>
  <label for="surname">Surname:</label><br>
  <input type="text" id="surname" name="surname">
</form>
```

Este formulario básico se utiliza para recopilar dos datos del usuario, su nombre y su apellido. Además, no incluye ningún atributo en la definición porque los atributos del formulario no son obligatorios.

Formularios HTML

Existen diferentes tipos de controles de entrada que se pueden incluir en una web desde, como:

Input type	Descripción
text	Un control de texto de una sola línea para incluir palabras u oraciones pequeñas. Este es el valor predeterminado de entrada..
radio	Un control de botón de opción que permite seleccionar un solo valor entre varias opciones con el mismo valor de nombre.
checkbox	Un control de casilla de verificación que permite seleccionar o anular la selección de valores individuales.
date	Un control para escribir una fecha (año, mes y día, sin tiempo). Abre un selector de fecha o ruedas numéricas para año, mes y día cuando está activo en exploradores compatibles.
file	Control de archivo que permite al usuario seleccionar un archivo. Utilice el atributo accept para definir los tipos de archivos que el control puede seleccionar.
password	Campo de control de texto de una sola línea cuyo valor está oscurecido. Alertará al usuario si el sitio no es seguro.

Formularios HTML

Hay otros tipos de control como botones que nos permiten ejecutar algunos comportamientos en el formulario.

Input type	Descripción
submit	Un control de botón para enviar el formulario según el valor de la acción de atributo del formulario.
reset	Un control de botón para restablecer el contenido de la entrada del formulario a valores predeterminados.
button	Un control de botón pulsador para ejecutar un comportamiento utilizando JS u otra tecnología. Este control no tiene ningún comportamiento predeterminado.

```
<button type="button" onclick="alert('Hello World!')">Click Me!</button>
```

Formularios HTML

Además, hay algunas etiquetas especiales para incluir información adicional en el formulario en lugar de entradas y botones.:

- Textarea: Es un control de entrada especial para insertar una entrada multilínea (área de texto).

```
<textarea rows="4" cols="50"></textarea>
```

- Selects: Es un control de entrada especial para definir una lista desplegable para la selección individual o múltiple.

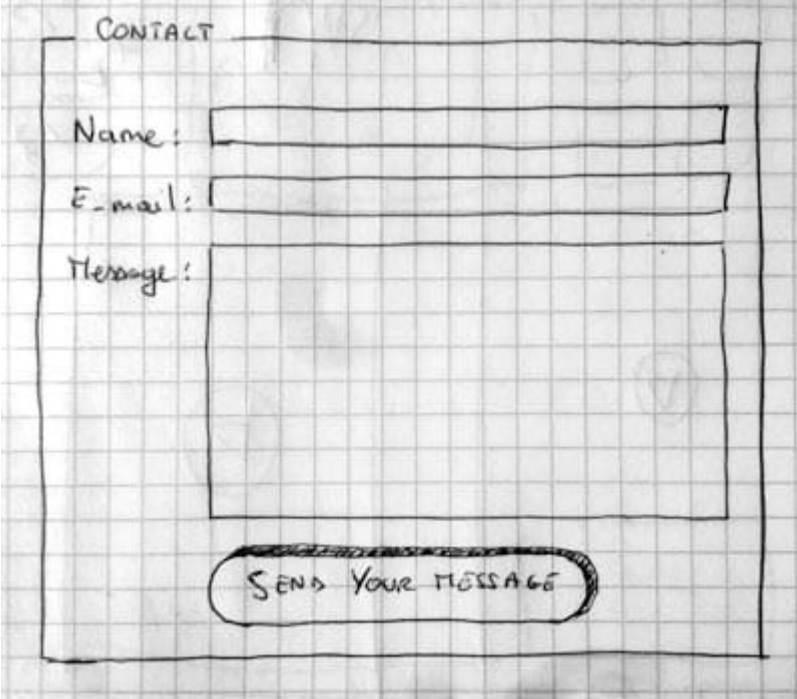
```
<select name="cars" multiple="true">  
  <option value="1">Ferrari</option>  
  <option value="2">Lamborghini</option>  
</select>
```


Formularios HTML

Formularios HTML

La estructura básica de un formulario HTML es así:

```
<form action="/my-handling-form-page" method="post">
  <ul>
    <li>
      <label for="name">Name:</label>
      <input type="text" id="name" name="user_name"/>
    </li>
    <li>
      <label for="mail">Email:</label>
      <input type="email" id="mail" name="user_email"/>
    </li>
    <li>
      <label for="msg">Message:</label>
      <textarea id="msg" name="user_message"></textarea>
    </li>
  </ul>
  <button type="submit">Send your
message</button>
</form>
```



A hand-drawn sketch of a contact form on graph paper. The form is titled "CONTACT" at the top. It contains three input fields: "Name:" followed by a text input box, "E-mail:" followed by a text input box, and "Message:" followed by a larger text area. At the bottom of the form is a rounded rectangular button labeled "SEND YOUR MESSAGE".

Carpetas

Carpetas y ubicaciones

Un sitio web o una aplicación web se compone de diferentes archivos que generalmente se distribuyen en diferentes carpetas. Esto significa que es importante definir las rutas para encontrar los diferentes archivos:

- Una ruta absoluta describe la ubicación desde el directorio raíz del sistema de archivos.
- Una ruta relativa describe la ubicación de un archivo en relación con el directorio actual (de trabajo). Solemos usar rutas relativas y hay algunas reglas importantes para construirlas:
 - / es la raíz de la unidad actual.
 - ./ es el directorio actual.
 - ../ es el padre del directorio actual.

Contenedores flotantes

Contenedores flotantes

Contenedores (divs)

La etiqueta `<div>` es el contenedor genérico para el contenido de flujo. No tiene ningún efecto en el contenido o el diseño hasta que se diseña de alguna manera usando CSS (por ejemplo, el estilo se aplica directamente a él, o algún tipo de modelo de diseño como Flexbox se aplica a su el

HTML Demo: `<div>`RESET

HTML	CSS	OUTPUT
<pre>1 <div class="warning"> 2 4 <p>Beware of the leopard</p> 5 </div> 6</pre>		

Contenedores flotantes

Método 1 - método de bloque en línea

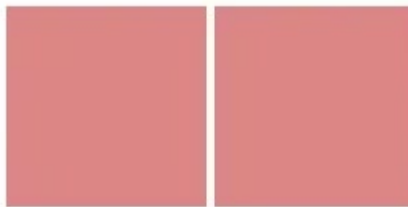
Esta es la forma tradicional de mostrar diferentes contenedores juntos.

```
<!DOCTYPE html>

<head>
</head>

<body>
  <div style="display: inline-block; width: 100px; height: 100px; background: #ce8888;"></div>
  <div style="display: inline-block; width: 100px; height: 100px; background: #ce8888;"></div>
</body>

</html>
```



Contenedores flotantes

Método 2 - método flexbox

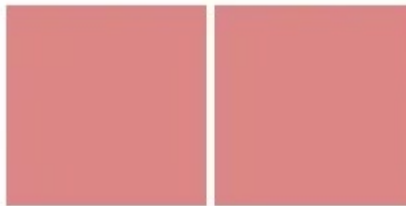
El método flexbox es una forma moderna de diseñar el diseño de la página web y flexbox no es una propiedad única, su módulo completo, tiene una serie de características.

```
<!DOCTYPE html>

<head>
  <link rel="stylesheet" href="index.css" />
</head>

<body>
  <div style="display: flex;">
    <div style="background: #ce8888; flex-basis: 100px; height: 100px; margin: 5px;"></div>
    <div style="background: #ce8888; flex-basis: 100px; height: 100px; margin: 5px;"></div>
  </div>
</body>

</html>
```



Método 3 - método Grid

CSS grid es otra forma de diseñar una página web.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Grid Example</title>
    <style>
      </style>
    </head>
    <body>
      <div style="display: grid; grid-template-columns: 100px 100px; grid-template-rows: 100px;
grid-column-gap: 5px;">
        <div style="background: #ce8888;"></div>
        <div style="background: #ce8888;"></div>
      </div>
    </body>
  </html>
```

