[1.Introducción 3](#_Toc172149670)

[2.Sintaxis 3](#_Toc172149671)

[2.1 Desafío 4](#_Toc172149672)

[3.CSS externo 6](#_Toc172149673)

[3.1 Desafío 6](#_Toc172149674)

[4.CSS interno 7](#_Toc172149675)

[4.1 Desafío 8](#_Toc172149676)

[5.CSS en línea 8](#_Toc172149677)

[5.1 Desafío 9](#_Toc172149678)

[6.Introducción 9](#_Toc172149679)

[7.Por etiqueta 9](#_Toc172149680)

[7.1 Desafío 10](#_Toc172149681)

[8.Por ID 11](#_Toc172149682)

[8.1 Desafío 11](#_Toc172149683)

[9.Por clase 12](#_Toc172149684)

[9.1 Desafío 12](#_Toc172149685)

[10.Múltiples opciones 14](#_Toc172149686)

[10.1 Desafío 14](#_Toc172149687)

[11.Colores 16](#_Toc172149688)

[11.1 Desafío 16](#_Toc172149689)

[12.Fondo 17](#_Toc172149690)

[12.1 Desafío 17](#_Toc172149691)

[13.Tamaños 18](#_Toc172149692)

[**Unidades absolutas** 18](#_Toc172149693)

[**Unidades relativas** 19](#_Toc172149694)

[13.1 Desafío 19](#_Toc172149695)

[14.Width & Height 20](#_Toc172149696)

[14.1 Desafío 20](#_Toc172149697)

[15.Border 21](#_Toc172149698)

[15.1 Desafío 22](#_Toc172149699)

[16.Tamaños de borde 22](#_Toc172149700)

[16.1 Desafío 23](#_Toc172149701)

[17.Desafío de propiedades 24](#_Toc172149702)

[17.1 Desafío 24](#_Toc172149703)

[18.Introducción 26](#_Toc172149704)

[19.Margen 26](#_Toc172149705)

[19.1 Desafío 27](#_Toc172149706)

[20.Padding 28](#_Toc172149707)

[20.1 Desafío 29](#_Toc172149708)

[21.Tamaño del box 30](#_Toc172149709)

[21.1 Desafío 30](#_Toc172149710)

[22.Sombra del box 31](#_Toc172149711)

[22.1 Desafío 32](#_Toc172149712)

[23.Border Radius 33](#_Toc172149713)

[23.1 Desafío 33](#_Toc172149714)

[24.Desafío de modelado del box 34](#_Toc172149715)

[24.1 Desafío 34](#_Toc172149716)

[25.Color del texto 36](#_Toc172149717)

[25.1 Desafío 36](#_Toc172149718)

[26.Alineación de texto 37](#_Toc172149719)

[26.1 Desafío 37](#_Toc172149720)

[27.Decoración de texto 38](#_Toc172149721)

[27.1 Desafío 39](#_Toc172149722)

[28.Sombra de texto 40](#_Toc172149723)

[28.1 Desafío 40](#_Toc172149724)

[29.Fuentes de texto 41](#_Toc172149725)

[29.1 Desafío 42](#_Toc172149726)

[30.Desafío de texto 43](#_Toc172149727)

[30.1 Desafío 43](#_Toc172149728)

[31.Introducción 44](#_Toc172149729)

[32.Flex Box 44](#_Toc172149730)

[32.1 Desafío 44](#_Toc172149731)

[33.Dirección del flex 45](#_Toc172149732)

[33.1 Desafío 45](#_Toc172149733)

[34.Justify Content 47](#_Toc172149734)

[34.1 Desafío 47](#_Toc172149735)

[35.Align Items 48](#_Toc172149736)

[35.1 Desafío 49](#_Toc172149737)

[36.El centro perfecto 50](#_Toc172149738)

[36.1 Desafío 50](#_Toc172149739)

[37.Desafío Flex Box 51](#_Toc172149740)

[37.1 Desafío 51](#_Toc172149741)

# 1.Introducción

**CSS** (Cascading Style Sheets) es el lenguaje utilizado para **diseñar** páginas web.

En este curso nos sumergiremos en los fundamentos de este lenguaje, ¡después de completarlo tendrás las herramientas para diseñar tu próximo sitio web!

Antes de comenzar, debe tener conocimientos básicos de HTML, si no los tiene, complete el curso de HTML.

¡Feliz codificación!

# 2.Sintaxis

CSS es un tipo especial de lenguaje, consiste en **reglas**.

Cada regla tiene un **selector** y un **bloque de declaración**, en el siguiente formato,

selector {  
    bloque de declaración   
}

El selector selecciona el elemento HTML al que desea aplicar estilo.

El bloque de declaración contiene una o más declaraciones separadas por un punto y coma ().;

Cada declaración incluye el nombre y el valor de la propiedad separados por dos puntos (:), en el siguiente formato:

name: value;

Para concluir, aquí hay un ejemplo completo,

div {  
    background: blue;  
    width: 50px;  
    height 30px;  
}

¡Más adelante nos sumergiremos en cada detalle!

# 2.1 Desafío

Principiante

¡Vamos a escribir tu primer código CSS!

Se le dan dos archivos, index.html y styles.css.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

    <body>

        <p>

            Introducción a CSS

        </p>

    </body>

</html>

CSS

/\* Write CSS code here \*/

Siga los pasos:

1. Navegar/abrir archivo styles.css: este es el archivo CSS (archivos con sufijo .css)
2. Copie y pegue el siguiente código en el archivo

p {  
    color: red;  
    font-size: 24px  
}

Observe que el texto se vuelve más grande y rojo.

# 3.CSS externo

Así que tienes el conocimiento de HTML básico, ¿cómo puedes agregar estilo a tu página web?

Tres opciones:

* CSS externo
* CSS interno
* CSS en línea

**El CSS externo** crea un archivo CSS (archivo con sufijo .css) y lo vincula a la página del archivo HTML. ¡Utilizaremos a lo largo de este curso principalmente esta opción!

Por ejemplo, para vincular el archivo a su archivo HTML, use la siguiente sintaxis, styles.css

<link rel="stylesheet" href="styles.css">

Dentro de la etiqueta <head>, por lo que en un sitio web completo se verá así,

<!DOCTYPE html>  
<html>  
<head>  
    <link rel="stylesheet" href="styles.css">  
</head>  
<body>  
  
    <!-- More HTML code here -->  
</body>  
</html>

También puedes enlazar varios archivos CSS de esta manera, ¡sólo tienes que colocar varias etiquetas de enlace!

# 3.1 Desafío

Principiante

Se le da un archivo HTML (index.html) y un archivo CSS (styles.css), su trabajo es vincular el archivo CSS al HTML utilizando la técnica CSS externa.



HTML

<!DOCTYPE html>

<html>

<head>

</head>

<body>

    <p>Introduction to CSS</p>

</body>

</html>

CSS

P {

    color: blue;

}

*Una vez hecho esto, el texto debería volverse azul.*

# 4.CSS interno

**El CSS interno** es una forma de escribir el código CSS dentro del archivo HTML.

Para ello utilizamos la etiqueta (dentro de la etiqueta head), de la siguiente manera,<style>

HTML

<!DOCTYPE html>  
<html>  
<head>  
    <style>  
        body {  
            background-color: yellow;  
        }  
          
        /\* Más código CSS aquí \*/  
    </style>  
</head>  
<body>  
    <h1>Introduction to CSS</h1>  
</body>  
</html>

Como puedes ver arriba, ¡sólo tienes que escribir tu código CSS dentro de la etiqueta de estilo!

# 4.1 Desafío

Principiante

Agregue el siguiente código CSS al código HTML dado:

CSS

body {

background-color: black;

}

h1 {

color: white;

}

**¡Usa la técnica CSS interna**!

HTML

<!DOCTYPE html>

<html>

<head>

</head>

<body>

    <h1>Introdución a CSS</h1>

</body>

</html>

# 5.CSS en línea

**El CSS en línea** es una forma de aplicar estilo a un elemento HTML específico.

Se hace agregando la propiedad al elemento, la sintaxis es un poco diferente porque no necesitamos el selector ahora (seleccionamos el elemento directamente),style

<div style="width:50px; height:50px; background:green;"/>

En el ejemplo anterior, añadimos CSS en línea al elemento div.

***Aviso:****debe separar las declaraciones de estilo con punto y coma*

# 5.1 Desafío

Principiante

Aplique estilo a la etiqueta que se proporciona en el código HTML mediante CSS en línea, con los siguientes estilos:div

width:50px; height:50px; background:green;

<!DOCTYPE html>

<html>

    <body>

        <div></div>

    </body>

</html>

# 6.Introducción

Recordatorio de una regla en sintaxis CSS,

selector {  
    bloque de declaración  
}

**El selector CSS** selecciona los elementos HTML a los que desea **aplicar estilo**.

Profundizaremos en algunos selectores simples y más utilizados.

*Nota: ¡puede seleccionar varios elementos con un selector!*

# 7.Por etiqueta

HTML se construye a partir de **etiquetas** div, p, input, span y muchas más.

Para seleccionar por etiqueta HTML, especifique la etiqueta que desea seleccionar,

p {  
    color: blue;  
}

En el ejemplo anterior, todos los elementos p (párrafos) se seleccionan y se les aplica un estilo de color azul.

# 7.1 Desafío

Principiante

En el archivo CSS, seleccione todos los elementos que se dan en el archivo HTML y agréguele la siguiente propiedad CSS: div

HTML

border: 2px solid black;

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div>

        <p>”Algo de texto”</p>

        <div>

            <p>”Algo de más de texto”</p>

        </div>

    </div>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 8.Por ID

En HTML se le pueden dar muchos atributos a cada elemento, uno de los atributos es id.

¡El id de un elemento es **único** en toda la página!

Por ejemplo, si tiene el siguiente elemento:

<div id="myelement"></div>

Para seleccionar ese elemento **específico** con el id, use el símbolo de almohadilla (#), seguido del id deseado.

#myelement{  
    text-align: center;  
    color: red;  
}

En el ejemplo anterior, selecciona el elemento específico con id myelement y le da estilo.

# 8.1 Desafío

Principiante

Tenga en cuenta que cada uno de los elementos del archivo HTML ya tiene un identificador único.h1

Selecciona el **segundo** elemento h1 y añádele el siguiente estilo,

text-align: center;  
color: red;

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <h1 id="header1">Bienvenido</h1>

    <h1 id="header2">A CSS</h1>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 9.Por clase

Otro atributo HTML es class.

Una clase es compartida por muchos elementos, mientras que un id es único sólo para un elemento.

Supongamos que tiene los siguientes elementos:

<body>  
    <div class="box"> </div>  
    <div class="box"> </div>  
    <div class="box"> </div>  
</body>

Para seleccionar elementos por su class, escriba un punto (.), seguido del nombre de la clase deseada.

.box {  
    background: cyan;  
    border: 1px solid black;  
}

En el ejemplo anterior, se seleccionan todos los elementos con clase box y se les aplica estilo.

# 9.1 Desafío

Principiante

Se le proporcionan archivos HTML y CSS.

Diseñe los elementos con el nombre de la clase con box:

background: cyan;  
padding: 16px;

Diseñe los elementos con el nombre de la clase con text:

text-align: center;

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="box">

        <h2 class="text">

            Welcome

        </h2>

        <p class="text">

            Lorem ipsum dolor sit amet, consectetur adipiscing elit. In consectetur ornare libero vel ultrices. Vivamus sit amet purus tellus. Sed gravida mauris mollis dapibus fermentum. Nulla sed consequat magna. Aliquam enim augue, finibus quis tellus nec, feugiat iaculis mi. Ut aliquet nunc ut enim tincidunt dictum. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam dapibus ipsum facilisis, sodales arcu quis, semper dolor. Morbi ultricies nibh feugiat lacinia egestas. Curabitur et convallis neque. Nam nulla enim, malesuada in diam non, pharetra consectetur libero. Aenean eget justo ut neque tempor fermentum. Integer feugiat suscipit erat eget faucibus.

        </p>

    </div>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 10.Múltiples opciones

Es posible agrupar varios selectores al mismo tiempo, por ejemplo, id1 y id2 son estilos exactamente iguales:

#id1 {  
    color: red;  
    font-size: 12px;  
}  
  
#id2 {  
    color: red;  
    font-size: 12px;  
}

Usando el **selector de agrupación**, puede escribir lo anterior de la siguiente manera:

#id1, #id2 {  
    color: red;  
    font-size: 12px;  
}

¡De esta manera puedes minimizar tu código!

Como puede ver arriba, para agrupar varios selectores, simplemente sepárelos con comas.

# 10.1 Desafío

Principiante

¡Recapitulemos la selección!

El código HTML que se le proporciona ya incluye clases y atributos id, revíselo y asegúrese de detectar todos los elementos.

Las tareas:

- Selecciona la etiqueta body y dale estilo con lo siguiente,

background-color: blue;  
padding: 32px;

- Seleccione el id container y estilícelo con lo siguiente,

border: 1px solid white;  
color: white;  
padding: 8px 16px;

- Seleccione el id header y estilícelo con lo siguiente,

margin: 0;

- Selecciona la clase bold-text y dale estilo con lo siguiente,

font-weight: bold;  
border: 1px solid white;  
padding: 2px 4px;

- Selecciona la clase list-item y dale estilo con lo siguiente,

margin-bottom: 16px;

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="container">

        <h1 id="header">

            Simple Selectors

        </h1>

        <ul>

            <li class="list-item">Tag Selectors, for example - <span class="bold-text">div</span></li>

            <li class="list-item">Id Selectors, for example - <span class="bold-text">#myid</span></li>

            <li class="list-item">Class Selectors, for example - <span class="bold-text">.myclass</span></li>

        </ul>

    </div>

</body>

</html>



CSS

/\* Write CSS code here \*/

# 11.Colores

¡Hablemos de **los colores**!

En CSS hay nombres de color predefinidos como, blue, red u orange

Hay exactamente 140 colores predefinidos, compruébalos todos [aquí](https://htmlcolorcodes.com/color-names/).

Si necesitas más colores puedes usar colores por valores:

* Valores hexadecimales - #2a6e45#ed1
* Valores RGB - rgb(123, 23, 12)
* Valores HSL - hsl(9, 80%, 32%)

En este curso no profundizaremos en los diferentes valores de color.

*Nota: Los valores HEX son los más utilizados, puede declararlos comenzando con hash seguido de 6 o 3 caracteres hexadecimales (0 - 9, a - e)*

También hay un nombre de color especial: transparent es el color predeterminado de muchos fondos.

# 11.1 Desafío

Principiante

¡Colorea el párrafo en cualquier color que no sea negro!

Para colorear el texto, use la propiedad CSS color. (¡Aprenderemos más sobre esto más adelante!)

¿Cómo puedes seleccionarlo sin cambiar el código HTML?



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <p>

        Yo sé CSS!

    </p>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 12.Fondo

Cualquier elemento HTML puede tener un fondo.

Para agregar color de fondo, use la propiedad background-color,

div {  
    background-color: #333;  
}

Lo anterior dará un color de fondo a todos los elementos.#333div

También puede utilizar la propiedad background para especificar un color de fondo.

# 12.1 Desafío

Principiante

Asigne al párrafo cualquier color de fondo que no sea (predeterminado transparent).

Utilice el **selector de id**para seleccionar el párrafo.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <p id="text">

        Yo sé CSS!

    </p>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 13.Tamaños

En CSS aplicamos **tamaños** a muchas propiedades.

CSS tiene varias unidades diferentes para expresar un tamaño.

**Unidades absolutas**

Las unidades absolutas se representan de la misma manera en todos los dispositivos y tamaños

| **Unidad** | **Descripción** | **Ejemplo** |
| --- | --- | --- |
| px | Píxeles | 8px |
| cm | Centímetros | 2.5cm |
| mm | Milímetros | 3mm |
| in | Pulgadas | 5in |

**Los píxeles** son la unidad absoluta más utilizada, ya que es relativa al tamaño del dispositivo.

**Unidades relativas**

Las unidades relativas se representan en relación con un elemento específico de la máquina.

| **Unidad** | **Descripción** | **Ejemplo** |
| --- | --- | --- |
| % | Tamaño relativo al elemento **primario** | 80% |
| vh | En relación con la altura del dispositivo (Altura de vista) | 20vh |
| vw | En relación con el ancho del dispositivo (ancho de vista) | 35vw |
| em | En relación con el tamaño de fuente del elemento | 5em |

# 13.1 Desafío

Principiante

Seleccione el único div de la página, estilícelo con las propiedades width y height, asígnele valores diferentes a 0.

*Profundizaremos en el ancho y la altura en la próxima lección*

Tenga en cuenta que ya tiene un CSS en línea en el div que le da estilo con un color de fondo.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div style="background: yellow;"/>

</body>

</html>



CSS

/\* Write CSS code here \*/

# 14.Width & Height

En la lección anterior, ha utilizado las propiedades de CSS width y height.

Como probablemente hayas adivinado, las propiedades width y height se utilizan para establecer el ancho y el alto del elemento.

#box {  
    width: 30%;  
    height: 64px;  
}

En el ejemplo anterior se aplica estilo al elemento con id box con 30% que es del ancho y alto de 64px.

# 14.1 Desafío

Principiante

Se le da una página HTML con tres divs anidados, su tarea es darles width y height.

¡Intente configurar los divs internos con tamaño relativo usando %, selecciónelos por sus identificadores!

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="box1" style="background: pink;">

        <div id="box2" style="background: orange;">

            <div id="box3" style="background: red;"/>

        </div>

    </div>

</body>

</html>



CSS

/\* Write CSS code here \*/

# 15.Border

Agreguemos bordes.

Border tiene **tres propiedades CSS**:

1. border-width - Especifica la anchura del borde.
2. border-color - Especifica el color del borde.
3. border-style - Especifica el tipo de borde.

Algunos estilos de borde posibles,

* dotted - Borde punteado
* dashed - Borde discontinuo
* solid - Borde sólido (**el más utilizado**)
* double - Borde Deouble

También puede especificar todas las propiedades en una, utilizando la propiedad de borde simple,

div {  
    border: 4px solid red;  
}

El ejemplo anterior da estilo a todos los elementos div, es lo mismo que escribir:

div {  
    border-width: 4px;  
    border-style: solid;  
    border-color: red;  
}

# 15.1 Desafío

Principiante

¡Tu turno!

Dale estilo al párrafo con el **borde** que elijas.

*Esta es tu oportunidad de probar todos los estilos de borde posibles*

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <p style="padding: 8px;">

        Me encanta estudiar CSS!!

    </p>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 16.Tamaños de borde

También es posible estilizar cada lado del borde por separado (4 lados).

Para ello, utilice las siguientes propiedades:

**Estilo**

* border-top-style
* border-right-style
* border-bottom-style
* border-left-style

**Color**

* border-top-color
* border-right-color
* border-bottom-color
* border-left-color

**Ancho**

* border-top-width
* border-right-width
* border-bottom-width
* border-left-width

Al igual que border puede usar todas las propiedades del lado del borde usando border-top, border-right, border-bottom, border-left.

Por ejemplo

.container {  
    border-left-style: solid;  
    border-left-color: blue;  
    border-left-width: 5px;  
}

O en una versión más corta,

.container {  
    border-left: 5px solid blue;  
}

# 16.1 Desafío

Principiante

Asigne estilo al borde izquierdo del elemento div.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

    <body>

        <div id="container">

            <p>

                “Algo de texto”

            </p>

        </div>

    </body>

</html>

CSS

/\* Write CSS code here \*/

# 17.Desafío de propiedades

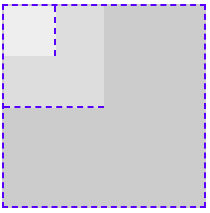
¡Recapitulemos!

# 17.1 Desafío

Principiante

Al igual que antes, se les dan tres divs anidados, asígneles a todos el borde, el color de fondo, el ancho y la altura, ¡pésela bien!

Por ejemplo:



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="box1">

        <div id="box2">

            <div id="box3"/>

        </div>

    </div>

</body>

</html>

CSS

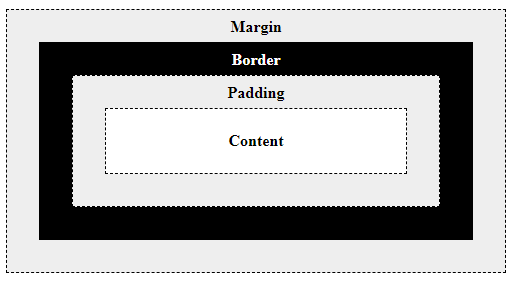
/\* Write CSS code here \*/

# 18.Introducción

Todos los elementos HTML se **tratan** como cuadros.  
El modelo de caja CSS es **básicamente** una caja alrededor de **cada** elemento HTML.

Consta de: margen, borde, relleno y el contenido.

La siguiente imagen ilustra el modelo de caja:



* **Contenido**: el contenido del cuadro donde se muestra el elemento.
* **Relleno:** despeja un área alrededor del contenido. El acolchado es transparente
* **Borde**: un borde que rodea el relleno y el contenido
* **Margen**: despeja un área fuera del borde. El margen es transparente

Los modelos de caja le permiten agregar bordes alrededor de los elementos y definir el espaciado entre los elementos.

# 19.Margen

La propiedad CSS margin se utiliza para crear **espacio** alrededor de los elementos, fuera de los bordes definidos.

Para especificar el margen de cada lado de un elemento, puede utilizar:

* margin-top
* margin-right
* margin-bottom
* margin-left

Por ejemplo

.container {  
    margin-top: 2px;  
    margin-right: 3px;  
    margin-bottom: 4px;  
    margin-left: 5px;  
}

Es lo mismo que,

.container {  
    margin: 2px 3px 4px 5px;  
}

El formato de sintaxis es margin: top right bottom left;

Si especifica solo dos valores, utilizará el siguiente formato, margin: top-bottom right-left

.container {  
    margin: 2px 4px;  
}

Es lo mismo que,

.container {  
    margin-top: 2px;  
    margin-right: 4px;  
    margin-bottom: 2px;  
    margin-left: 4px;  
}

Si especifica solo un valor, aplicará el mismo valor para todos los lados,

.container {  
    margin: 4px;  
}

¡Lo anterior se suma a todos los lados! margin 4px

# 19.1 Desafío

Principiante

Agregue margin de 16px al div. (margen de todos los lados de 16px)

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div>

        <p>

            Me encanta estudiar CSS!!

        </p>

    </div>

</body>

</html>

CSS

div {

    background-color: #333;

}

p {

    color: white;

}

# 20.Padding

La propiedad CSS padding se utiliza para generar **espacio** alrededor del contenido de un elemento, **dentro**de los bordes definidos.

padding tiene exactamente el mismo formato que margin.

Con cuatro valores,

.container {  
    padding: 2px 4px 6px 8px;  
}

O

.container {  
    padding-top: 2px;  
    padding-right: 4px;  
    padding-bottom: 6px;  
    padding-left: 8px;  
}

¡Es decir, margin también puedes tener solo dos valores o uno!

# 20.1 Desafío

Principiante

Agregue padding de 8px al div. (relleno desde todos los lados de 8px)

Y agregue margin de 0 a p.

*¡Notar! Los párrafos (p) tienen margen por defecto, es por eso que agregamos aquí margen cero*

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div>

        <p>

            Me encanta estudiar CSS!!

        </p>

    </div>

</body>

</html>

CSS

div {

    background-color: #333;

}

p {

    color: white;

}

# 21.Tamaño del box

box-sizing nos permite incluir el padding y border en el total width y height.

De forma predeterminada, el tamaño real (ancho o alto) es el tamaño + relleno + borde.

div {  
    width: 100px;  
    padding: 16px;  
    border: 1px solid black;  
}

En el ejemplo anterior, el ancho real del div será:

100 + 32 (16 relleno de cada lado) + 2 (1 borde de cada lado) = 134px

¡Y no 100px!

Al configurar box-sizing:, el relleno y el borde se incluirán en ancho y alto, border-box;

div {  
    width: 100px;  
    padding: 16px;  
    border: 1px solid black;  
    box-sizing: border-box;  
}

En el ejemplo anterior, el div será **exactamente** de 100px.

# 21.1 Desafío

Principiante

Se le da un recuadro rojo (div).

Complete los siguientes pasos en el box:

1. Agregue un relleno mayor que 0. (¿Creció?)
2. Agregue cualquier borde con un ancho mayor que 0. (¿Creció?)
3. Agregue de box-sizing border-box (¿Se encogió?)

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="box" />

</body>

</html>

CSS

#box {

    width: 100px;

    height: 100px;

    background-color: red;

}

# 22.Sombra del box

box-shadow se utiliza para aplicar sombras a un elemento.

La propiedad box-shadow utiliza el formato:

box-shadow: horizontal-offset vertical-offset blur-radius spread-radius color;

Por ejemplo

.container {  
    box-shadow: 10px 10px 8px 2px cyan;  
}

Lo anterior aplicará la sombra de la caja con desplazamiento 10px horizontal, 10px desplazamiento vertical, 8px radio de desenfoque, 2px radio de extensión y color cyan.

También es posible **omitir**el desenfoque o el radio de extensión.

Por ejemplo

.container {  
    box-shadow: 8px 8px 4px blue;  
}

El ejemplo anterior no establecerá el radio de propagación.

.container {  
    box-shadow: 8px 8px blue;  
}

El ejemplo anterior no establecerá el radio de desenfoque y propagación.

# 22.1 Desafío

Principiante

Se le da un box con un sombreado ya aplicado, su tarea es jugar con el box-shadow y cambiarlo, ¡hacerlo hermoso!

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="box" />

</body>

</html>

CSS

#box {

    background: blue;

    width: 120px;

    height: 120px;

    margin: 16px;

    box-shadow: 8px 8px 2px 2px #1e1e1e;

}

# 23.Border Radius

También es posible redondear los bordes de un elemento mediante la propiedad border-radius.

Por ejemplo

#box {  
    border: 1px solid black;  
    border-radius: 8px;  
}

En el ejemplo anterior se establece el radio del borde para todas las esquinas de 8px.

Para establecer cada esquina con un radio diferente, use el formato,

border-radius: top-left top-right bottom-right bottom-left;

Por ejemplo

#box {  
    border: 1px solid black;  
    border-radius: 8px 4px 0px 6px;  
}

# 23.1 Desafío

Principiante

Agregue un border-radius en #box.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="box">

        Me encanta estudiar CSS!!

    </div>

</body>

</html>

CSS

#box {

    padding: 8px;

    background-color: #eee;

    border: 1px solid black;

}

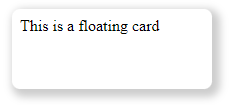
# 24.Desafío de modelado del box

# 24.1 Desafío

Principiante

Crea una **tarjeta flotante** usando HTML y CSS.

He aquí un ejemplo,



Crea un div **con algo de texto** dentro y agrega las siguientes propiedades (¡con los valores que elijas!):

* width y height
* margin y padding
* border-radius
* box-shadow

También es opcional usar las siguientes propiedades,

* box-sizing
* border
* background-color
* color (aún no se ha aprendido, pero se usa para cambiar el color del texto)



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <!-- Write HTML code here -->

</body>

</html>

CSS

/\* Write CSS code here \*/

# 25.Color del texto

¡Vamos a darle estilo al **texto**!

Ya has visto la propiedad color en lecciones anteriores.

La propiedad color se utiliza para establecer el color del texto.

p {  
    color: rgb(120, 65, 101);  
}

En el ejemplo anterior se aplicará estilo a todos los párrafos de la página con el color de texto de rgb(120, 65, 101)

La propiedad color funciona principalmente junto con la propiedad background-color, es importante establecer el **contraste** correcto entre el color de fondo y el color del texto.

Por ejemplo, el color del texto oscuro no debe estar sobre un color de fondo oscuro.

# 25.1 Desafío

Principiante

Dale un color de fondo y un texto en él div.

Como puedes ver, ¡el **contraste**no es el correcto!

Establezca el color del texto del párrafo en white.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div style="background: #1e1e1e;">

        <p>

            “Esto es algo de texto”

        </p>

    </div>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 26.Alineación de texto

text-align se utiliza para establecer la alineación **horizontal** de un texto.

Tres valores posibles:

* right - alineado a la derecha.
* left - alineado a la izquierda.
* center - Alineado al centro.

Por ejemplo

h1 {  
    color: red;  
    text-align: center;  
}

# 26.1 Desafío

Principiante

¡Configura el encabezado para que esté alineado en el **centro**!

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <h1>Bienvenido</h1>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 27.Decoración de texto

text-decoration se utiliza para añadir una **línea de decoración** al texto.

Los valores posibles para la decoración de texto:

* underline - Añade una línea de decoración debajo del texto.
* overline - Añade una línea de decoración sobre el texto.
* line-through - Añade una línea de decoración a través del texto.
* none - Sin línea de decoración (por defecto).

Por ejemplo:

#deletedText {  
    text-decoration: line-through;  
}

También es posible agregar color como segundo valor,

#deletedText {  
    text-decoration: line-through red;  
}

El tercer valor especifica el estilo de la línea, los valores posibles son:

* solid -predeterminado
* double
* dotted
* dashed
* wavy

Ejemplo con tres valores,

#deletedText {  
    text-decoration: line-through red double;  
}

El cuarto valor especifica el grosor de la línea,

#deletedText {  
    text-decoration: line-through red double 2px;  
}

Puede establecerlos todos por separado también utilizando las siguientes propiedades,

* text-decoration-line (el tipo de línea)
* text-decoration-color
* text-decoration-style
* text-decoration-thickness

# 27.1 Desafío

Principiante

Agregue una línea superior al texto dado, ¡intente diseñarlo con diferentes valores!

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <p>

        Me encanta estudiar CSS!!

    </p>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 28.Sombra de texto

Como en la sombra del cuadro, también es posible agregar sombra al texto usando la propiedad text-shadow.

El formato es el mismo que el de box-shadow, **excepto** que no tiene:spread-radius

text-shadow: horizontal-offset vertical-offset blur-radius color;

Por ejemplo

.text {  
    text-shadow: 8px 8px 4px #ccc;  
}

*Al igual que con la sombra de la caja, aquí también es posible****omitir****el desenfoque y el radio de extensión.*

# 28.1 Desafío

Principiante

Añade sombra al texto.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <h2 class="text">

        Me encanta estudiar CSS!!

    </h2>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 29.Fuentes de texto

Elegir la fuente correcta tiene un gran impacto en la forma en que los usuarios experimentan un sitio web.

La propiedad CSS font-family establece la fuente del texto,

.text {  
    font-family: Arial;  
}

En CSS hay múltiples fuentes predefinidas, aquí hay un par de ellas:

* Times New Roman
* Georgia
* Garamond
* Arial
* Verdana
* Helvetica
* Mensajero Nuevo
* Consola Lucida
* Mónaco
* Script de pincel MT
* Escritura lúcida
* Copperplate
* Papiro

No todos los navegadores son compatibles con todas las fuentes, por lo que suele ser una práctica común establecer varias fuentes.

Cada fuente después de la primera fuente definida es una "fuente alternativa" en caso de que la fuente anterior no sea compatible con el navegador.

 .text {  
    font-family: Papyrus, Copperplate, "Courier New", Arial;  
}

***Nota****: Si el nombre de la fuente tiene más de una palabra, debe estar entre comillas, como: "Times New Roman".*

# 29.1 Desafío

Principiante

Configura el texto con una fuente especial, ¡asegúrate de que funcione en tu navegador!

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <p>

        Me encanta CSS!!

    </p>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 30.Desafío de texto

# 30.1 Desafío

Principiante

Dale estilo a este encabezado con todo lo que sabes.

Agregue las siguientes propiedades:

* text-shadow
* font-family
* text-align de center

Opcional:

* color
* text-decoration

Ejemplo



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <h1>

        I love CSS!

    </h1>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 31.Introducción

Tal vez hayas notado que los juegos de diseño son muy limitados.

div es un diseño vertical de forma predeterminada y la característica faltante más sencilla es colocar el contenido en un diseño horizontal.

Esta es la razón por la que se introdujo el tipo de pantalla flex.

*Flexbox es una propiedad súper utilizada en CSS!*

# 32.Flex Box

Para definir un flexbox, en el **contenedor** divañade:

display: flex;

*El diseño de la caja flexible es horizontal de forma predeterminada.*

# 32.1 Desafío

Principiante

Haz que el recipiente div sea una flexbox.

*¡Observe el cambio en la orientación del diseño!*

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="container">

        <div class="innerBox">

            1

        </div>

        <div class="innerBox">

            2

        </div>

        <div class="innerBox">

            3

        </div>

    </div>

</body>

</html>

CSS

.innerBox {

    padding: 16px;

    background-color: #ccc;

    margin: 8px;

}

# 33.Dirección del flex

¡Profundicemos en las propiedades de flexbox!

flex-direction define en qué dirección el contenedor apila los elementos flexibles.

Los valores de esta propiedad son:

* row - Apila los elementos horizontalmente (de izquierda a derecha)
* row-reverse - Apila los elementos horizontalmente al revés (de derecha a izquierda)
* column - Apila los elementos verticalmente (de arriba a abajo)
* column-reverse - Apila los elementos verticalmente en reversa (de abajo hacia arriba)

# 33.1 Desafío

Principiante

1. Haga que el recipiente div sea una caja flexible (¿no está seguro? consulte la lección anterior).

2. Pruebe **todos los** valores de flex-direction posibles en el contenedor div. Al final, usa el valor row-reverse.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="container">

        <div class="innerBox">

            1

        </div>

        <div class="innerBox">

            2

        </div>

        <div class="innerBox">

            3

        </div>

    </div>

</body>

</html>

CSS

.innerBox {

    padding: 16px;

    background-color: #ccc;

    margin: 8px;

}

# 34.Justify Content

justify-content se utiliza para alinear los elementos del cuadro flexible en el **eje principal**.

Si el diseño del cuadro row flexible es horizontal (row-reverse o justify-content), alinea los elementos en el eje horizontal.

Si el diseño del cuadro flexible column es vertical (column-reverse o justify-content), alinea los elementos en el eje vertical.

Valores para:justify-content

* center - Alinea los elementos en el centro del contenedor
* flex-start - Alinea los elementos al principio del contenedor (predeterminado)
* flex-end - Alinea los elementos en el extremo del contenedor
* space-around - Muestra los elementos con espacio antes, entre y después de ellos
* space-between - Muestra los elementos con espacio entre ellos
* space-evenly - muestra los elementos con espacio antes, entre y después de ellos, todos los espacios son iguales

# 34.1 Desafío

Principiante

Se le da un contenedor de flexbox, con color blue.

Pruebe todos los valores de justify-content en el contenedor, al final establézcalo en space-evenly.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="container">

        <div class="innerBox">

            1

        </div>

        <div class="innerBox">

            2

        </div>

        <div class="innerBox">

            3

        </div>

    </div>

</body>

</html>

CSS

.innerBox {

    padding: 16px;

    background-color: #ccc;

    margin: 8px;

}

.container {

    background-color: blue;

    display: flex;

}

# 35.Align Items

align-items se utiliza para alinear los elementos de la flexbox en el **eje transversal**.

Si el diseño del cuadro flexible row es horizontal (row-reverse o align-items), alinea los elementos en el eje vertical.

Si el diseño del cuadro flexible column es vertical (column-reverse o align-items), alinea los elementos en el eje horizontal.

Valores para:align-items

* center - Alinea los elementos en el centro del contenedor
* flex-start - Alinea los elementos en la parte superior del contenedor
* flex-end - Alinea los elementos en la parte inferior del contenedor
* stretch - Alinea los elementos para llenar el contenedor (predeterminado)

# 35.1 Desafío

Principiante

Se le da un contenedor de flexbox, con color blue y altura de 200px.

Pruebe todos los valores de align-items en el contenedor, al final establézcalo en flex-end.

HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="container">

        <div class="innerBox">

            1

        </div>

        <div class="innerBox">

            2

        </div>

        <div class="innerBox">

            3

        </div>

    </div>

</body>

</html>

CSS

.innerBox {

    padding: 16px;

    background-color: #ccc;

    margin: 8px;

}

.container {

    background-color: blue;

    height: 200px;

    display: flex;

}

# 36.El centro perfecto

Usando flex box es posible **centrar todo tanto horizontal como verticalmente**,

#flex-container {  
    display: flex;  
    align-items: center;  
    justify-contet: center;  
}

Como puedes ver arriba, simplemente crea un contenedor de flex box con propiedades align-items: center; y justify-contet: center;.

# 36.1 Desafío

Principiante

**Centrar perfectamente todo.**

* Cada div dentro de div
* Cada texto dentro de div

*¡Recordar! Es necesario convertir cada contenedor div en una flex box.*



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div class="container" style="width: 300px; height: 300px; background: yellow;">

        <div class="innerContainer" style="width: 80px; height: 80px; background: blue;">

            <b style="color: white">Hello</b>

        </div>

        <div class="innerContainer" style="width: 80px; height: 80px; background: red">

            <b style="color: white">CSS</b>

        </div>

    </div>

</body>

</html>

CSS

/\* Write CSS code here \*/

# 37.Desafío Flex Box

¡Recapitulemos CSS!

# 37.1 Desafío

Principiante

Se le da una estructura HTML, su tarea es hacer el siguiente diseño:

https://firebasestorage.googleapis.com/v0/b/coddy-for-creators.appspot.com/o/courses%2F9yHGYzsM32tUfMj9EwHy%2F63HDoNWhY2GAOIRYqQG9%2FScreenshot%20from%202023-03-27%2016-51-30.png?alt=media&token=3ceeb6c0-958f-41f0-9bec-06c53fddfc4a

No cambies el CSS dado o los identificadores de los divs, ¡puedes agregar el nombre de la clase y más código CSS!

**Opcional - Dale un buen estilo**

He aquí un ejemplo,



HTML

<!DOCTYPE html>

<html>

<head>

    <link rel="stylesheet" href="styles.css">

</head>

<body>

    <div id="b1">

        <div id="b2">

            <span>I</span>

        </div>

        <div id="b3">

            <span>❤</span>

        </div>

        <div id="b4">

            <div id="b5">

                <span>C</span>

            </div>

            <div id="b6">

                <span>S</span>

            </div>

            <div id="b7">

                <span>S</span>

            </div>

        </div>

    </div>

</body>

</html>

CSS

#b1 {

    height: 80px;

}

#b4 {

    width: 40px;

}