



# DIV, SPAN Y CLASS: ESTRUCTURA Y ESTILO EN HTML

# <DIV>

- Definición:
  - <div> es un contenedor de nivel de bloque utilizado para agrupar otros elementos en HTML.
  - Es ideal para estructurar partes de la página y aplicar estilos o scripts específicos.
- Características:
  - Por defecto, ocupa el ancho completo disponible.
  - Utilizado principalmente para dividir el contenido en secciones.

```
<div>
  <h1>Bienvenido a mi sitio web</h1>
  <p>Esta es la sección principal del contenido.</p>
</div>
```

# <SPAN>

- Definición:
  - <span> es un contenedor de nivel en línea que se utiliza para agrupar pequeñas partes del texto o contenido.
  - Es ideal para aplicar estilos o scripts a partes específicas dentro de un texto.
- Características:
  - No interrumpe el flujo del contenido (a diferencia de <div>).
  - Se utiliza principalmente para personalizar estilos.

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

# CLASS

- Definición:
  - class es un atributo global que se utiliza para asignar un nombre a un elemento HTML.
  - Permite aplicar estilos CSS o manejar interacciones JavaScript de forma específica.
- Características:
  - Un elemento puede tener múltiples clases separadas por espacios.
  - Es una herramienta clave para el diseño web.

```
<div class="contenedor">  
  <h2 class="titulo">Título del Artículo</h2>  
  <p class="texto">Este es el contenido del artículo.</p>  
</div>
```

# CÓMO SE CONECTAN

- Relación General:
  - <div> y <span> son contenedores que organizan el contenido.
  - class se utiliza para identificarlos y personalizarlos con CSS o JavaScript.
- Usos Comunes:
  - <div> se utiliza para agrupar secciones grandes.
  - <span> se usa para trabajar en detalles específicos dentro del contenido.
  - class es el puente que conecta HTML con CSS y JavaScript.

```
<!-- HTML -->
<div class="seccion">
  <h1 class="titulo">Mi Página Web</h1>
  <p class="contenido">Este es un ejemplo de cómo
  usar <span class="resaltado">HTML</span>, <span
  class="resaltado">CSS</span>, y <span
  class="resaltado">JavaScript</span>.</p>
</div>
```

```
/* CSS */
.titulo {
  color: blue;
  font-size: 24px;
}

.contenido {
  font-family: Arial, sans-serif;
}

.resaltado {
  color: red;
  font-weight: bold;
}
```

# BENEFICIOS DE USAR <DIV>, <SPAN> Y CLASS

- Flexibilidad:
  - Ayudan a organizar y estructurar el contenido.
- Compatibilidad:
  - Totalmente compatibles con CSS y JavaScript.
- Mantenimiento:
  - Facilitan la actualización y personalización del diseño.



# EJERCICIO

- Objetivo:

Crear una página simple que utilice <div>, <span>, y class para estructurar y personalizar el contenido.

- Instrucciones:

- Crea un archivo llamado index.html.
- Escribe el siguiente código:

```
<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Ejercicio de Div, Span y Class</title>
</head>
<body>
  <div class="seccion">
    <h1 class="titulo">Página de Prueba</h1>
    <p>Este es un ejemplo de un <span
class="resaltado">párrafo</span> con una palabra resaltada.</p>
  </div>
</body>
</html>
```



# RESUMEN FINAL

- `<div>` y `<span>` son fundamentales para organizar el contenido en HTML.
- `class` conecta HTML con CSS y JavaScript, permitiendo personalización y funcionalidad.
- Comprender su relación es clave para estructurar y diseñar páginas web modernas.



# CONEXIÓN ENTRE <DIV>, <SPAN>, CLASS Y LOS ELEMENTOS SEMÁNTICOS

Los elementos semánticos de HTML5 como <header>, <main>, <section>, <article>, y <footer> están diseñados para describir el significado del contenido en una página, mientras que <div> y <span> son genéricos y no proporcionan información semántica sobre el propósito del contenido.

# RELACIÓN Y USOS

- Semántica vs. Genérico
  - Los elementos semánticos:
    - Describen claramente la función del contenido. Esto mejora la accesibilidad, la SEO (optimización para motores de búsqueda) y la comprensión del código.
    - Ejemplo: <header> indica la cabecera de la página o sección; <article> define un contenido independiente.
  - <div> y <span>:
    - No tienen significado semántico. Son contenedores neutros que organizan el contenido y se usan cuando no hay un elemento semántico adecuado.

# USO DE <DIV> CON ELEMENTOS SEMÁNTICOS

- Aunque los elementos semánticos tienen su propio propósito, puedes usar <div> dentro de ellos para agrupar contenido adicional o aplicar estilos.
- Aquí, <header> proporciona la semántica, mientras que <div class="logo"> agrupa elementos relacionados para facilitar el diseño.

```
<header>
  <div class="logo">Mi Sitio Web</div>
  <nav>
    <ul>
      <li><a
href="#inicio">Inicio</a></li>
      <li><a href="#about">Acerca
de</a></li>
    </ul>
  </nav>
</header>
```

# USO DE <SPAN> CON ELEMENTOS SEMÁNTICOS

- Se utiliza para trabajar en detalles específicos, como resaltar palabras clave o estilizar partes del contenido dentro de un elemento semántico.
- Aquí, <article> describe el propósito del contenido, mientras que <span> resalta una parte del texto.

```
<article>
```

```
  <h2>Aprende HTML</h2>
```

```
  <p>HTML significa <span  
class="resaltado">HyperText Markup  
Language</span>, y es esencial para  
construir páginas web.</p>
```

```
</article>
```

# USO DE CLASS CON ELEMENTOS SEMÁNTICOS

- class se usa ampliamente con elementos semánticos para agregar estilos y controlarlos con CSS o JavaScript.
- Aquí, class="destacados" y class="articulo" permiten aplicar estilos específicos a los elementos <section> y <article>.

```
<section class="destacados">
  <article class="articulo">
    <h2>Artículo 1</h2>
    <p>Contenido del artículo 1.</p>
  </article>
  <article class="articulo">
    <h2>Artículo 2</h2>
    <p>Contenido del artículo 2.</p>
  </article>
</section>
```

# CUÁNDO USAR <DIV> Y <SPAN> VS. ELEMENTOS SEMÁNTICOS

Situación	Elemento	Por qué
Agrupar contenido para dar estilo	<div>	Útil para agrupar elementos cuando no hay un significado semántico claro.
Resaltar texto o contenido específico	<span>	Ideal para trabajar en pequeñas partes del contenido sin interrumpir el flujo del texto.
Crear secciones de contenido claro	<section>	Representa una sección temática con un propósito claro.
Contenido independiente	<article>	Representa contenido autónomo que puede leerse de forma independiente (como artículos de blog o noticias).
Encabezado de una página o sección	<header>	Define la cabecera de la página o de una sección específica.
Contenido principal de la página	<main>	Define el contenido único y principal de la página.
Pie de página	<footer>	Proporciona información como créditos, enlaces legales, o información de contacto.



# CONCLUSIÓN

- Elementos semánticos como `<header>`, `<main>`, `<section>`, `<article>`, y `<footer>` describen el propósito del contenido, mejorando la accesibilidad y la SEO.
- `<div>` y `<span>` son genéricos y útiles para agrupar y personalizar contenido, especialmente cuando no hay un elemento semántico adecuado.
- `class` conecta ambos mundos (HTML y CSS/JavaScript), proporcionando la flexibilidad necesaria para personalizar el diseño y la funcionalidad.

Entender cómo combinar estos elementos y atributos es clave para construir páginas web bien estructuradas, accesibles y estilizadas.



# DIFERENCIA ENTRE ID Y CLASS

Característica	id	class
Propósito	Identificación única.	Agrupación de elementos similares.
Unicidad	Solo debe aplicarse a un elemento por página.	Puede aplicarse a varios elementos.
Usos principales	Navegación, anclajes, y JavaScript.	Aplicar estilos y reutilización con CSS y JS.
Prioridad en CSS	Más específico y con mayor prioridad.	Menor prioridad (puede usarse para grupos).
Selección en CSS	Seleccionado con #id en CSS.	Seleccionado con .class en CSS.

# POR QUÉ USAR AMBOS EN UN ELEMENTO

## id para Identificación Única (Navegación o Anclajes)

- Ejemplo:  
Si quieres crear un anclaje para navegar a la sección "Inicio", necesitas un id:  
**`<a href="#inicio">Ir a Inicio</a>`**  
El navegador se desplazará automáticamente al elemento con id="inicio". Esto es útil para menús o navegación interna.

## class para Agrupación y Estilización

Caso de uso: Usas class para aplicar estilos o comportamientos reutilizables a varios elementos.

Ejemplo:

Si tienes varias secciones que deseas estilizar de manera similar, puedes usar la misma clase en todas ellas:

**`<section id="inicio" class="intro">`**

**`<section id="about" class="intro">`**

Luego en CSS:

```
.intro {  
    background-color: lightblue;  
    padding: 20px;  
}
```

# VENTAJAS DE USAR AMBOS

- Flexibilidad:
  - Combinar id y class te permite lograr funcionalidad específica con id y estilización reutilizable con class.
- Mejor Organización:
  - id define elementos clave únicos, como puntos de navegación.
  - class permite agrupar elementos con estilos o comportamientos comunes.
- Separación de Propósitos:
  - id se enfoca en la funcionalidad o la identificación única del elemento (anclajes, eventos con JavaScript).
  - class se enfoca en el diseño y reutilización de estilos.

# EJEMPLO PRÁCTICO

```
<section id="inicio" class="intro">
  <h1>Bienvenido</h1>
  <p>Este es el inicio de nuestra
  página.</p>
</section>
```

```
<section id="about" class="intro">
  <h1>Acerca de Nosotros</h1>
  <p>Conoce más sobre nuestra
  empresa.</p>
</section>
```

```
.intro {
  background-color: #f0f0f0;
  padding: 20px;
  border: 1px solid #ccc;
}
```

```
<nav>
  <ul>
    <li><a
    href="#inicio">Inicio</a></li>
    <li><a href="#about">Acerca
    de</a></li>
  </ul>
</nav>
```