
CSS BASICO

EVAN ALBERTO AGUILAR GARCIA

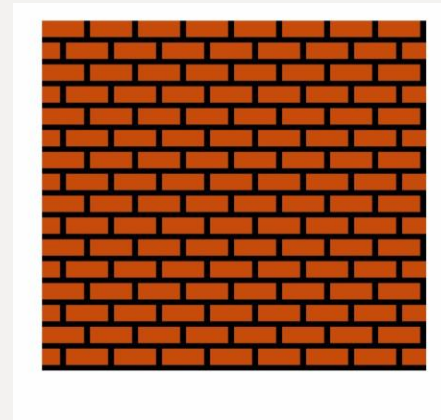
¿QUE ES Y COMO SURGUE CSS?

CSS, que significa "Cascading Style Sheets" (Hojas de Estilo en Cascada), es un lenguaje utilizado para definir el aspecto y la presentación de un documento HTML (o XML). Surgió como una solución a la necesidad de separar el contenido de un sitio web de su presentación visual. Antes de CSS, el formato y el estilo de un documento web estaban directamente integrados en el HTML, lo que hacía que los cambios en la presentación fueran difíciles de implementar y mantener.

CSS surgió como una especificación del World Wide Web Consortium (W3C) en 1996, aunque su desarrollo comenzó mucho antes. La idea básica detrás de CSS es permitir que los diseñadores y desarrolladores web especifiquen el diseño, el color, la fuente y otros aspectos visuales de un documento HTML de manera más eficiente y flexible.



Analogía de CSS



características



Selección de elementos: CSS permite seleccionar elementos específicos de un documento HTML para aplicar estilos, ya sea mediante selección de etiquetas, clases, identificadores u otros atributos.

Propiedades de estilo: CSS ofrece una amplia gama de propiedades para controlar el aspecto visual de los elementos HTML, como color, tamaño, posición, márgenes, rellenos, tipografía, efectos de sombra, bordes, entre otros.

Diseño adaptable y sensible: CSS proporciona herramientas para crear diseños que se ajusten y respondan dinámicamente a diferentes tamaños de pantalla y dispositivos, lo que permite una experiencia de usuario consistente en una variedad de dispositivos, desde teléfonos móviles hasta computadoras de escritorio.

Ventajas



Separación de la estructura y el estilo: CSS permite separar claramente la estructura de un documento HTML del estilo visual aplicado a él. Esto facilita la organización del código, mejora la legibilidad y simplifica la tarea de mantenimiento y actualización del sitio web.

Flexibilidad y control: CSS ofrece un alto grado de flexibilidad y control sobre el diseño y la presentación de un sitio web. Los desarrolladores pueden especificar estilos detallados para cada elemento, lo que les permite crear diseños personalizados y adaptados a las necesidades específicas del proyecto.

Eficiencia: Al permitir la reutilización de estilos a través de hojas de estilo externas y reglas de estilo compartidas, CSS promueve la eficiencia en el desarrollo web. Los cambios en el diseño pueden implementarse de manera rápida y consistente en todo el sitio web simplemente modificando las reglas de estilo en un solo lugar.

Objetivo principal



El objetivo principal de CSS es separar la presentación visual de un documento HTML de su contenido estructural. Esto significa que CSS se utiliza para definir el aspecto y el estilo de un sitio web, incluyendo aspectos como el diseño, el color, la tipografía y los efectos visuales, sin afectar la estructura subyacente del contenido HTML.

Contenido Basico



color: Define el color del texto.

font-family: Especifica la familia de fuentes a utilizar para el texto.

font-size: Establece el tamaño del texto.

font-weight: Define el grosor del texto (por ejemplo, normal, negrita).

text-align: Alinea el texto dentro de su contenedor (por ejemplo, izquierda, derecha, centrado).

background-color: Define el color de fondo de un elemento.

padding: Define el espacio entre el borde de un elemento y su contenido.

margin: Establece el espacio entre los bordes de un elemento y sus elementos adyacentes.

border: Define un borde alrededor de un elemento.

width: Establece el ancho de un elemento.

height: Define la altura de un elemento.

display: Controla cómo se muestra un elemento en el diseño (por ejemplo, bloque, en línea, flex-box y grid-model).

float: Especifica si un elemento debe flotar a la izquierda o a la derecha de su contenedor.

position: Controla el método de posicionamiento de un elemento (por ejemplo, relativo, absoluto, fijo).

z-index: Define el orden de apilamiento de un elemento en relación con otros elementos.

opacity: Establece la opacidad de un elemento (valores entre 0 y 1).

text-decoration: Define la decoración del texto (por ejemplo, subrayado, tachado).

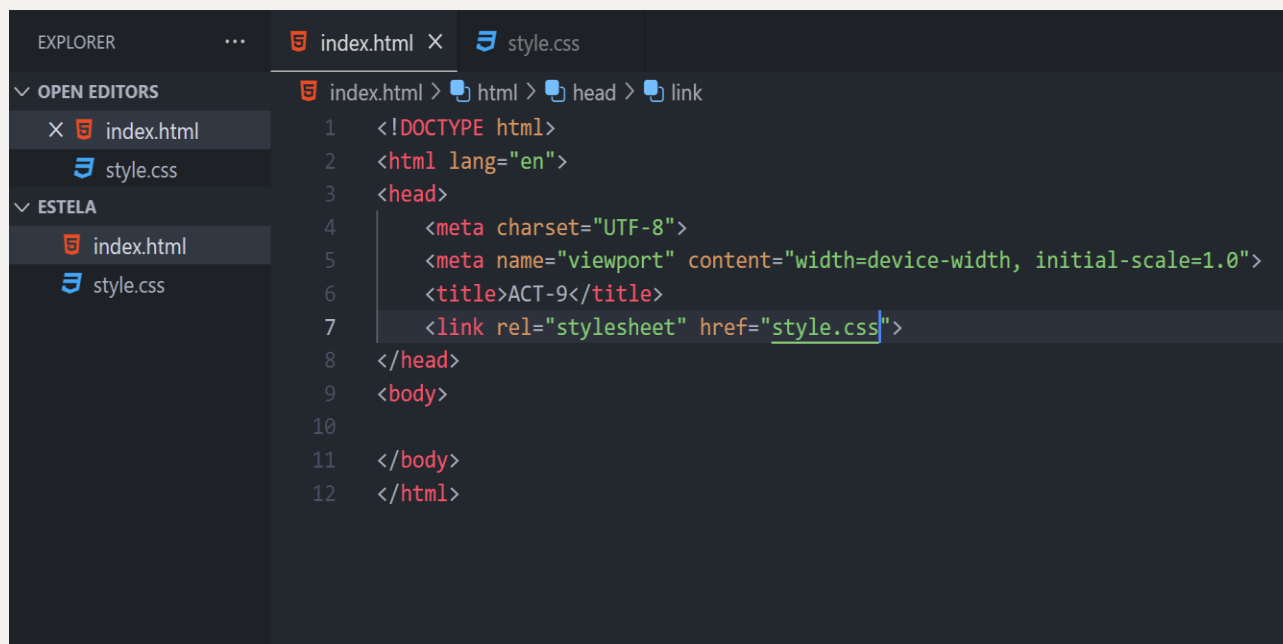
line-height: Establece la altura de línea del texto.

box-shadow: Agrega una sombra alrededor de un elemento.

border-radius: Define el radio de las esquinas de un elemento.

Formas de incluir estilos: `<link>`

Integra el diseño de manera eficiente directamente desde una hoja de estilo CSS, lo cual no solo agrega claridad al código, sino que también mejora significativamente la legibilidad al mantener separadas las reglas de estilo CSS del marcado HTML. Esta práctica no solo facilita la organización y el mantenimiento del código, sino que también promueve una estructura más limpia y modular, lo que a su vez contribuye a una mejor escalabilidad y mantenibilidad del proyecto en general.

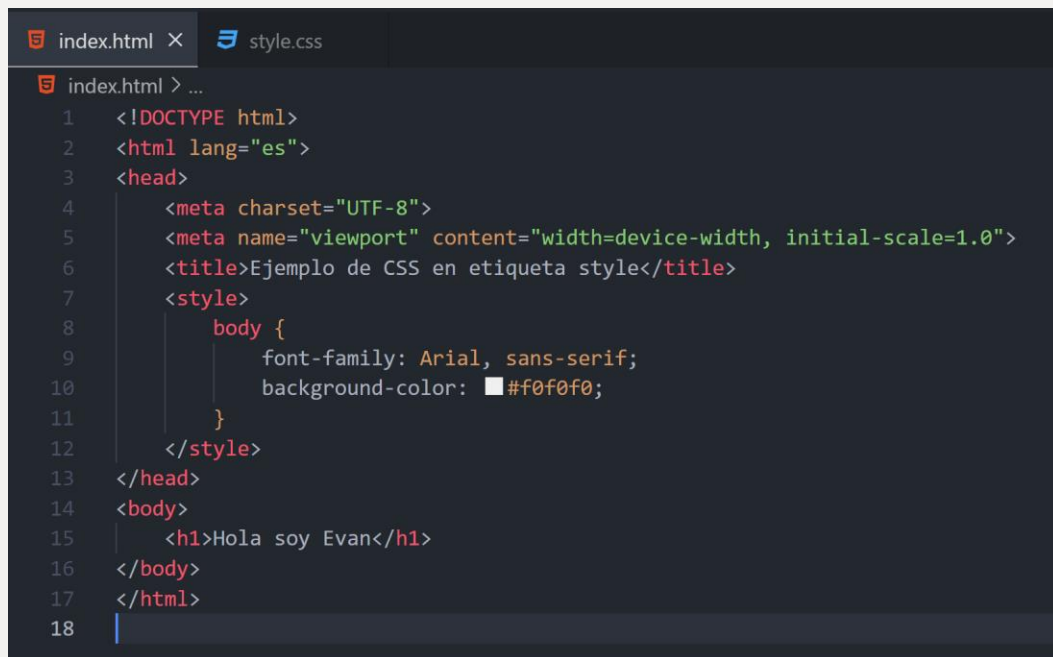


The screenshot shows a code editor with two files open: `index.html` and `style.css`. The `index.html` file is selected, and the editor displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>ACT-9</title>
7   <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10
11 </body>
12 </html>
```

The `<link rel="stylesheet" href="style.css">` tag is highlighted, demonstrating how to include an external CSS file using the `<link>` element.

Formas de incluir estilos: <style>

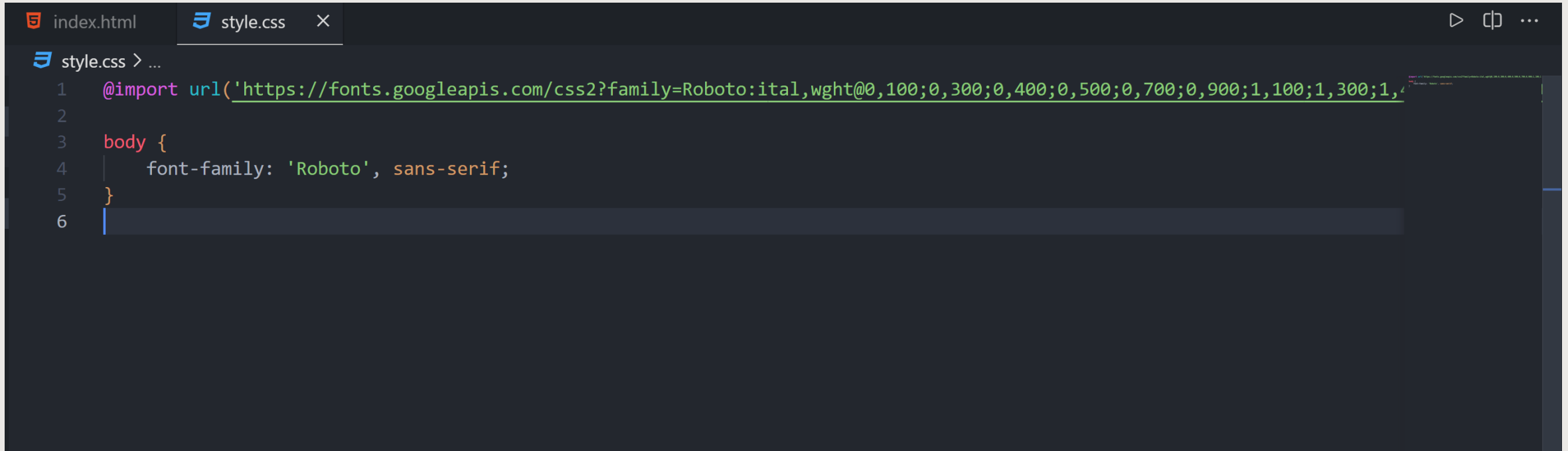


```
index.html x style.css
index.html > ...
1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Ejemplo de CSS en etiqueta style</title>
7   <style>
8     body {
9       font-family: Arial, sans-serif;
10      background-color: #f0f0f0;
11    }
12  </style>
13 </head>
14 <body>
15   <h1>Hola soy Evan</h1>
16 </body>
17 </html>
18
```

La etiqueta <style> en HTML se utiliza para incluir reglas de estilo CSS dentro del documento HTML mismo. Esta etiqueta se coloca típicamente dentro de la sección <head> del documento y puede contener una serie de declaraciones CSS que especifican cómo se deben visualizar los elementos HTML en la página. Al usar la etiqueta <style>, puedes definir estilos para todos los elementos del documento o para un conjunto específico de elementos, lo que permite personalizar la apariencia de la página de manera precisa y coherente. Además, el uso de la etiqueta <style> facilita la organización y el mantenimiento del código al mantener todos los estilos relacionados con el documento dentro del mismo archivo HTML.

Sentencia @import

La regla @import en CSS permite importar estilos desde archivos externos, como fuentes de Google Fonts, facilitando la incorporación de diferentes estilos y fuentes en un documento CSS.

A screenshot of a code editor with a dark theme. The editor has two tabs at the top: 'index.html' and 'style.css'. The 'style.css' tab is active. The code in the editor is as follows:

```
1  @import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=block');
2
3  body {
4      font-family: 'Roboto', sans-serif;
5  }
6
```

The code is color-coded: '@import' is purple, 'url' is blue, the URL is green, 'body' is red, and 'font-family' is blue. The line numbers 1 through 6 are on the left side of the editor.


Sintaxis de css

La sintaxis básica de CSS sigue la estructura de regla de la siguiente manera:

Selector: Indica los elementos HTML a los que se aplicarán los estilos.

Propiedad: Define la característica que se va a cambiar, como color, tamaño o posición.

Valor: Especifica el valor que se asignará a la propiedad.

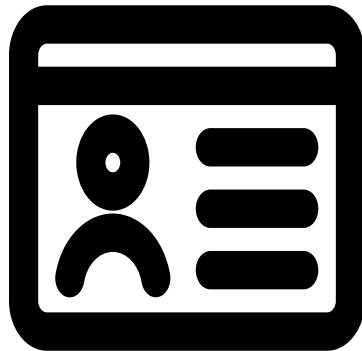
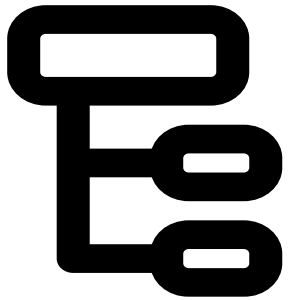
A screenshot of a code editor with a dark theme. The editor shows a file named 'style.css' with a 'selector' icon. The code is as follows:

```
1 selector {  
2     propiedad: valor;  
3 }  
4
```

The word 'propiedad' is underlined with a wavy orange line, indicating a warning or error. The line numbers 1, 2, 3, and 4 are on the left side of the editor.

Que son los selectores

Los selectores en CSS son patrones que permiten identificar y seleccionar elementos HTML en un documento para aplicarles estilos. Estos patrones pueden basarse en varios criterios, como el tipo de elemento, su clase, su ID, su posición dentro de la estructura del documento, sus atributos, entre otros.



Selectores basicos

Selector de tipo: Selecciona elementos basados en el nombre de la etiqueta HTML, como p para párrafos o h1 para encabezados de nivel 1.

Selector de clase: Selecciona elementos que tienen un atributo class específico, como .clase para elementos con la clase "clase".

Selector de ID: Selecciona un elemento con un atributo id específico, como #id para elementos con el ID "id".

```
style.css > #Id
1  h1 {
2      color: wheat;
3  }
4
5  .class {
6      color: aqua;
7  }
8
9  #Id {
10     color: bisque;
11 }
12
```

```
index.html x style.css
index.html > ...
1  <!DOCTYPE html>
2  <html lang="es">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-wi
6      <title>Ejemplo de selectores</title>
7  </head>
8  <body>
9      <h1>Hola soy Evan</h1>
10     <p class="class">¿Como estas?</p>
11     <p id="Id">Saludos</p>
12 </body>
13 </html>
14
```

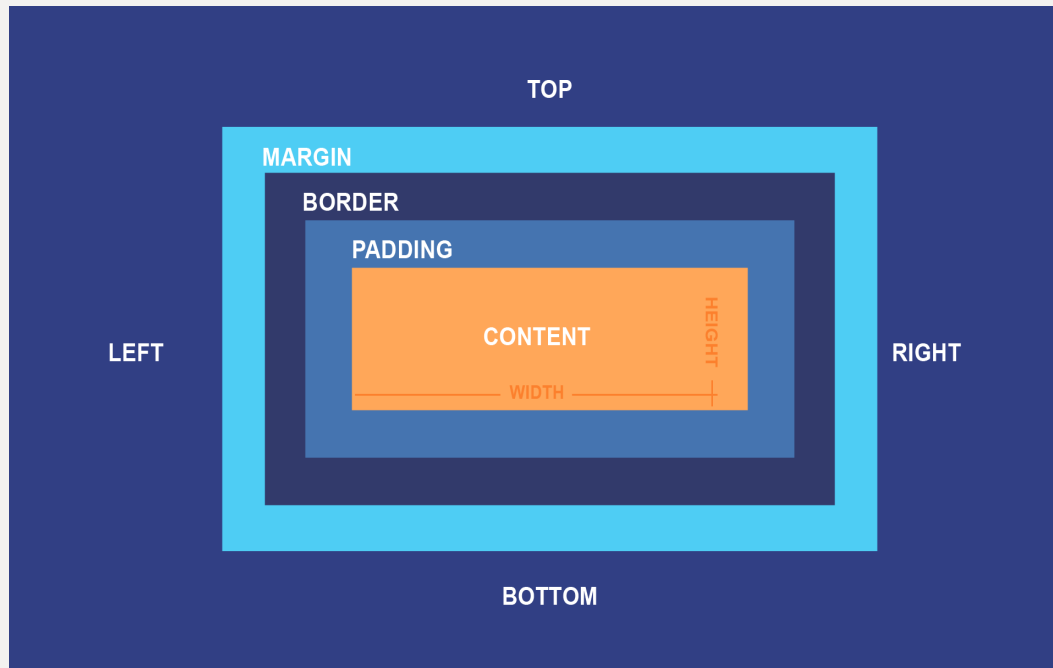
Notacion de colores

Notación hexadecimal: Utiliza una combinación de seis caracteres alfanuméricos que representan los valores de los componentes rojo, verde y azul (RGB) de un color. Por ejemplo, #FF0000 representa el color rojo puro, donde FF representa la intensidad máxima de rojo y 00 representa la ausencia de verde y azul.

Notación RGB: Utiliza la función rgb() para definir un color en términos de sus componentes rojo, verde y azul. Por ejemplo, rgb(255, 0, 0) representa el mismo color rojo puro que el ejemplo anterior en notación hexadecimal.

```
index.html style.css X
style.css > .BtnCheck
1 .BtnCheck {
2     background-color: #fff;
3     /* Notacion hexadecimal*/
4
5     color: rgb(0, 0, 0);
6     /*Notacion RGB) */
7
8     color: wheat;
9     /*Notacion basica*/
10 }
```

Modelado de cajas



El modelo de caja (box model) en CSS es un concepto fundamental que describe cómo se calcula el tamaño total de un elemento en una página web. Este modelo considera que cada elemento HTML se representa como una "caja" rectangular, que consiste en contenido, padding, borde y margen. Estas son las partes principales del modelo de caja:

Contenido (Content): Es el área dentro de la caja que contiene el contenido real del elemento, como texto, imágenes o cualquier otro elemento HTML anidado.

Padding: Es el espacio entre el contenido del elemento y su borde. Se puede especificar un padding para cada lado de la caja (arriba, abajo, izquierda, derecha) y se utiliza para crear espacio adicional alrededor del contenido.

Borde (Border): Es el borde que rodea el contenido y el padding del elemento. Se puede definir el estilo, el ancho y el color del borde.

Margen (Margin): Es el espacio entre el borde del elemento y los elementos adyacentes. Se puede especificar un margen para cada lado de la caja y se utiliza para controlar la separación entre elementos.

ejemplo



Codigo sin padding y margin

```
index.html > html > body > h3
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width,
6   <title>CodingLab</title>
7   <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10   <header>
11     <h1>Contenido</h1>
12   </header>
13
14   <h3>soy otro contenido</h3>
15 </body>
16 </html>

style.css > header
1 header {
2   display: flex;
3   flex-direction: row;
4   justify-content: center;
5   background-color: #f5deb3;
6 }
```


Con padding y sin padding

Sin padding



padding



```
style.css
1 header {
2   display: flex;
3   flex-direction: row;
4   justify-content: center;
5   background-color: #bisque;
6   padding: 50px;
7 }
8
9 h3 {
10  text-align: center;
11 }
```

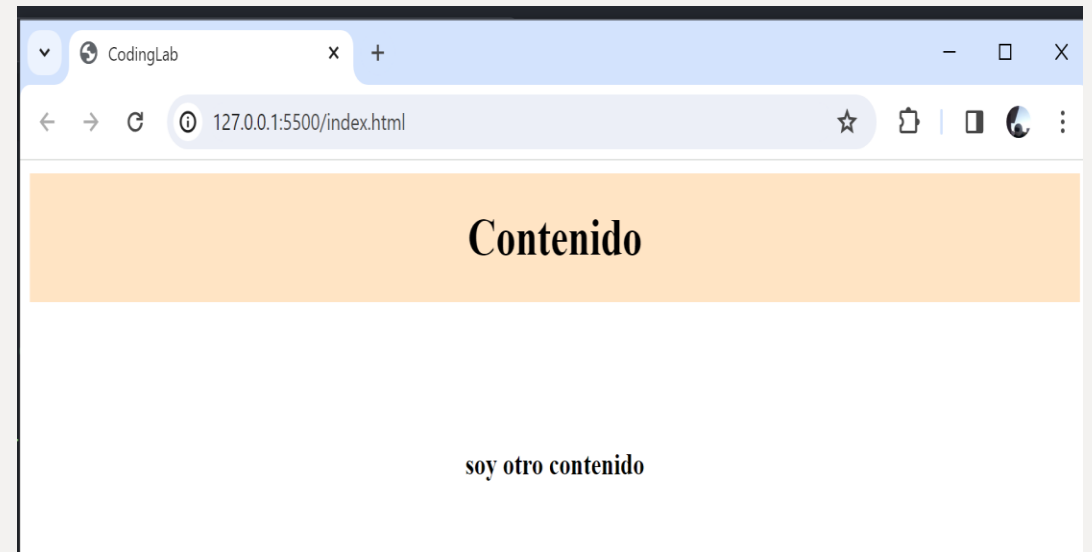
Con margin y sin margin

```
style.css > header
1 header {
2   display: flex;
3   flex-direction: row;
4   justify-content: center;
5   background-color: bisque;
6   margin-bottom: 90px;
7 }
8
9 h3 {
10  text-align: center;
11 }
```

Sin margin



Margin



Creditos

Mozilla

FreeCodeCamp

Udemy

