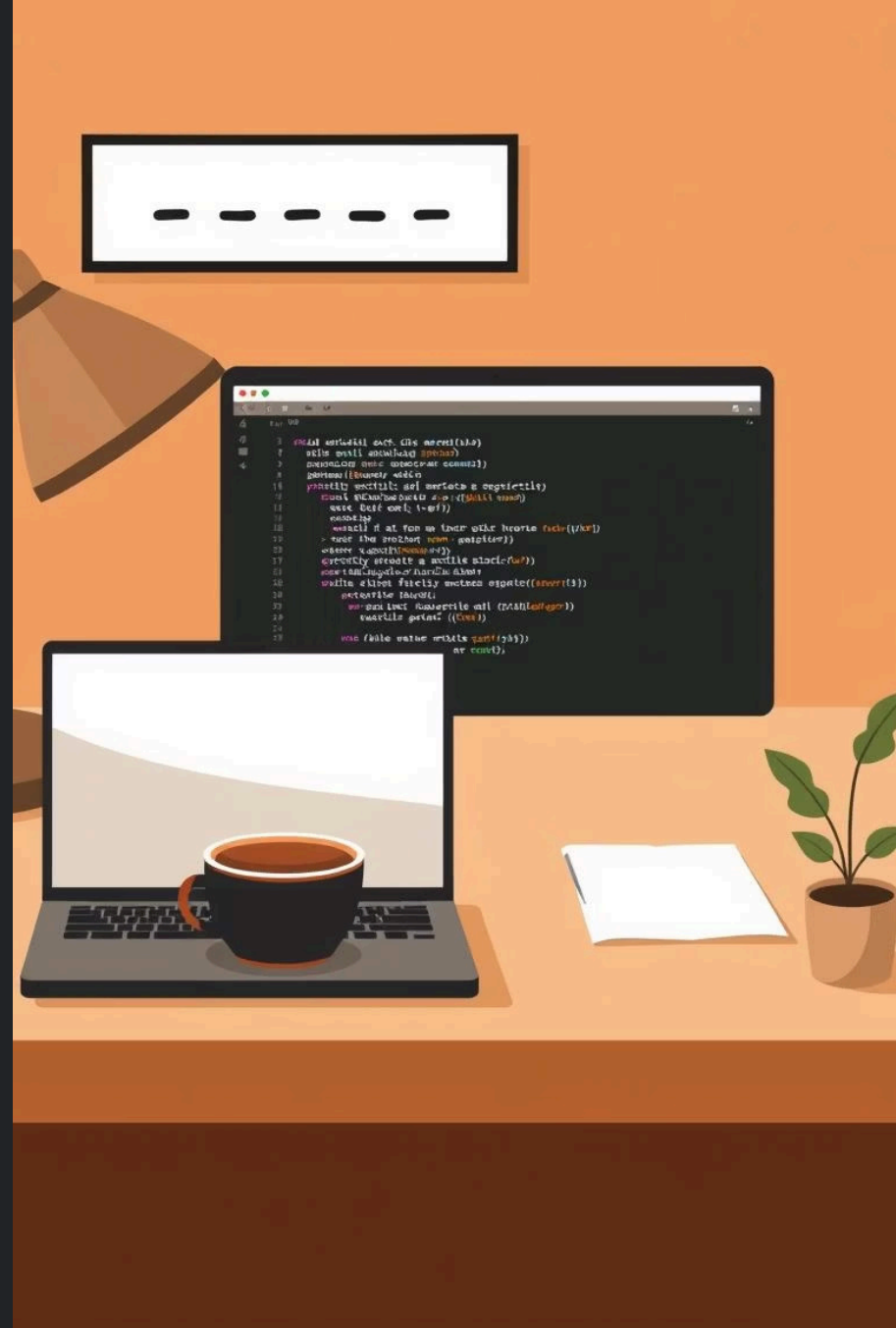


# Desvelando CSS: Estilizando tu Web

Bienvenidos a esta introducción a CSS, el lenguaje que da vida y estilo a nuestras páginas web. En esta presentación, exploraremos los fundamentos de CSS, desde su propósito hasta cómo aplicarlo y algunos conceptos clave para que puedas empezar a estilizar tus propios proyectos web.



## Agenda

# ¿Qué veremos hoy?

### 1 ¿Qué es CSS?

Entendiendo su función y por qué es esencial.

### 2 Sintaxis Básica

Anatomía de una regla CSS: selectores, propiedades y valores.

### 3 Métodos de Inclusión

Cómo vincular CSS a tu HTML: externo, interno e en línea.

### 4 Selectores Comunes

Seleccionando elementos específicos para estilizar.

### 5 Propiedades Esenciales

Color, tipografía, cajas y posicionamiento.

### 6 Modelo de Caja

Cómo los navegadores representan los elementos HTML.

### 7 Flexbox Básico

Una introducción a la alineación y distribución de elementos.

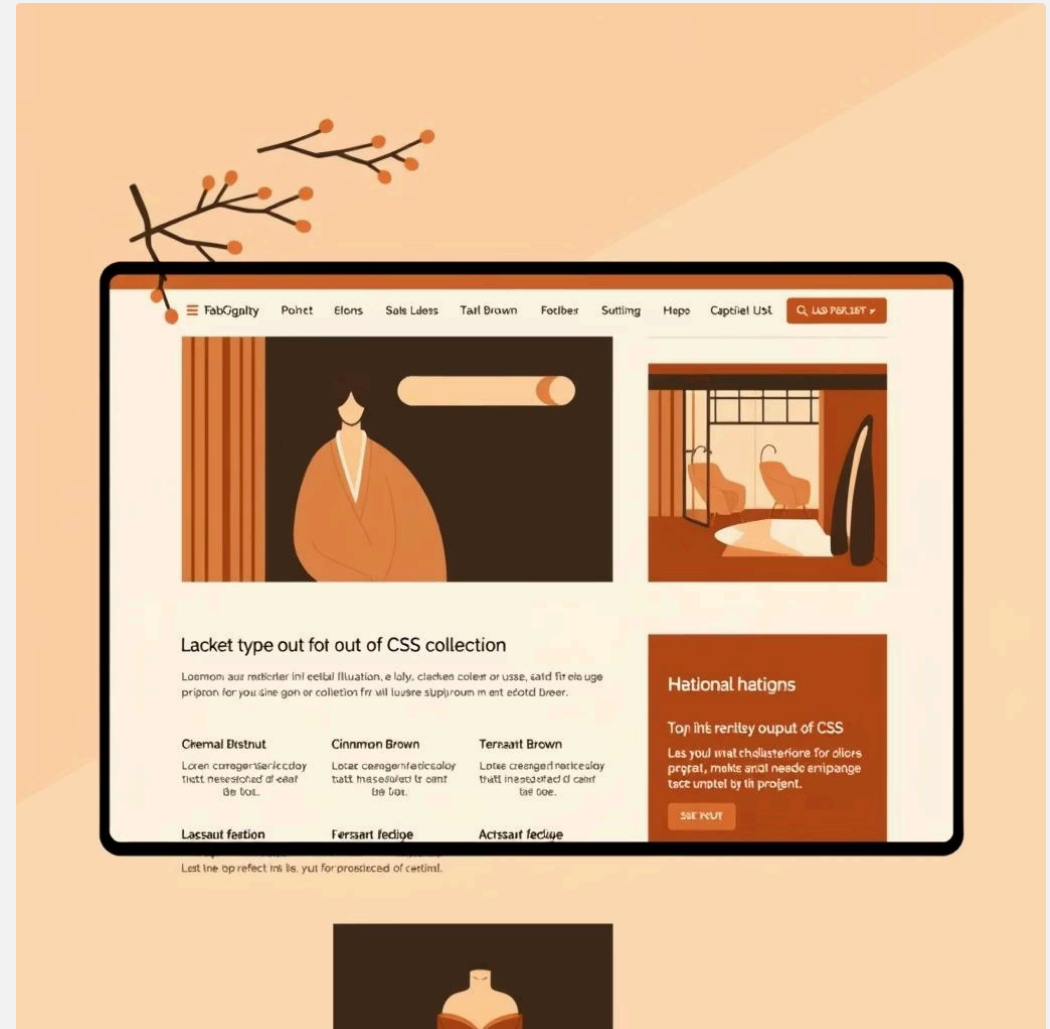
### 8 Ejercicio Práctico

¡Manos a la obra! Aplica lo aprendido.

# ¿Qué es CSS?

CSS, o Cascading Style Sheets (Hojas de Estilo en Cascada), es un lenguaje de hojas de estilo que se utiliza para describir la presentación de un documento escrito en HTML. Imagina HTML como el esqueleto de tu página web; CSS es la ropa, el maquillaje y la decoración que lo hacen visualmente atractivo.

Con CSS, puedes controlar el color de los textos, el tipo de fuente, el espaciado entre párrafos, el tamaño de los elementos, la disposición de las columnas y muchas otras características visuales de tu sitio web. Su objetivo principal es separar el contenido de la presentación, haciendo que tu código sea más limpio, fácil de mantener y adaptable a diferentes dispositivos.



# Sintaxis Básica de CSS

Una regla CSS típica está compuesta por un **selector**, que apunta al elemento HTML que quieres estilizar, y un bloque de **declaración**, que contiene una o más **propiedades** y sus **valores**.

```
selector {  
  propiedad: valor;  
  propiedad2: valor2;  
}
```

Por ejemplo, para cambiar el color de un párrafo a azul:

```
p {  
  color: blue;  
  font-size: 16px;  
}
```

- **Selector** (p): Selecciona todos los elementos de párrafo.
- **Propiedad** (color): Define el color del texto.
- **Valor** (blue): El color que se aplicará.
- **Declaración** (color: blue;): La combinación de propiedad y valor.



CSS < >  
sete.10

# Métodos de Inclusión de CSS

Hay tres formas principales de vincular tu código CSS a tu documento HTML:



## 1. Hoja de Estilos Externa

Es la forma más común y recomendada. Creas un archivo `.css` separado (por ejemplo, `style.css`) y lo vinculas a tu HTML usando la etiqueta `<link>` dentro de la sección `<head>`.

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



## 2. Hoja de Estilos Interna

Colocas el CSS directamente dentro de la etiqueta `<style>` en la sección `<head>` de tu documento HTML. Útil para estilos específicos de una sola página.

```
<head>
  <style>
    h1 {
      color: #BE6841;
    }
  </style>
</head>
```



## 3. Estilos en Línea

Aplicas estilos directamente a un elemento HTML usando el atributo `style`. Generalmente se desaconseja, ya que mezcla contenido y presentación, dificultando el mantenimiento.

```
<p style="color: #84482D;
font-weight: bold;">
  Este es un párrafo estilizado.
</p>
```

# Selectores Comunes

Los selectores son cruciales para indicar a qué elementos HTML deben aplicarse las reglas CSS. Aquí tienes los más comunes:

1

## Selector de Elemento (Tipo)

Selecciona todos los elementos de un tipo específico. Por ejemplo, `h1` selecciona todos los encabezados de nivel 1.

```
h1 { color: #1C1D1F; }
```

2

## Selector de ID

Selecciona un único elemento con un id específico. Los IDs son únicos en una página. Se usa el símbolo `#`.

```
#miTitulo { font-size: 24px; }
```

3

## Selector de Clase

Selecciona todos los elementos con una class específica. Las clases pueden aplicarse a múltiples elementos. Se usa el símbolo `.`.

```
.resaltado { background-color: yellow; }
```

4

## Selector Universal

Selecciona todos los elementos de la página. Se usa el símbolo `*`. Útil para reinicios de estilos.

```
* { margin: 0; padding: 0; }
```

# Propiedades Esenciales de CSS

CSS ofrece cientos de propiedades, pero algunas son fundamentales para empezar a estilizar:



## Color y Fondo

color para el texto, background-color para el fondo de un elemento.

```
p { color: #BE6841; }
```



## Tipografía

font-family, font-size, font-weight, text-align, etc., para controlar el texto.

```
h2 { font-family: Arial; }
```



## Modelo de Caja

width, height, margin, padding, border para dimensiones y espaciado de elementos.

```
div { border: 1px solid #1C1D1F; }
```



## Posicionamiento

display (block, inline, flex, grid), position (static, relative, absolute, fixed) para el layout.

```
img { display: block; }
```



# El Modelo de Caja de CSS

Cada elemento HTML se renderiza como una caja rectangular invisible, y el Modelo de Caja de CSS describe cómo se estructuran estas cajas. Entenderlo es fundamental para el layout y el espaciado.

- **Contenido:** El texto o la imagen real dentro del elemento.
- **Padding (Relleno):** Espacio transparente entre el contenido y el borde. Aumenta el tamaño visible de la caja.
- **Border (Borde):** Una línea alrededor del padding y el contenido.
- **Margin (Margen):** Espacio transparente fuera del borde, que separa el elemento de otros elementos adyacentes.

Puedes controlar cada una de estas propiedades para ajustar el tamaño y la posición de tus elementos en la página.



# Introducción a Flexbox

Flexbox es un módulo de diseño CSS que facilita la alineación y distribución de elementos dentro de un contenedor. Es especialmente útil para crear layouts responsivos y complejos con pocas líneas de código.

## Contenedor Flex (Parent)

Para usar Flexbox, primero defines un contenedor como flex con `display: flex;`. Los hijos directos de este contenedor se convierten en ítems flex.

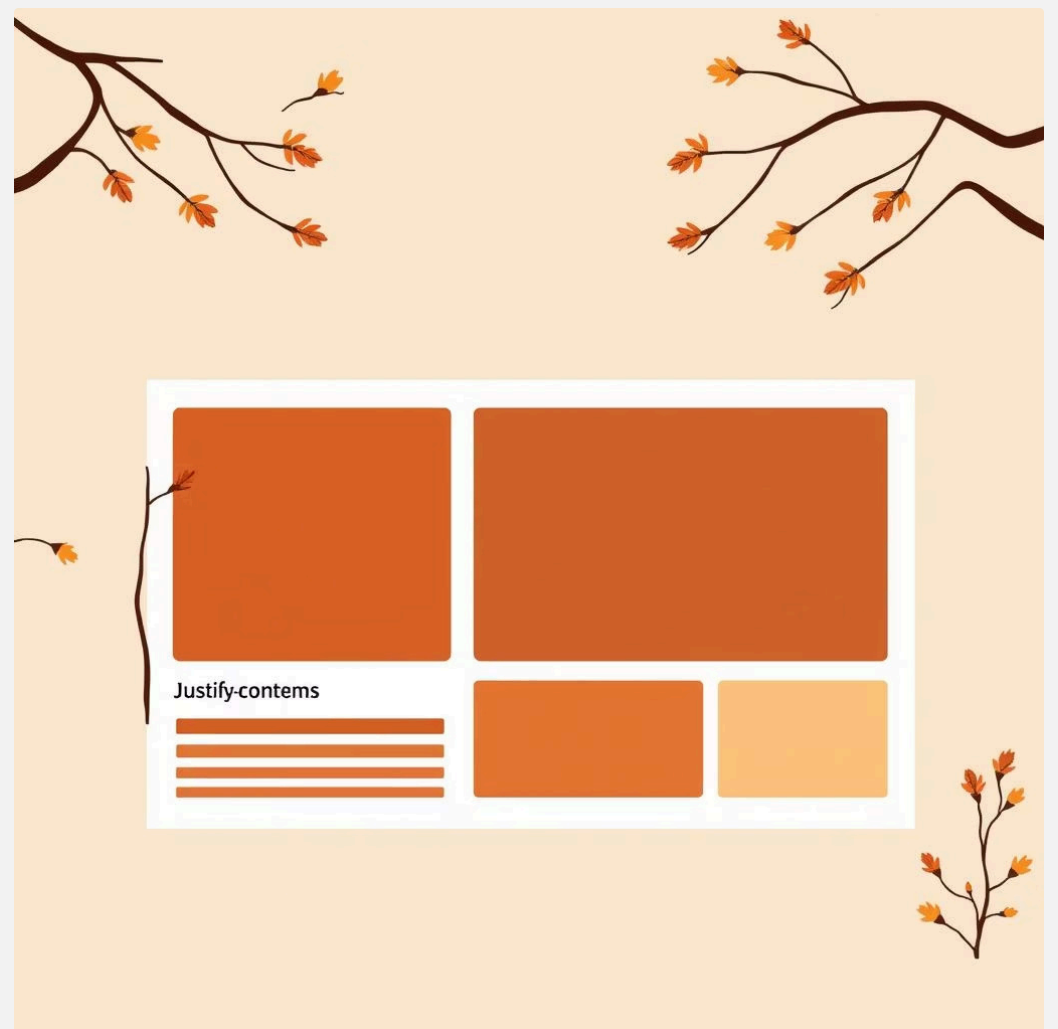
```
.contenedor {  
  display: flex;  
  justify-content: space-around;  
  align-items: center;  
}
```

- `justify-content`: Alinea los ítems a lo largo del eje principal (horizontal por defecto).
- `align-items`: Alinea los ítems a lo largo del eje transversal (vertical por defecto).

## Ítems Flex (Children)

Los ítems dentro del contenedor flex se ajustarán automáticamente. Puedes controlar su tamaño y orden.

```
.item {  
  flex: 1; /* Permite que el ítem crezca y se encoja */  
  padding: 10px;  
  background-color: #B57152;  
  color: white;  
}
```



# ¡Manos a la Obra! Ejercicio Práctico

Ahora es tu turno. Crea un archivo HTML y un archivo CSS externo para aplicar los siguientes estilos:

- Crea un encabezado `<h1>` con el texto "Mi Primera Página Estilizada".
- Crea tres párrafos `<p>` con contenido dummy.
- Añade un `<div>` que contenga dos imágenes pequeñas.
- Aplica los siguientes estilos usando un archivo CSS externo (`style.css`):

## Estilo del Encabezado

- Color del texto: `#84482D`
- Alineación del texto: centrada

## Estilo de los Párrafos

- Tamaño de fuente: 18px
- Espaciado de línea: 1.5em
- Color del texto: `#1C1D1F`

## Estilo del Contenedor de Imágenes

- Display: flex
- Justify-content: space-around
- Borde: 2px sólido `#BE6841`
- Padding: 20px

## Estilo de las Imágenes

- Ancho: 150px
- Margen: 10px