

Flexbox part1

Edición Web I

Flexbox

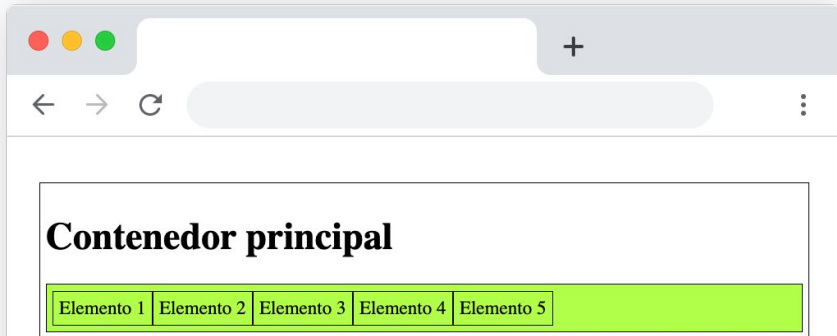
Display:flex;

Sistema de maquetación especialmente apropiado para hacer estructuras de una sola dimensión (filas o columnas, en lugar de una rejilla de elementos).

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

Display: flex;

Flex ocupará todo el ancho del padre.



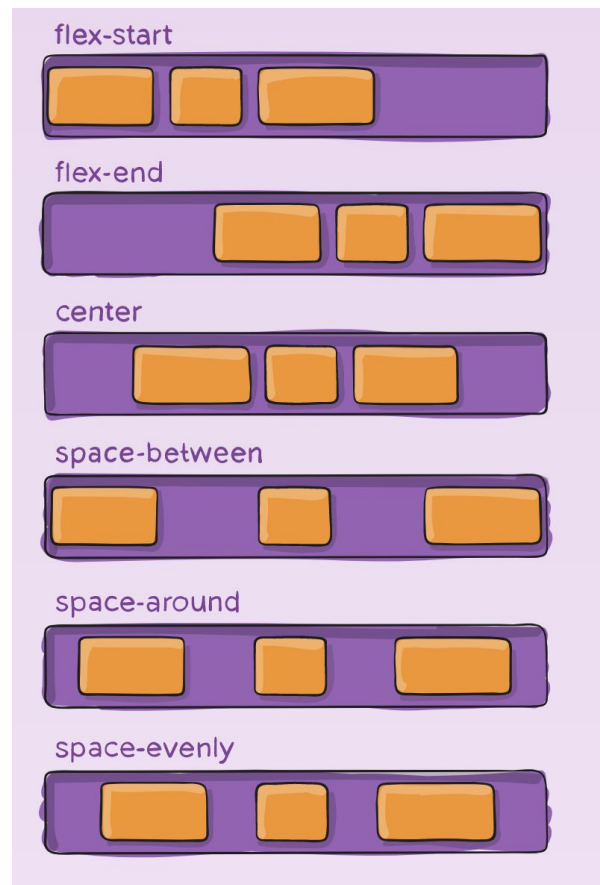
Con **display:flex;** conseguimos que cualquier elemento incluyendo los contenedores **<div>** los podamos poner **en línea** uno al lado del otro.

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Document</title>
5      </head>
6      <style>
7          div {
8              border: 1px solid black;
9              padding: 5px;
10         }
11         .probando-flex {
12             background-color: greenyellow;
13             display: flex;
14         }
15     </style>
16     <body>
17         <div class="container">
18             <h1>Contenedor principal</h1>
19             <div class="probando-flex">
20                 <div>Elemento 1</div>
21                 <div>Elemento 2</div>
22                 <div>Elemento 3</div>
23             </div>
24         </div>
25     </body>
26 </html>
```

justify-content

Alineación de los elementos horizontalmente.

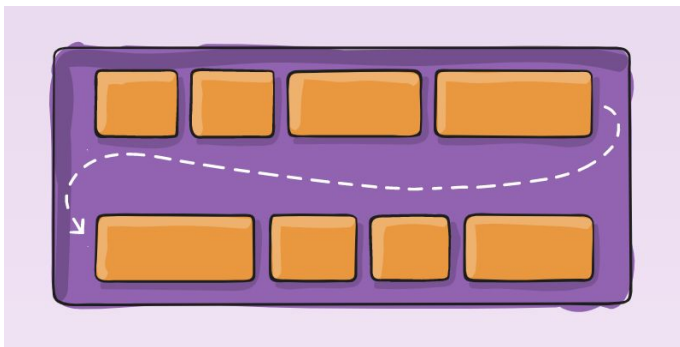
```
.contenedor{  
  display: flex;  
  justify-content: space-around;  
}
```



flex-wrap

wrap: Los elementos de la capa se distribuirán en una o varias filas en función de su tamaño.

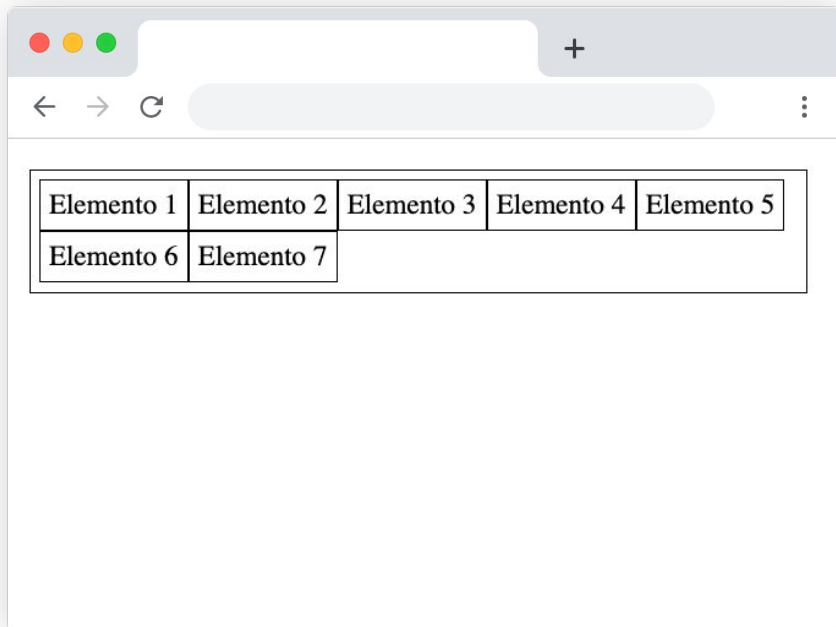
nowrap: Los elementos de la capa estarán sí o sí en una sola fila.



```
.container-wrap {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
}  
  
.container-no-wrap {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: nowrap;  
}
```

flex-wrap

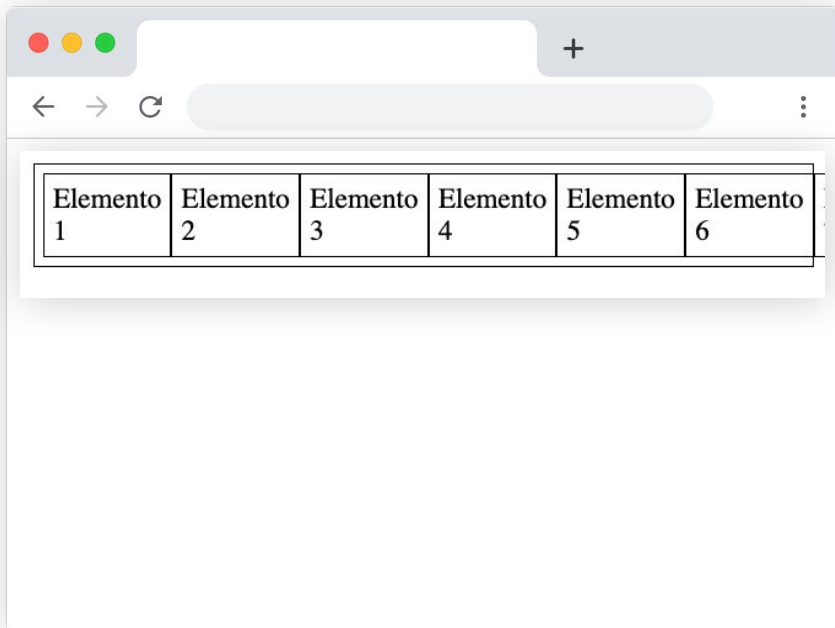
Flex-wrap:wrap;



```
6      <style>
7          div {
8              border: 1px solid black;
9              padding: 5px;
10         }
11         .container-wrap {
12             display: flex;
13             flex-direction: row;
14             flex-wrap: wrap;
15         }
16     </style>
17     <body>
18         <div class="container-wrap">
19             <div>Elemento 1</div>
20             <div>Elemento 2</div>
21             <div>Elemento 3</div>
22             <div>Elemento 4</div>
23             <div>Elemento 5</div>
24             <div>Elemento 6</div>
25             <div>Elemento 7</div>
26         </div>
27     </body>
```

flex-wrap

Flex-wrap: nowrap;

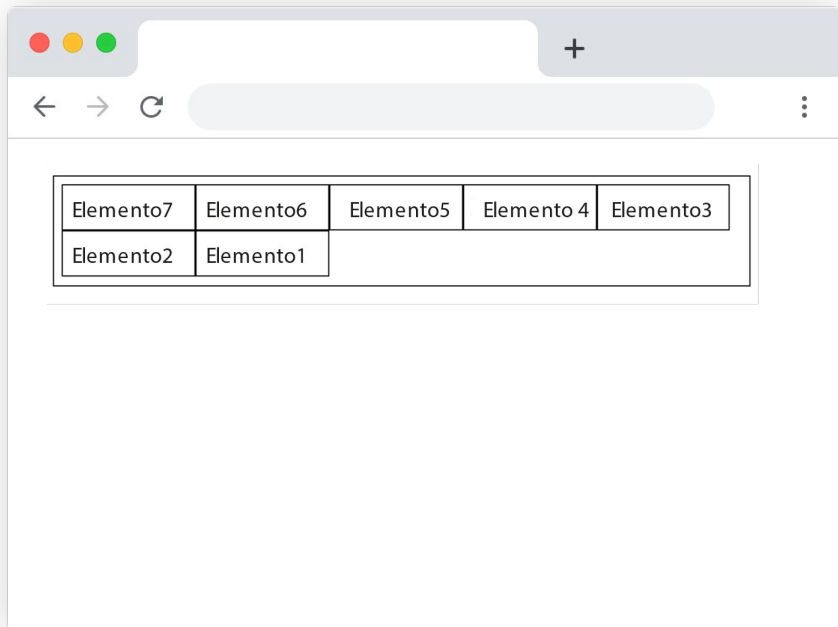


```
6  <style>
7    div {
8      border: 1px solid black;
9      padding: 5px;
10   }
11   .container-no-wrap {
12     display: flex;
13     flex-direction: row;
14     flex-wrap: nowrap;
15   }
16 </style>
17 <body>
18   <div class="container-no-wrap">
19     <div>Elemento 1</div>
20     <div>Elemento 2</div>
21     <div>Elemento 3</div>
22     <div>Elemento 4</div>
23     <div>Elemento 5</div>
24     <div>Elemento 6</div>
25     <div>Elemento 7</div>
26   </div>
27 </body>
```

flex-wrap

Flex-wrap: wrap-reverse;

Invierte el orden que mostramos los elementos.



```
6 <style>
7   div {
8     border: 1px solid black;
9     padding: 5px;
10  }
11  .container-no-wrap {
12    display: flex;
13    flex-direction: row;
14    flex-wrap: nowrap;
15  }
16 </style>
17 <body>
18   <div class="container-no-wrap">
19     <div>Elemento 1</div>
20     <div>Elemento 2</div>
21     <div>Elemento 3</div>
22     <div>Elemento 4</div>
23     <div>Elemento 5</div>
24     <div>Elemento 6</div>
25     <div>Elemento 7</div>
26   </div>
27 </body>
```

gap

Gap:8px;

Espacio entre elementos.

*Si **utilizamos justify-content:flex-start**; todos los elementos estarían pegados, con **gap** creamos un espacio de 8px entre cada elemento.*



```
6  <style>
7    div {
8      border: 1px solid black;
9      padding: 5px;
10   }
11   .container-no-wrap {
12     display: flex;
13     flex-direction: row;
14     flex-wrap: nowrap;
15   }
16 </style>
17 <body>
18   <div class="container-no-wrap">
19     <div>Elemento 1</div>
20     <div>Elemento 2</div>
21     <div>Elemento 3</div>
22     <div>Elemento 4</div>
23     <div>Elemento 5</div>
24     <div>Elemento 6</div>
25     <div>Elemento 7</div>
26   </div>
27 </body>
```

Grow

Grow es una forma de **asignar tamaño horizontal** de forma flexible, por lo que asignaremos la propiedad **flex-grow** al **elemento hijo** y no al contenedor.

También se puede utilizar solo **flex:1**;

```
.contenedor{  
    display:flex;  
    justify-content: space-between;  
    flex-wrap: wrap;  
}  
  
.item{flex-grow: 1;}
```

Grow

Flex-grow:1;

Factor de crecimiento.

Es un valor proporcional al ancho de pantalla, sería como si lo definieramos por fracciones:

Sin factor de crecimiento



Factor de crecimeinto solo al primer elemento



Mismo factor de crecimiento



Diferente factor de crecimiento



[Ir al ejemplo](#)

Ejemplo 1

Centrar 3 contenedores
hijo con una imagen cada
uno de ellos:



```
.contenedor-padre{  
  display:flex;  
  justify-content:center;  
  border:1px solid black;  
  padding:8px;  
}
```

```
.contenedor{  
  width:800px;  
  display:flex;  
  justify-content:center;  
  gap:8px;  
  padding:8px;  
  background-color: gray;  
}
```

```
<div class="contenedor-padre">  
  <div class="contenedor">  
    <div></div>  
    <div></div>  
    <div></div>  
  </div>  
</div>
```

Ejemplo 2: barra navegación



```
<nav>

  <div>
    
  </div>

  <div class="botones">
    <a href="" class="link">Boton 1</a>
    <a href="" class="link">Boton 2</a>
    <a href="" class="link">Boton 3</a>
  </div>

  <div>
    <a href=""></a>
    <a href=""></a>
    <a href=""></a>
  </div>

</nav>
```

Ejemplo 2: barra navegación



```
nav{  
  display:flex;  
  justify-content:space-between;  
  Background-color:gray;  
  padding:16px;  
  margin:6px;  
  border-radius:8px;  
}
```

```
.logo{  
  border-radius:500px;  
  width:30px;  
}
```

```
.botones{  
  background-color:rgba(255,255,255,0.4);  
  padding:16px;  
  border-radius:8px;  
  display:flex;  
}
```

```
.link{  
  font-size:18px;  
  color:white;  
  text-decoration:none;  
  border:2px solid white;  
  border-radius:24px;  
  padding: 6px 16px;  
}
```

[Ir al ejemplo](#)

Resumen propiedades flex

`display: flex;` Permite colocar los elementos en el eje horizontal

`justify-content` Diferentes alineaciones de los elementos

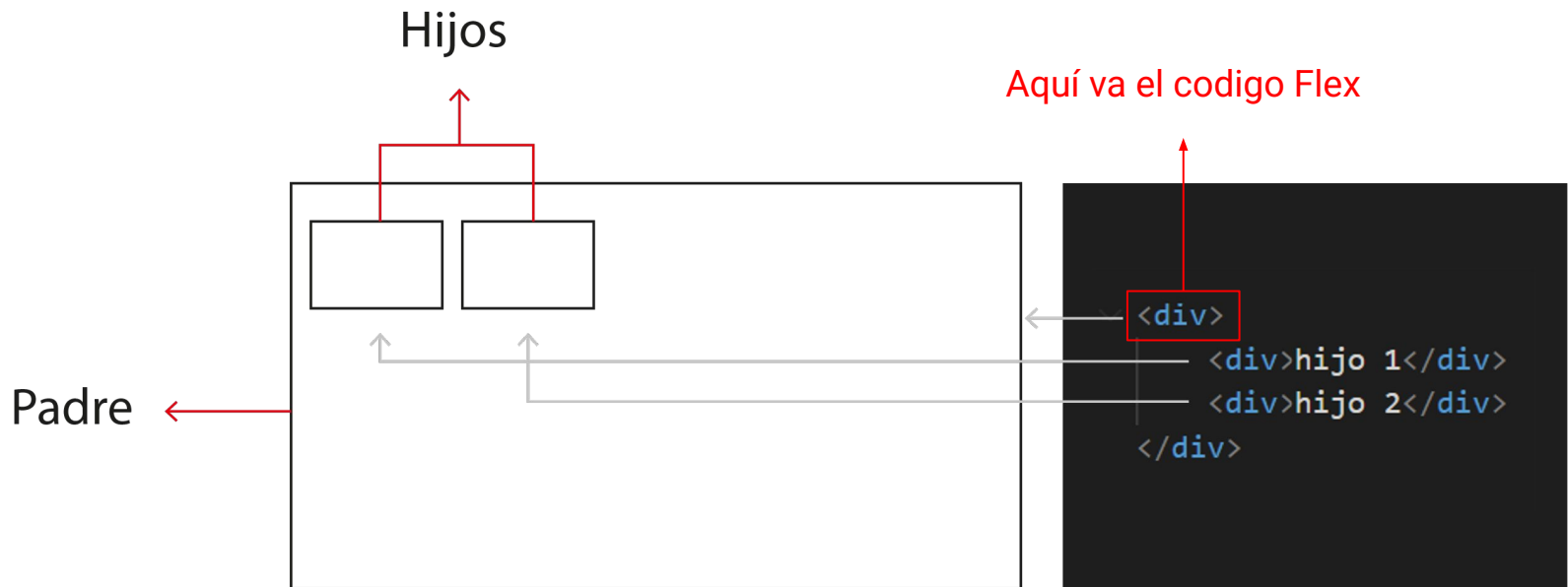
`flex-wrap` Los elementos se sitúan en una o varias líneas

`gap` Espacio entre los elementos

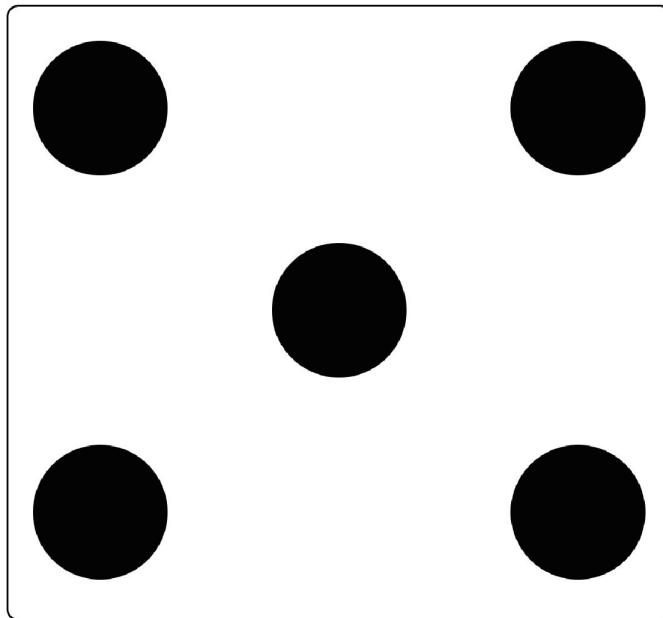
`flex-grow` Asignación anchura flexible o relativa

Recordatorio Importante

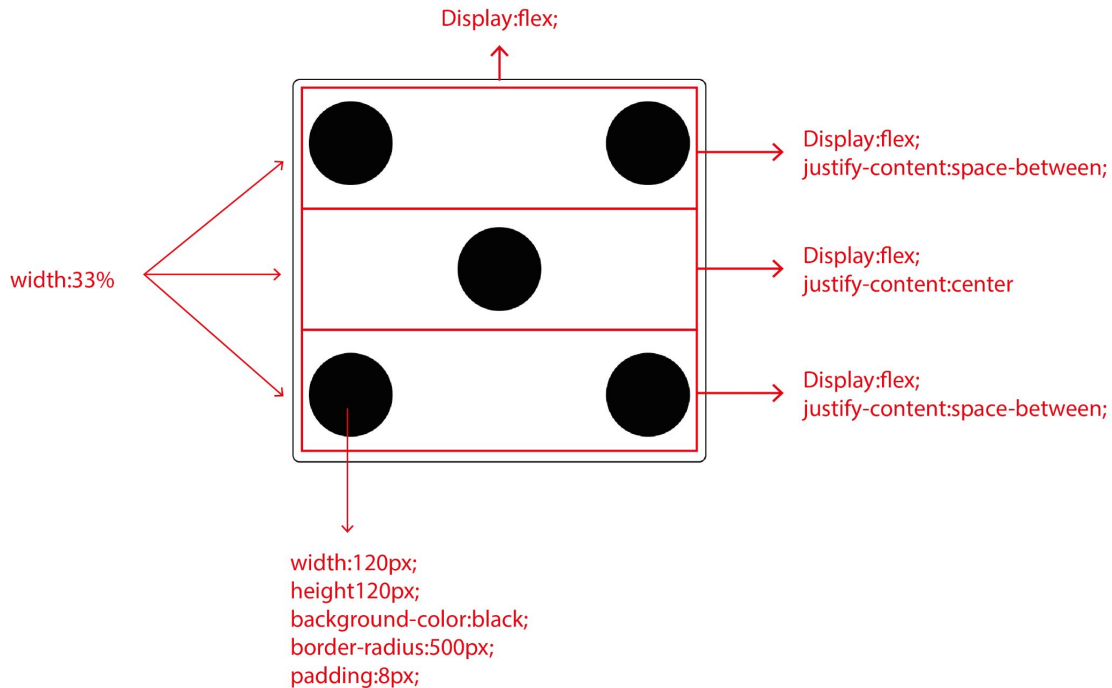
Las propiedades flex se aplica al contenedor padre... NO a los elementos que queremos alinear.



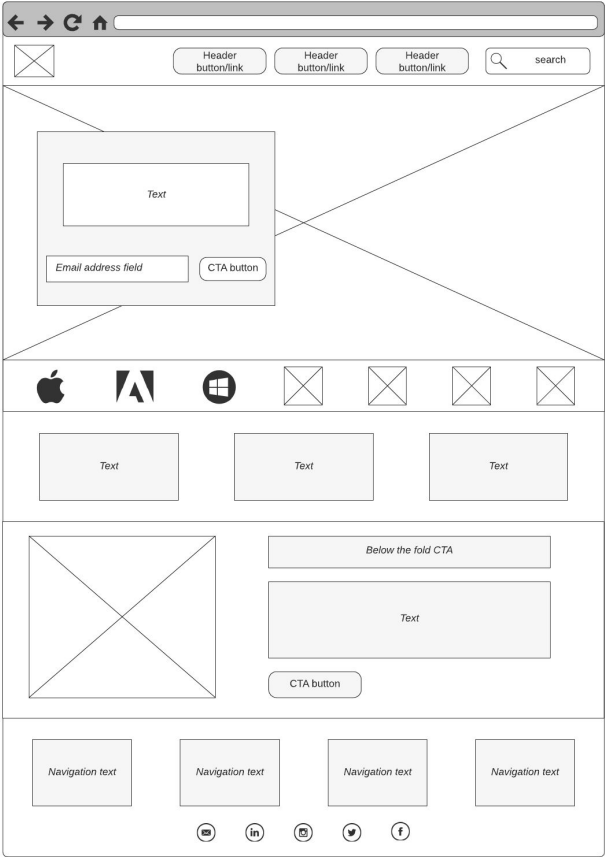
Ejemplo 3: dado en css



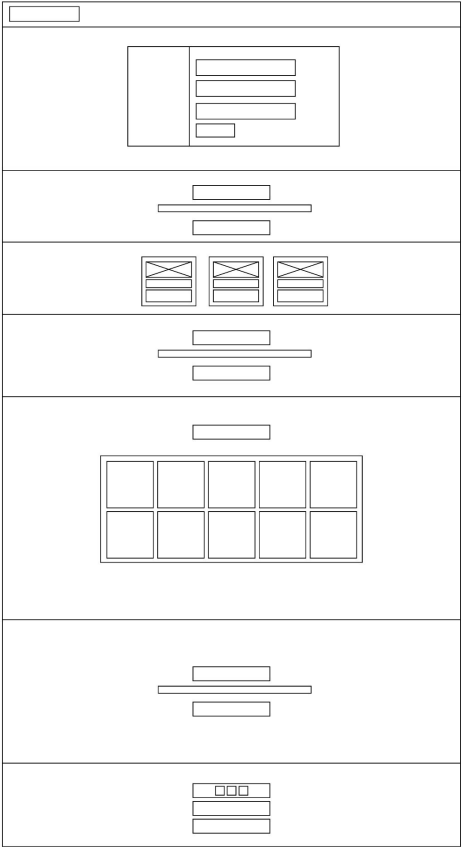
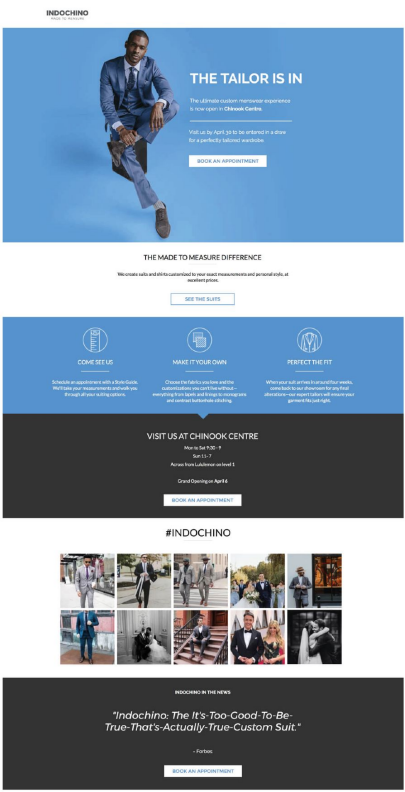
Ejemplo 3: dado en css



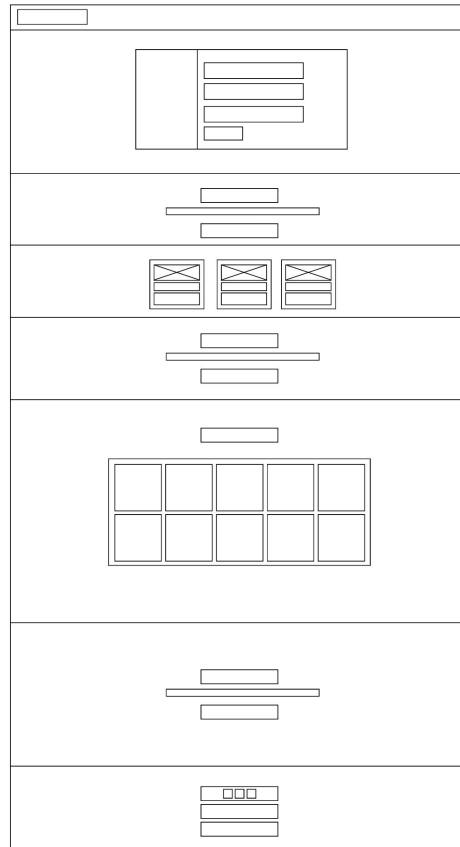
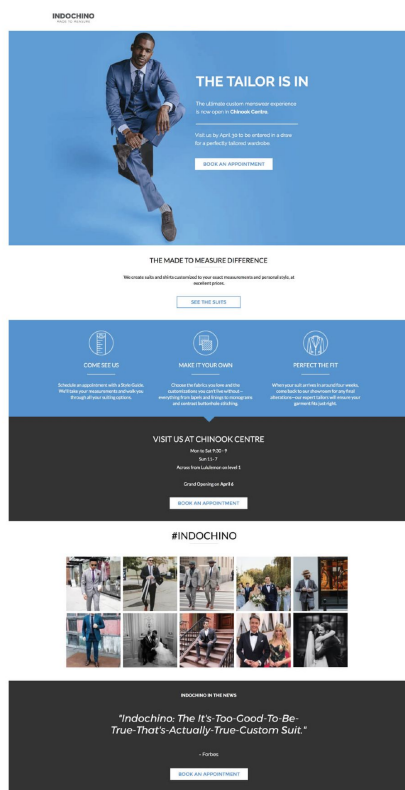
[Ir al ejemplo](#)



Ejemplo wireframe



Ejemplo wireframe



→ Display:flex;
justify-content:center

→ Display:flex;
justify-content:center

→ flex-wrap:wrap;

Ejercicio 1

Ordenando cajas

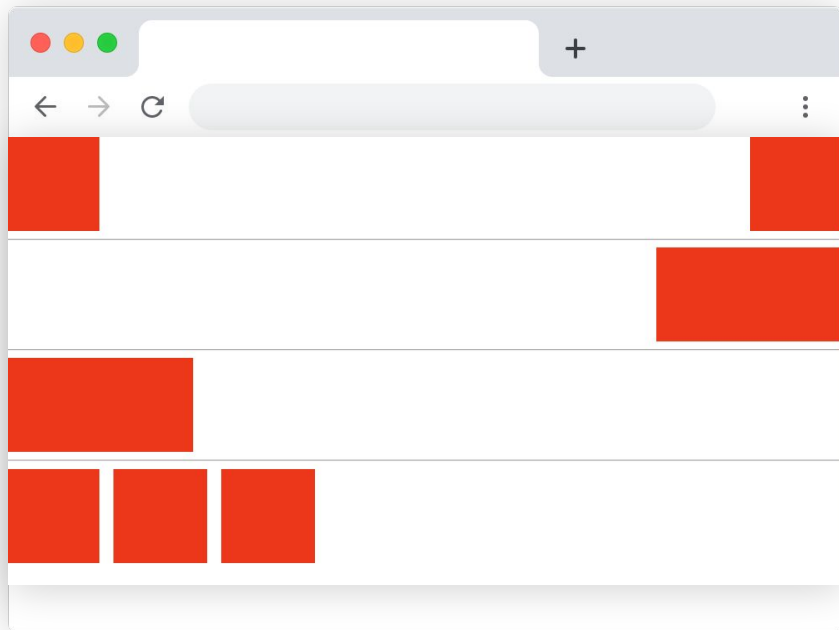
Tenemos que probar las propiedades de **justify-content**.

Para ello crearemos una clase `.box` que será 100 de alto y 100 de ancho.

Dentro de un padre meteremos tantas cajas como vemos en la imagen.

Entre cada padre añadiremos un `hr`.

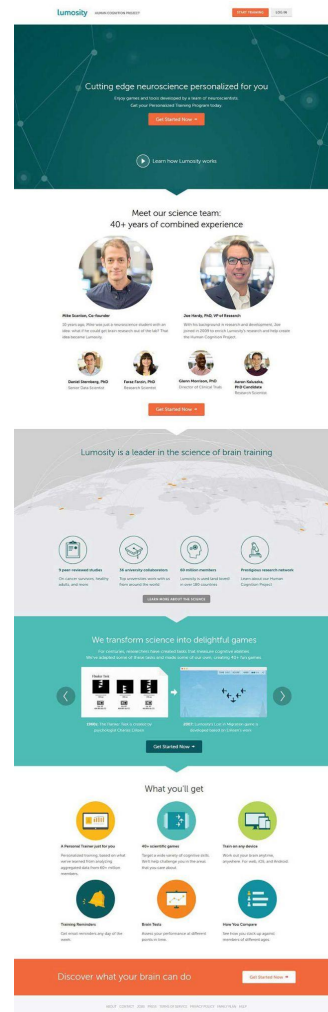
La carpeta se llamará "ordenando_cajas".



Ejercicio 2

Crear un wireframe

Crea la estructura de la página “wireframe”
puedes utilizar un programa de diseño o a mano.



Ejercicio 3

Crear un wireframe

La carpeta se llamará “portfolio”.

Buscaremos 3 o 4 imágenes de la temática que queramos (preferiblemente trabajos vuestros) de un tamaño de 800x420.

Mis fotos
preferidas



Montañas
rocosas

Pequeña descripción del paisaje. Vemos unas montañas con un fondo de nubes.



Árboles
rosas

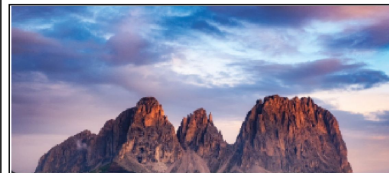
Pequeña descripción del paisaje. Vemos un camino con unos árboles de flores rosas.



Cascadas

Pequeña descripción del paisaje. En la imagen vemos muchas cascadas con mucha vegetación.

Ayuda:



Montañas
rocosas

Pequeña descripción del paisaje. Vemos unas montañas con un fondo de nubes.

ABOUT
ME

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut

Think.
Create.
Inspire.

Esta línea es un área de inspiración. Esta línea es un área de inspiración. Esta línea es un área de inspiración.

Escríbame
emilipamela.com

Lláname
+34 000 00 00

En redes
Instagram
Facebook

Página de referencia

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>