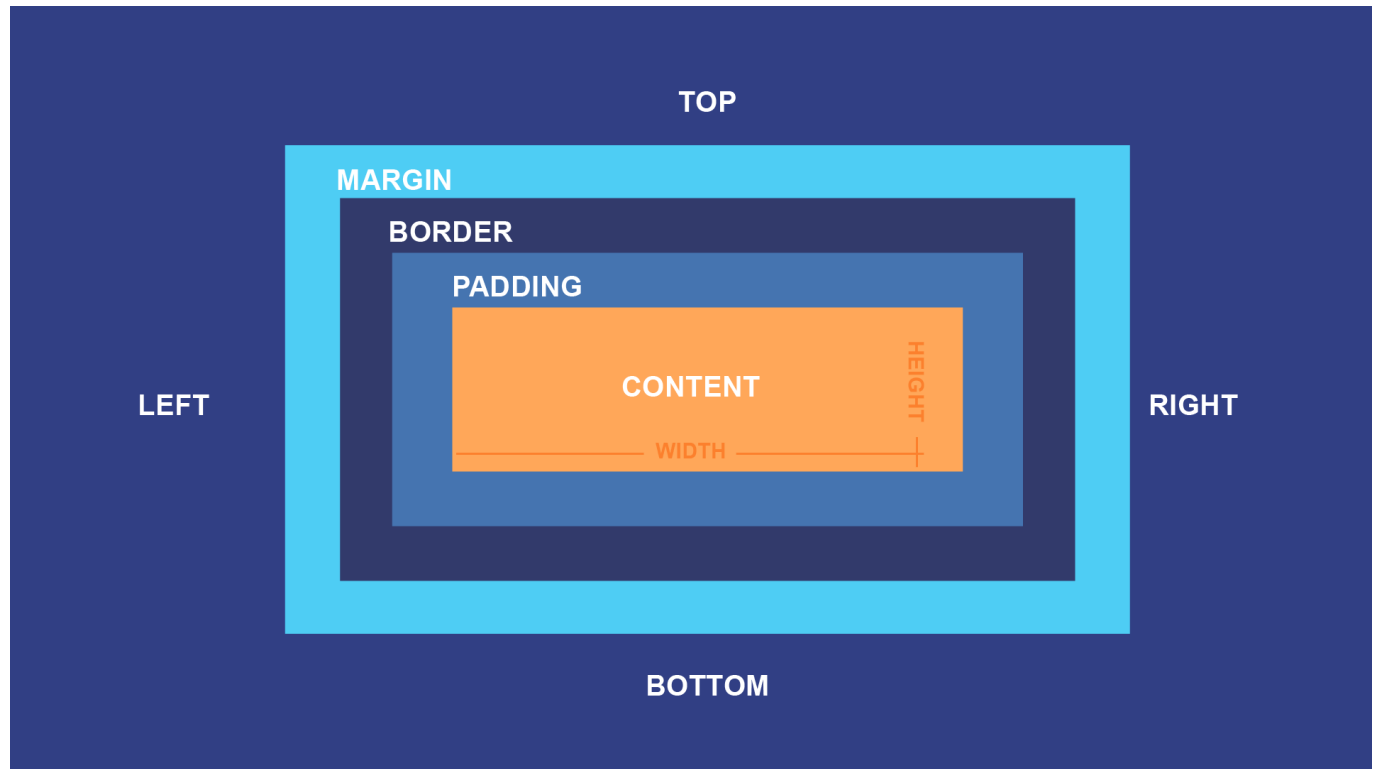


# 01-03 CSS POSITION (by bluuweb)

## Modelo de caja (box model)

Todo en CSS tiene una caja alrededor, y comprender estas cajas es clave para poder crear diseños con CSS o para alinear elementos con otros elementos.



## box-sizing

Para mantener el ancho, sin importar la cantidad de relleno, puede usar la propiedad `box-sizing: border-box`. Esto hace que el elemento mantenga su ancho; Si aumenta el relleno, el espacio de contenido disponible disminuirá.

- [css-tricks](#)

```
html {  
  box-sizing: border-box;  
}  
*,  
*:before,  
*:after {  
  box-sizing: inherit;  
}
```

## Display

- La propiedad CSS display especifica si un elemento es tratado como block o inline element

- [display](#)
- [¿De línea o de bloque?](#)

## display: block

- Un elemento de bloque siempre comienza en una nueva línea y ocupa todo el ancho disponible.
- Permite establecer un ancho y alto.
- Se respetan los márgenes / rellenos superiores e inferiores.

## display:inline

- No respeta el ancho ni alto.
- Padding se desborda cubriendo otros elementos.
- No respeta los márgenes verticales (top-bottom).

## display: inline-block

- **permite establecer un ancho y alto** en el elemento.
- Se respetan los márgenes / rellenos superiores e inferiores
- En comparación con `display: block`: la principal diferencia es que `display: inline-block` **no agrega un salto de línea después del elemento**, por lo que el elemento puede sentarse junto a otros elementos.

## Práctica Navbar

- [Pseudo-classes](#): Una pseudoclase CSS es una palabra clave que se añade a los selectores y que especifica un estado especial del elemento seleccionado. Por ejemplo, `:hover` aplicará un estilo cuando el usuario haga hover sobre el elemento especificado por el selector.
- [:hover](#):

```
<nav class="navbar">
  <a href="#" class="navbar-link active">Inicio</a>
  <a href="#" class="navbar-link">Servicios</a>
  <a href="#" class="navbar-link">Nosotros</a>
  <a href="#" class="navbar-link">Contacto</a>
</nav>
<main>
  <h1>Inicio</h1>
</main>
```

```
html {
  box-sizing: border-box;
}
*,
*:before,
*:after {
  box-sizing: inherit;
}
```

```
body {
  font-family: "Roboto", sans-serif;
}

.navbar {
  background-color: #293462;
  padding: 1rem;
  text-align: center;
}

.navbar-link {
  display: inline-block;
  background-color: #d61c4e;
  color: white;
  padding: 2rem 1.5rem;
  text-decoration: none;
  font-weight: 300;
  margin-right: 1rem;
  margin-left: 1rem;
}

.navbar-link:hover {
  background-color: #4f63b6;
}

.active {
  background-color: #4f63b6;
}
```

## float

- **float:** La propiedad CSS float ubica un elemento al lado izquierdo o derecho de su contenedor, permitiendo a los elementos de texto y en línea aparecer a su costado. El elemento es removido del normal flujo de la página, aunque aún sigue siendo parte del flujo (a diferencia del posicionamiento absoluto).
- float no se utiliza para hacer columnas, para eso tenemos mejores alternativas como flexbox o grid.

```
<body>
  <main>
    <h1>Float</h1>
    
    <p>
      Lorem ipsum dolor sit amet consectetur adipisicing elit.
      Officia soluta, aliquid fuga quo reprehenderit, reiciendis
      amet unde dolor quis aspernatur libero voluptatibus odio
      animi laborum quaerat enim mollitia delectus? Commodi!
    </p>
  </main>
</body>
```

```
h1 {
  text-align: center;
  text-transform: uppercase;
}

.img-300 {
  width: 300px;
  float: left;
  margin-right: 1rem;
  margin-bottom: 1rem;
}
```

- Hay dos tipos de elementos en línea: elementos en línea reemplazados y elementos en línea no reemplazados. En general, los elementos no reemplazados son aquellos cuyo contenido está contenido en el documento, mientras que los **elementos reemplazados** son aquellos cuyo contenido está fuera del documento (img).
- **La propiedad de altura y ancho se aplica a los elementos en línea reemplazados.**
- [stackoverflow](#)

## CSS Flexbox

- [css-tricks](#)
- [Mozilla](#)
- <https://cssreference.io/>

## Contenedor

Envolvemos todos nuestros items dentro de un div contenedor, el cual tendrá la clase `display: flex`

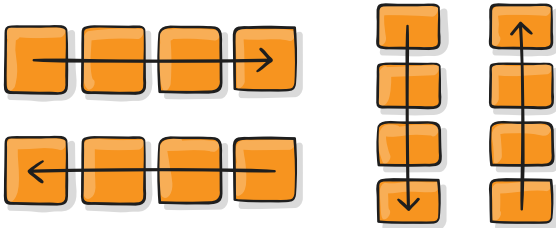
```
<div class="border flex">
  <div class="border item">item 1</div>
  <div class="border item">item 2</div>
  <div class="border item">item 3</div>
</div>
```

```
.border {
  border: 2px solid #000;
  border-radius: 10px;
}

.item {
  background-color: tomato;
  text-align: center;
  width: 200px;
  padding: 10px;
  margin: 10px;
```

```
}  
  
.flex {  
  display: flex;  
}
```

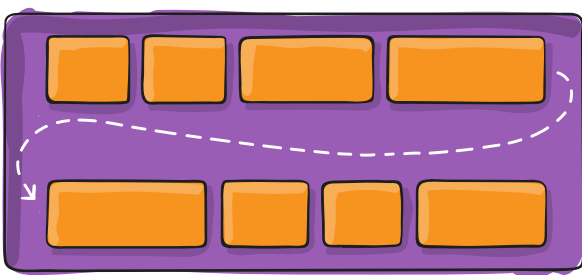
## flex-direction



- row (default)
- row-reverse
- column
- column-reverse

```
.flex {  
  display: flex;  
  flex-direction: row-reverse;  
}
```

## flex-wrap



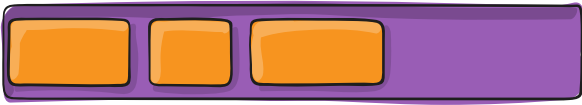
- nowrap (default)
- wrap
- wrap-reverse

Disminuir pantalla para visualizar cambios:

```
.flex {  
  display: flex;  
  flex-wrap: wrap;  
}
```

## justify-content

flex-start



flex-end



center



space-between



space-around



space-evenly

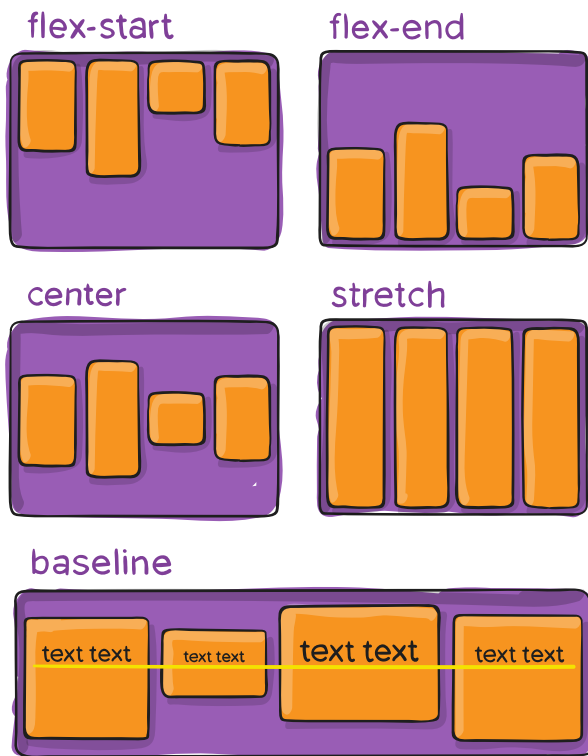


- flex-start (default)
- flex-end
- center
- space-between
- space-around
- space-evenly

```
.flex {  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: center;  
}
```

## align-items

Nos sirve para posicionar verticalmente los elementos si estos tienen alturas diferentes.



```
/* Colocar a un item */
.h-200 {
  height: 200px;
}
```

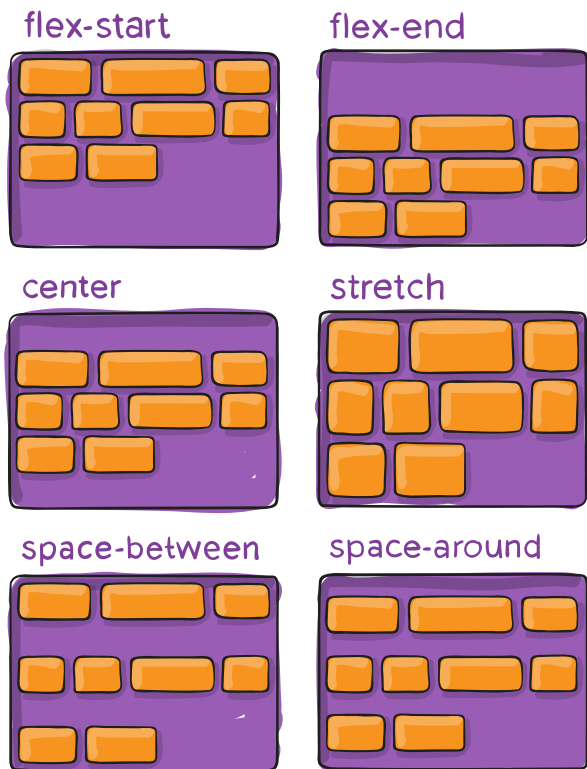
```
<div class="border flex">
  <div class="border item">item 1</div>
  <div class="border item h-200">item 2</div>
  <div class="border item">item 3</div>
</div>
```

- stretch (default)
- flex-start
- flex-end
- center
- baseline

```
.flex-container {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
}
```

## align-content

No confundir con `align-items` ya que ahora alineamos **todo el contenedor verticalmente** (independiente si existen más filas).



- flex-start
- flex-end
- space-between
- space-around
- space-evenly

```
<div class="border flex h-700">
  <div class="border item">item 1</div>
  <div class="border item h-200">item 2</div>
  <div class="border item">item 3</div>
</div>
```

```
.h-700 {
  height: 700px;
}

.flex {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  /* align-items: flex-start; */
}
```



```
    align-content: space-between;
}
```

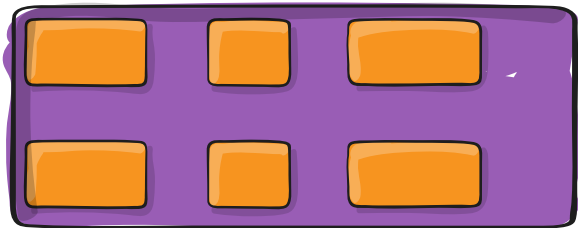
## gap

- controla explícitamente el espacio entre elementos flexibles.

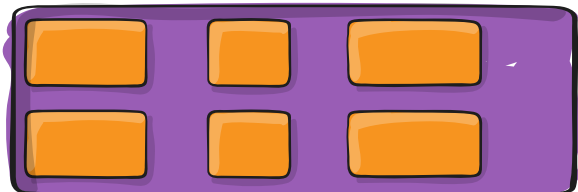
gap: 10px



gap: 30px



gap: 10px 30px



```
.item {
  background-color: tomato;
  text-align: center;
  width: 200px;
  padding: 10px;
  /* margin: 10px; */
}

.flex {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  /* align-items: flex-start; */
  align-content: space-around;
  row-gap: 10px;
  column-gap: 30px;
  /* gap: 10px 30px; row-gap column gap */
}
```

## Práctica

- [font-awesome CDN](#)

- icons

```
<!DOCTYPE html>
<html lang="es">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Position</title>

    <!-- Roboto Fonts -->
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link
      href="https://fonts.googleapis.com/css2?family=Roboto:wght@100;300;500;700&display=swap"
      rel="stylesheet"
    />

    <!-- CDN font-awesome -->
    <link
      rel="stylesheet"
      href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.1.1/css/all.min.css"
    />

    <!-- Normalize CSS -->
    <link rel="stylesheet" href="assets/css/normalize.css" />

    <!-- Style CSS -->
    <link rel="stylesheet" href="assets/css/style.css" />
  </head>
  <body>
    <h1 class="text-center">Características</h1>
    <section class="flex">
      <article class="card text-center">
        <i class="fa-brands fa-facebook fw-3"></i>
        <p>
          Lorem ipsum dolor, sit amet consectetur adipisicing elit.
          Corporis quo amet deleniti saepe, incidunt, magnam inventore
          impedit, quidem nemo cum delectus! Laboriosam pariatur
          reprehenderit doloribus soluta neque, dolorum quas harum.
        </p>
      </article>
      <article class="card text-center">
        <i class="fa-brands fa-facebook fw-3"></i>
        <p>
          Lorem ipsum dolor, sit amet consectetur adipisicing elit.
          Corporis quo amet deleniti saepe, incidunt, magnam inventore
          impedit, quidem nemo cum delectus! Laboriosam pariatur
          reprehenderit doloribus soluta neque, dolorum quas harum.
        </p>
      </article>
    </section>
  </body>
</html>
```

```

<article class="card text-center">
  <i class="fa-brands fa-facebook fw-3"></i>
  <p>
    Lorem ipsum dolor, sit amet consectetur adipisicing elit.
    Corporis quo amet deleniti saepe, incidunt, magnam inventore
    impedit, quidem nemo cum delectus! Laboriosam pariatur
    reprehenderit doloribus soluta neque, dolorum quas harum.
  </p>
</article>
</section>
</body>
</html>

```

```

html {
  box-sizing: border-box;
}
*,
*:before,
*:after {
  box-sizing: inherit;
}

body {
  font-family: "Roboto", sans-serif;
}

.text-center {
  text-align: center;
}

.fw-3 {
  font-size: 3rem;
}

.flex {
  display: flex;
  gap: 1rem;
}

```

## CSS Grid

- [complete guide grid](#)

```

<div class="grid">
  <article>
    <p>
      Lorem, ipsum dolor sit amet consectetur adipisicing elit.
      Laudantium distinctio tempora exercitationem itaque laborum.
      Eveniet ullam ut suscipit deleniti repellat perspiciatis
    </p>
  </article>
</div>

```

```
        nulla sint, culpa nemo praesentium aspernatur blanditiis
        quas nihil.
    </p>
</article>
<article>
    <p>
        Lorem, ipsum dolor sit amet consectetur adipisicing elit.
        Laudantium distinctio tempora exercitationem itaque laborum.
        Eveniet ullam ut suscipit deleniti repellat perspiciatis
        nulla sint, culpa nemo praesentium aspernatur blanditiis
        quas nihil.
    </p>
</article>
<article>
    <p>
        Lorem, ipsum dolor sit amet consectetur adipisicing elit.
        Laudantium distinctio tempora exercitationem itaque laborum.
        Eveniet ullam ut suscipit deleniti repellat perspiciatis
        nulla sint, culpa nemo praesentium aspernatur blanditiis
        quas nihil.
    </p>
</article>
<article>
    <p>
        Lorem, ipsum dolor sit amet consectetur adipisicing elit.
        Laudantium distinctio tempora exercitationem itaque laborum.
        Eveniet ullam ut suscipit deleniti repellat perspiciatis
        nulla sint, culpa nemo praesentium aspernatur blanditiis
        quas nihil.
    </p>
</article>
</div>
```

```
.grid {
    display: grid;
    grid-template-columns: 1fr 1fr 1fr;
    /* grid-template-columns: repeat(3, 1fr); */
    /* grid-template-columns: repeat(auto-fit, minmax(15rem, 1fr)); */
    /* grid-template-columns: repeat(auto-fit, minmax(min(100%, 20rem), 1fr)); */
    gap: 1rem;
}
```