



**Intro a CSS**



1. ¿Qué es CSS?
2. Vincular a HTML
3. Reglas y cascada
4. Selectores
5. Modelo de caja
6. Propiedades
7. Layout: Flexbox
8. Layout: Grid
9. Diseño responsive



# ¿Qué es CSS?

**CSS (Cascading Style Sheets), es un lenguaje de diseño gráfico para definir y crear la presentación de un Documento estructurado escrito en HTML. Su principal objetivo es separar el contenido del documento de su presentación visual, incluyendo aspectos como los colores, fuentes, y la distribución de los elementos. Esto permite mejorar la accesibilidad, proporcionar más flexibilidad y control en la especificación de características de presentación, y reducir la complejidad y la repetición en el contenido estructural.**

# Vinculación a HTML

El uso de reglas de CSS en HTML se puede hacer de varias maneras, concretamente 3:

- **En línea**

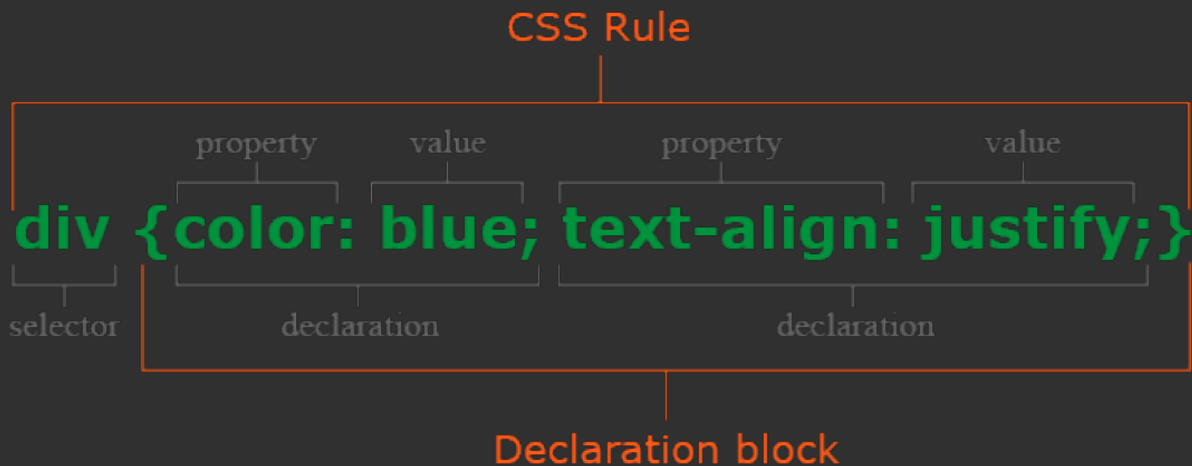
```
<h1 style="text-align:center">CSS</h1>
```

- **Incrustación**

```
<html>
  <head>
    <style type="text/css">
      h1 {text-align:center;}
    </style>
  </head>
  <body>
    <h1>CSS</h1>
  </body>
</html>
```

- **Vinculación**

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



Formato de una regla

# Cascada

La cascada en CSS es un sistema de prioridades que determina qué estilos se aplican a un elemento cuando hay reglas contradictorias. Este mecanismo se base en tres factores principales: importancia, especificidad y orden.

La importancia se refiere a cómo las declaraciones `!important` tienen prioridad sobre las que no la tienen.

La especificidad compara selectores; aquellos más específicos (por ejemplo, es más específico el selector de id que el de clase) tienen prioridad.

Finalmente, el orden de la regla implica, y de ahí el nombre de cascada, que la última en llegar es la prioritario, si el resto de elementos (importancia y especificidad son las mismas).

También hay que tener en cuenta que si existen reglas dentro del código html (incrustado) será preferente sobre las reglas vinculadas.



# Selectores

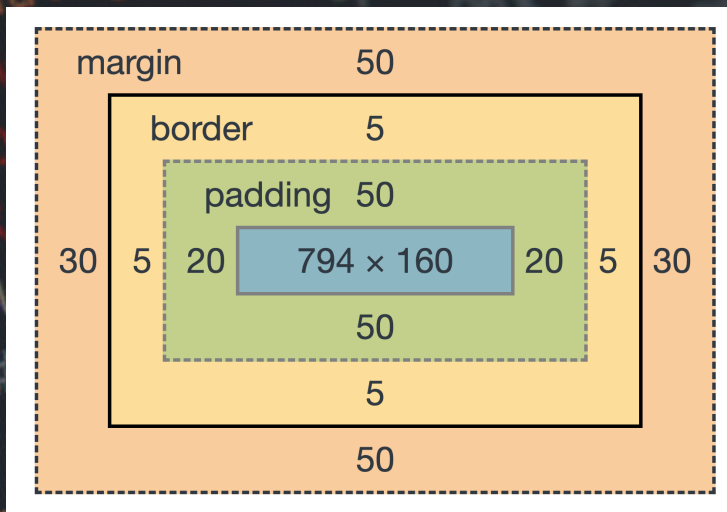
- universal
- de tipo
- de clase
- de id
- contexto (descendientes, padres, hermanos)
- propiedades
- pseudoclases
- pseudoelementos

```
/* Todos los elementos de la pagina */  
{ font: 1em/1.3 Arial, Helvetica, sans-serif; }  
  
/* Todos los parrafos de la pagina */  
{ color: #555; }  
  
/* Todos los párrafos contenidos en #primero */  
{ color: #336699; }  
  
/* Todos los enlaces la pagina */  
{ color: #CC3300; }  
  
/* Los elementos "em" contenidos en #primero */  
{ background: #FFFCC; padding: .1em; }  
  
/* Todos los elementos "em" de clase "especial" en toda la pagina */  
{ background: #FFCC99; border: 1px solid #FF9900; padding: .1em; }  
  
/* Elementos "span" contenidos en .normal */  
{ font-weight: bold; }
```



# Ahora practica





Modelo de caja

# Propiedades (Cheatsheets)

- [Codecademy](#)

1



**Ahora practica**



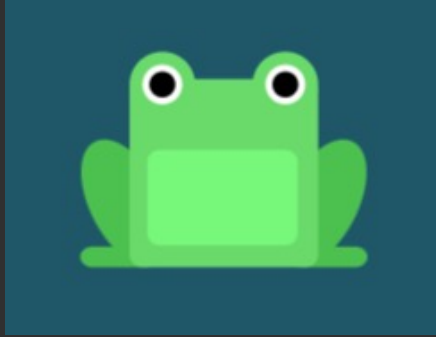
# Flexbox

**Flexbox, formalmente conocido como el Modelo de Caja Flexible, es un conjunto de propiedades CSS diseñadas para facilitar la distribución de espacio y alineación de elementos en interfaces de usuario de manera eficiente y predictiva.**

**Introducido en CSS3, Flexbox ofrece una solución a los problemas comunes de diseño web, como la alineación vertical, el espaciado igual entre elementos, y la capacidad de ajustar los elementos de la interfaz de usuario de manera dinámica para diferentes tamaños de pantalla. Funciona definiendo un contenedor flex (flex container) que altera el comportamiento de sus elementos hijos (flex items). Estos elementos pueden expandirse para llenar el espacio disponible, contraerse para evitar salirse de su contenedor, o distribuirse con espaciado uniforme a lo largo del eje principal o transversal del contenedor.**

**Con Flexbox, los desarrolladores pueden crear diseños complejos y responsivos con menos código y mayor flexibilidad, sin necesidad de floats o posicionamientos complicados.**

**Cheatsheet: <https://flexbox.malven.co/>**



<https://flexboxfroggy.com/#es>



**Ahora practica**



# Grid

CSS Grid Layout es un poderoso sistema bidimensional para el diseño de interfaces web, que permite a los desarrolladores crear complejas estructuras de diseño con relativa facilidad. Introducido como parte de CSS3, Grid proporciona un método para definir tanto filas como columnas, controlando la distribución de espacio y el alineamiento de los elementos en un diseño bidimensional.

A diferencia de Flexbox, que es más adecuado para diseños lineales (unidimensionales), Grid es capaz de manejar tanto la dimensión vertical como la horizontal simultáneamente, lo que lo hace ideal para construir layouts complejos que requieren un control preciso sobre las áreas de diseño. Los desarrolladores pueden definir explícitamente las áreas de grid, asignar elementos a estas áreas, y controlar el tamaño, la posición y el espaciado de los elementos dentro del grid.

Cheatsheet: <https://grid.malven.co/>



<https://cssgridgarden.com/#es>



**Ahora practica**

# Diseño responsive

## Necesario:

```
<meta name="viewport" content="width=device-width,initial-scale=1"/>
```

## Recomendable: Diseño Mobile First

## Media queries:

```
.column {  
    width: 100%;  
}  
  
@media (min-width: 600px) {  
    .column {  
        width: 50%;  
    }  
}
```



# Learnbydō

Cuatroochenta CAR

[learnbydo@cuatroochenta.com](mailto:learnbydo@cuatroochenta.com)