

# LENGUAJES DE MARCAS Y SISTEMAS DE GESTIÓN DE INFORMACIÓN

## UNIDAD 2:

# Utilización de lenguajes de marcas en entornos web



Elena Fernández Chirino

DEPARTAMENTO DE INFORMÁTICA

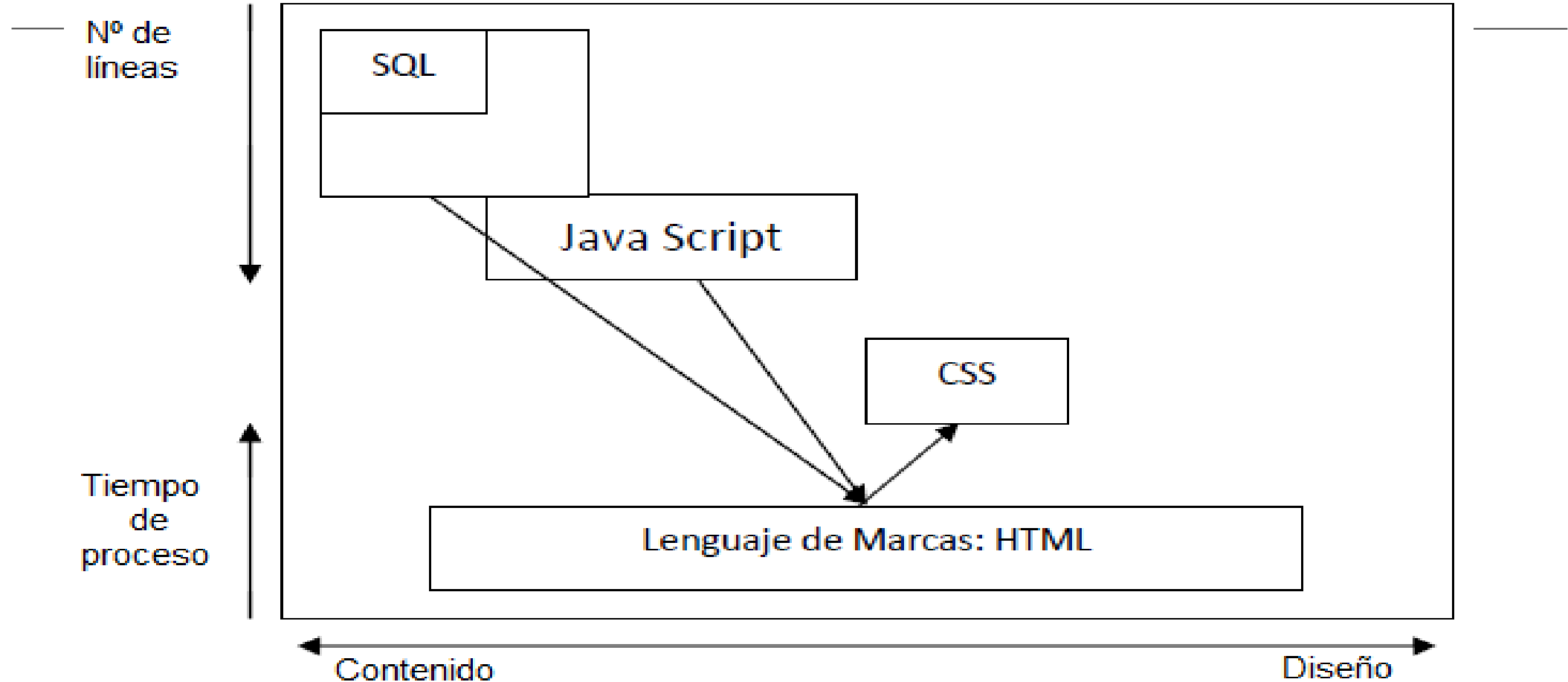
<http://www.iestrassierra.com>



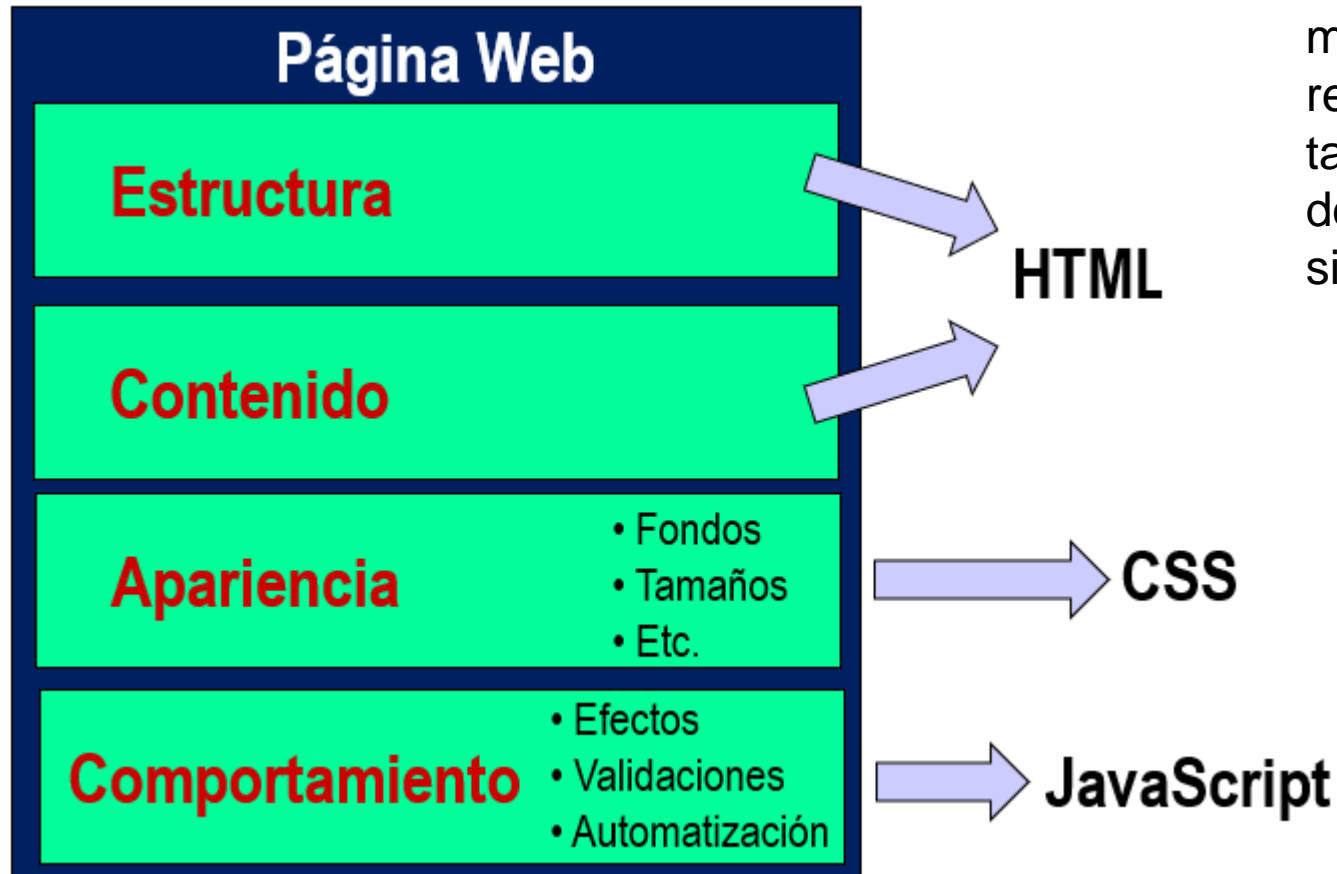
This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#).

# 0.- Introducción

**DISEÑO Y CREACIÓN DE PÁGINAS WEB:** Para diseñar y crear una Web necesitamos hacer uso de diferentes herramientas



# 1.- Del HTML al XHTML: evolución y versiones.



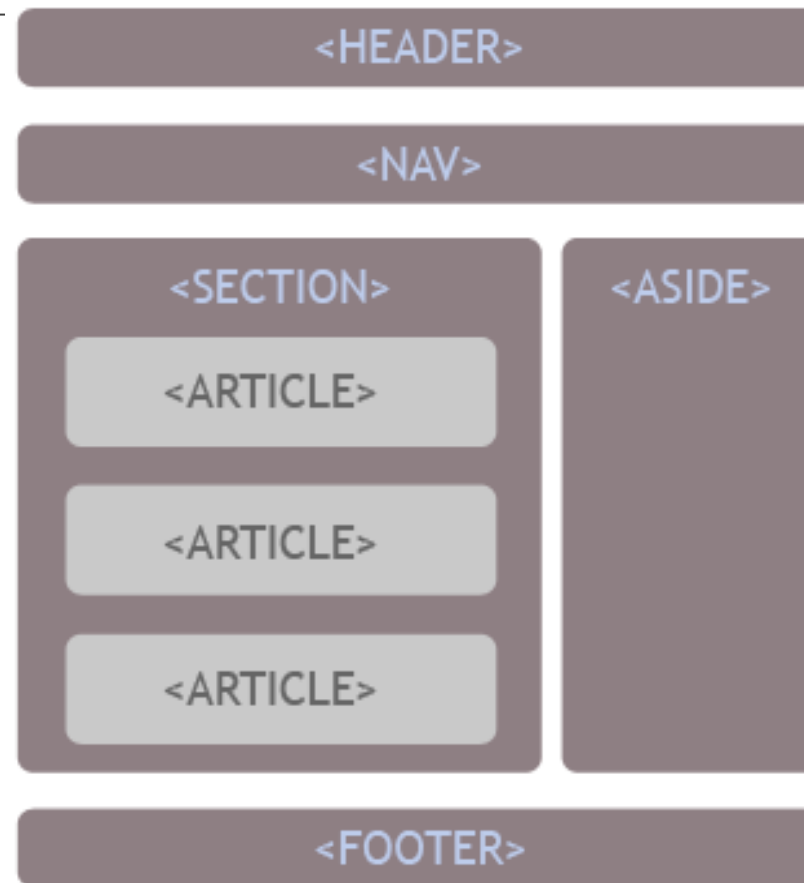
**HTML** es el lenguaje utilizado para crear la mayor parte de las páginas web. Es un estándar reconocido en todos los navegadores, por lo tanto, todos ellos visualizan una página HTML de forma muy similar independientemente del sistema operativo sobre el que se ejecutan.

**HTML, nacido en 1980** como un sistema de hipertexto, ha evolucionado a través de varias versiones, incorporando nuevas funcionalidades y **adaptándose a XML en XHTML**, hasta llegar a **HTML5**, que se centra en una única versión continuamente modificada y mejorada, **abandonando la idea de versiones numeradas**.

# 1.1.- HTML 5 ≈ HTML (nos olvidamos de las versiones numeradas)

Las novedades más destacables de HTML5 son:

1. **Archivos Multimedia:** Soporte integrado para contenido multimedia con los elementos `<audio>` y `<video>`, y el elemento `<canvas>` para gráficos y animaciones.
2. **Web Semántica:** Nuevas etiquetas como `header`, `nav`, `section`, `aside`, `footer`, `article`, `hgroup` y `figure` hacen el lenguaje más descriptivo y entendible tanto para desarrolladores como para buscadores.
3. **Mejoras en Formularios:** **Validaciones sin necesidad de JavaScript**, nuevos atributos y elementos semánticos para los inputs.



# 9. Capas

- Las capas son bloques de contenido HTML que se pueden organizar de manera dinámica y en 3D. Sus atributos (posición, visibilidad, etc.) se **definen en una hoja de estilo** y son **más útiles cuando se les aplican estilos CSS**. **Esto permite que una página web se adapte al dispositivo desde el que se visualiza.**
- El contenido de las capas se especifica en la parte principal de la página, siguiendo la filosofía de separar el contenido y su representación. Las capas pueden ser movidas y modificadas con un lenguaje de script, aunque las implementaciones de los navegadores pueden variar.
- Las **capas se definen normalmente con hojas de estilo y necesitan del lenguaje CSS**. Se pueden crear elementos movibles, ocultables y manipulables de forma dinámica.
- Las **capas pueden tener una posición absoluta o relativa**. La posición absoluta coloca la capa en cualquier parte de la ventana del navegador. La posición relativa describe la posición de la capa respecto al lugar donde aparece en el texto. Es decir, describe el desplazamiento de la capa respecto de donde se escribe.

En resumen, las capas son una herramienta poderosa para crear páginas web dinámicas y adaptables. Sin embargo, es importante tener en cuenta las diferencias entre los navegadores al escribir el código.

# 9. Introducción a CSS. Hojas de estilo

- ✓ Hasta ahora, hemos manejado la presentación de una Web directamente dentro de los elementos HTML por medio de atributos. Por ejemplo:

```
<h2 align="center">  
    <font color="blue" size="3" face="Times New Roman, serif">  
        <i>Apuntes de CSS</i>  
    </font>  
</h2>
```

- ✓ CSS permite separar el contenido de un documento de su presentación.

**En la hoja de estilos se define el formato de los encabezados h2:**

```
h2 {  
    text-align: center;  
    color: blue;  
    font: italic large "Times New Roman", serif;  
}
```

**En el documento HTML:**

```
<h2>Apuntes de CSS</h2>
```

## 9.1.1.- Ejemplo capa con cajas y CSS sencillo

Descargamos los siguientes archivos para ver un ejemplo sencillo de caja utilizando capas:

- [PlantillaCajas](#)
- [styleCajas](#)

```
<!DOCTYPE html> <!-- Declaración del tipo de documento. En este caso, es un documento HTML5 -->
<html lang="es"> <!-- Inicio del documento HTML. El atributo 'lang' especifica el idioma del contenido del documento -->
<head> <!-- Inicio de la sección 'head' que contiene metadatos y enlaces a hojas de estilo -->
  <meta charset="UTF-8"> <!-- Especifica la codificación de caracteres para el documento HTML -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Hace que la página web sea responsive -->
  <title>TemaWeb</title> <!-- Define el título del documento, que se muestra en la barra de título del navegador -->
  <link rel="stylesheet" href="styleCajas.css"> <!-- Enlace a la hoja de estilos CSS externa -->
</head> <!-- Fin de la sección 'head' -->
<body>
  <div class="caja caja-redonda caja-sombra caja-color"> <!-- Aquí delimitamos la capa e indicamos que utilice los estilos definidos -->
    <p>Contenido de la caja...</p>
  </div> <!-- Fin delimitación de la capa -->
</body>
</html>
```

Prueba a borrar los estilos caja, caja-redonda, caja-sombra, caja-color  
¡Y COMPRUEBA COMO CAMBIA TU WEB!

Estilos css que encontrarás en el archivo styleCajas.css

```
/* Estilo de caja genérico */
.caja {
  width: 300px;
  height: 200px;
  padding: 20px;
  border: 2px solid #000;
  margin: 10px;
}
```

```
/* Estilo de caja con esquinas redondeadas */
.caja-redonda {
  border-radius: 20px;
}
```

```
/* Estilo de caja con sombra */
.caja-sombra {
  box-shadow: 10px 10px 5px #888888;
}
```

```
/* Estilo de caja con fondo de color */
.caja-color {
  background-color: #F5F5F5;
}
```

## 9.1.2.- Realiza tu propio ejemplo

Descarga la siguiente [plantilla](#) HTML:

```
<!DOCTYPE html> <!-- Declaración del tipo de documento. En este caso, es un documento HTML5 -->
<html lang="es"> <!-- Inicio del documento HTML. El atributo 'lang' especifica el idioma del contenido del documento -->
<head> <!-- Inicio de la sección 'head' que contiene metadatos y enlaces a hojas de estilo -->
  <meta charset="UTF-8"> <!-- Especifica la codificación de caracteres para el documento HTML -->
  <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Hace que la página web sea responsive -->
  <title>TemaWeb</title> <!-- Define el título del documento, que se muestra en la barra de título del navegador -->
  <link rel="stylesheet" href="styles.css"> <!-- Enlace a la hoja de estilos CSS externa -->
</head> <!-- Fin de la sección 'head' -->
<body>
  <div id="cabecera">BIENVENIDO A MI PÁGINA WEB</div> <!-- Habrá que definir "cabecera" en el styles.css -->
</body>
</html>
```

**Crea un archivo .css para definir un estilo para la cabecera de tu web**

**OBSERVACIÓN IMPORTANTE** → Nos fijamos en las diferentes etiquetas usadas para las capas en los Dos ejemplos planteados

`<div class="caja caja-redonda caja-sombra caja-color">`

`<div id="cabecera">BIENVENIDO A MI PÁGINA WEB</div>`

**VAMOS A ACLARARLO**





## 9.1.2.1- Ejemplo capa y CSS para completar. OBSERVACIÓN PREVIA

**La diferencia entre usar id y class en HTML radica en cómo se utilizan para aplicar estilos CSS.**

- **Id:** asigna un nombre a un elemento determinado y debe haber un solo elemento con ese nombre en todo el documento. Los id son únicos para cada elemento y se utilizan cuando se necesita identificar un solo elemento.

En el ejemplo, si usamos `<div id="cabecera"> BIENVENIDO A MI PÁGINA WEB </div>`.

Solo debería haber un elemento con `id="cabecera"` en todo el documento.

- **Class:** asigna un elemento a una determinada clase y puede haber más de un elemento con el mismo atributo class. Las clases no son únicas y se pueden usar en varios elementos. Por ejemplo, si usamos

```
<div class="caja caja-redonda caja-sombra caja-color">
```

```
<p>Contenido de la caja...</p>
```

```
</div>
```

Podría haber más elementos con `class="cabecera"` en el mismo documento

## 9.1.2.1- Ejemplo capa y CSS para completar. OBSERVACIÓN PREVIA

Ahora bien, en la teoría del tema aparecen los siguientes elementos para las CAPAS

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 hojas de estilo.
class	Permite dar un estilo al contenido del elemento empleando las hojas de estilo.

¿Y si en vez de **class** o **id** usamos el atributo de capa **style**? ¿Qué diferencia hay con los otros dos? → style se utiliza para css interno (que es muy poco habitual)

El atributo style en HTML se utiliza para aplicar estilos CSS directamente a un elemento específico. Este método se conoce como CSS en línea y es muy poco usado

```
<div style="background-color: blue; color: white;">¡Hola, mundo!</div>
```

En este ejemplo el color del fondo de la capa es azul y el color del texto es blanco

## 9.1.2.1- Ejemplo capa y CSS para completar. OBSERVACIÓN PREVIA

**RESUMIENDO:** Las diferencias entre `style`, `class` e `id` son las siguientes:

---

- **Style:** aplica estilos directamente a un elemento específico- los estilos definidos de esta forma tienen la mayor prioridad y anularán cualquier estilo definido en una hoja de estilos externa o en una etiqueta en el encabezado.
- **Class:** hoy se utiliza para definir estilos que se pueden aplicar en múltiples elementos. Los estilos para una clase se definen en una hoja de estilos CSS con su nombre precedido de un punto “.”
- **Id:** Se utiliza para aplicar estilos a un elemento específico. Al igual que `class` los estilos para `id` se define en una hoja de estilos CSS. Sin embargo, a diferencia de `class`, un `id` debe ser único dentro de un documento y su nombre va precedido de “#”

## 9.1.2.- Volvemos a → Realiza tu propio ejemplo .css

Descarga el siguiente [documento HTML](#):

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0"> <!-- Hace que la página web sea responsive -->
    <title>TemaWeb</title>
    <link rel="stylesheet" href="estilo-general.css" type="text/css" media="all">
  </head>
  <body>
    <div id="cabecera">BIENVENIDOS A MI PÁGINA WEB</div>
  </body>
</html>
```

→ Crea un archivo .CSS para la cabecera de tu web

### NOTA → SINTAXIS .CSS

Cada uno de los estilos que componen una hoja de estilos CSS se denomina regla.

Cada regla se forma por:

- ✓ Selector: indica el elemento o elementos HTML a los que se aplica la regla CSS
- ✓ Llave de apertura, {
- ✓ Declaración: especifica los estilos que se aplican a los elementos.
  - Propiedad: permite modificar el aspecto de un atributo del elemento.
  - Valor: indica el nuevo valor del atributo modificado en el elemento.
- ✓ Llave de cierre, }.

Ejemplo: p{ color : blue; }

Se pueden definir estilos propios no exclusivos de elementos HTML. En ese caso irán precedidos de "." o "#", según vayan a ser utilizados más de una vez o no en el documento. Ejemplos

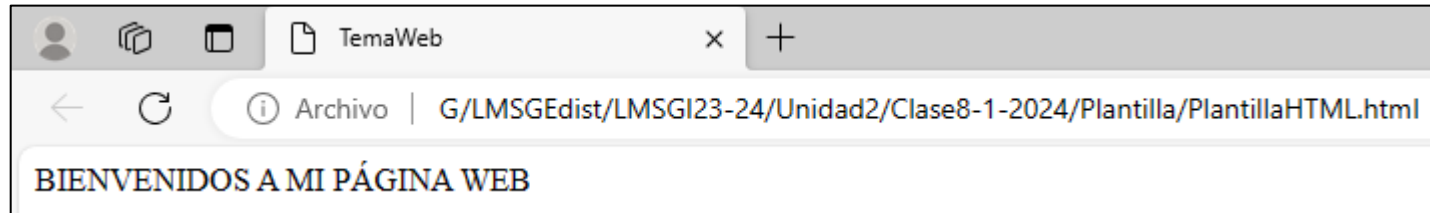
```
.logo {
  height: 100%;
  order: 3; /* Logo a la derecha */
}
```

```
.banner {
  height: 100%;
  order: 1; /* Banner a la izquierda */
}
```

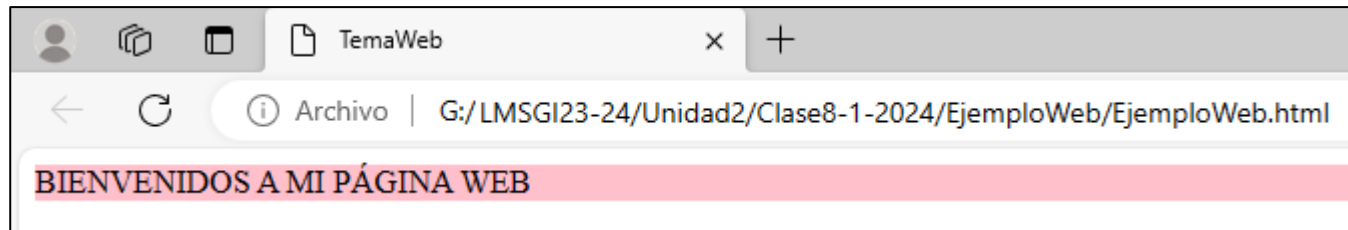
```
#cabecera {background-color: pink;}
```

## 9.1.2.- Posible Solución → Realiza tu propio ejemplo .css

Descarga la siguiente [plantilla](#) HTML ábrela con el navegador. Comprueba que carece de formato



Habría que crear un archivo styles.css en el que definirías un estilo sencillo para “cabecera” y volvieras a abrir con el navegador



**Solución:**  
[Archivo.html](#)  
[Archivo.css](#)

# 10.- Indicaciones para resolver la práctica

## 1.- Descripción de la tarea



### Caso práctico

**Marcas-Formación** planea publicar información sobre sus cursos de desarrollo web y facilitar el acceso a su bibliografía. Los alumnos de FCT, **Pablo** y **Lucía**, son asignados para esta tarea, para lo que se reúnen previamente con **Sofía**, gerente de la empresa. Sofía les sugiere usar Visual Studio Code para el código HTML y CSS. **Lucía** menciona que VS Code permite extensiones para facilitar el trabajo.

Una vez están de acuerdo en la herramienta que van a utilizar, **Pablo** propone crear el documento HTML y la hoja de estilos CSS asociada.

¡¡Es el momento de demostrar nuestros conocimientos en HTML y CSS!!



# 10.- Indicaciones para resolver la práctica

En esta Tarea queremos que elabores una **web** (archivo .html) con una **hoja de estilos** asociada (archivo .css).

**1º.-** La estructura de la página es como se muestra en la siguiente imagen (**Figura 1**):

[Plantilla HTML tarea 2](#)

[Plantilla CSS tarea 2](#)

