

# Flexbox

Ángel Villalba Fdez-Paniagua

# Table of Contents

1. Introducción .....	1
2. Atributos de los padres .....	1
2.1. <b>display</b> .....	1
2.2. flex-wrap .....	2
2.3. justify-content .....	3
2.4. align-items .....	4
3. Atributos de los hijos .....	5
3.1. order .....	5

# 1. Introducción

**Flexbox** es un nuevo diseño de layout que trae CSS3 para mejorar la forma en que colocamos los elementos en la página. También permite hacer **responsive design** de una forma más sencilla.

Una de las principales ventajas que trae consigo es que se reduce el uso de la propiedad **float**.

Lo que se consigue con flexbox es que todos los elementos hijos de un contenedor que usa flexbox, van a tener unas dimensiones flexibles y se van a ajustar al espacio que tienen.

## 2. Atributos de los padres

Los atributos que contienen los elementos padre de los elementos que se van a ir colocando dentro del contenedor son los siguientes.

### 2.1. display

Esta propiedad define el contenedor que va a usar **flexbox** y para usarlo hay que darle el valor **flex**.

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="./styles.css" >
  <title>Ejemplo - Flexbox</title>
</head>
<body>
  <div id="container">
    <div class="hijo">Hijo 1</div>
    <div class="hijo">Hijo 2</div>
    <div class="hijo">Hijo 3</div>
  </div>
</body>
</html>
```

```
#container {
  height: 500px;
  width: 80%;
  background-color: orange;
  margin: 50px auto;
  display: flex;
}

.hijo {
  background-color: black;
  width: 200px;
  height: 100px;
  color: white;
  margin: 30px;
  padding: 20px;
  text-align: center;
}
```

## 2.2. flex-wrap

Por defecto, los items de *flex* van a intentar ocupar todo el espacio del contenedor padre.

Con esta propiedad vamos a decirle que si los elementos no entran en la misma línea, que salte a otra línea los elementos.

Los valores que puede tomar son:

- no-wrap: no saltan
- wrap: saltan de arriba hacia abajo
- wrap-reverse: saltan de abajo hacia arriba

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="./styles.css" >
  <title>Ejemplo - Flexbox</title>
</head>
<body>
  <div id="container">
    <div class="hijo">Hijo 1</div>
    <div class="hijo">Hijo 2</div>
    <div class="hijo">Hijo 3</div>
    <div class="hijo">Hijo 4</div>
    <div class="hijo">Hijo 5</div>
  </div>
</body>
</html>
```

```
#container {  
  height: 500px;  
  width: 80%;  
  background-color: orange;  
  margin: 50px auto;  
  display: flex;  
  flex-wrap: wrap;  
}  
  
.hijo {  
  background-color: black;  
  width: 200px;  
  height: 100px;  
  color: white;  
  margin: 30px;  
  padding: 20px;  
  text-align: center;  
}
```

## 2.3. justify-content

Esta propiedad define como se tienen que alinear los elementos hijos a lo largo de la línea en la que se encuentran.

Los posibles valores son:

- flex-start: se alinean al principio
- flex-end: se alinean al final
- center: se alinean al centro
- space-between: se alinean dejando espacio entre ellos
- space-around: se alinean dejando espacio entre ellos y los bordes del contenedor padre

```
#container {  
  height: 500px;  
  width: 80%;  
  background-color: orange;  
  margin: 50px auto;  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: center;  
}  
  
.hijo {  
  background-color: black;  
  width: 200px;  
  height: 100px;  
  color: white;  
  margin: 30px;  
  padding: 20px;  
  text-align: center;  
}
```

## 2.4. align-items

Esta propiedad permite alinear los elementos de la misma línea verticalmente (algo que no se podía hacer hasta ahora con CSS).

Los valores que puede tomar son:

- flex-start: se alinean al principio
- flex-end: se alinean al final
- center: se alinean al centro verticalmente
- stretch: se alinean y hace que todos los elementos tengan la misma altura que el elemento más alto

```
#container {
  height: 500px;
  width: 80%;
  background-color: orange;
  margin: 50px auto;
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
  align-items: center;
}

.hijo {
  background-color: black;
  width: 200px;
  height: 100px;
  color: white;
  margin: 30px;
  padding: 20px;
  text-align: center;
}
```

## 3. Atributos de los hijos

Los elementos hijos también tienen una serie de propiedades que pueden usar.

### 3.1. order

Esta propiedad indica el orden en que se tienen que mostrar los elementos hijos.

La propiedad obtiene como valor un número, siendo el número más bajo el elemento que aparecerá primero.

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="./styles.css" />
  </head>
  <body>
    <div id="container">
      <div class="hijo">Hijo 1</div>
      <div class="hijo">Hijo 2</div>
      <div class="hijo hijo-prioritario">Hijo 3</div>
      <div class="hijo">Hijo 4</div>
      <div class="hijo hijo-prioritario">Hijo 5</div>
    </div>
  </body>
</html>
```

```
#container {  
  height: 500px;  
  width: 80%;  
  background-color: orange;  
  margin: 50px auto;  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: center;  
  align-items: center;  
}  
  
.hijo {  
  background-color: black;  
  width: 200px;  
  height: 100px;  
  color: white;  
  margin: 30px;  
  padding: 20px;  
  text-align: center;  
  order: 2;  
}  
  
.hijo-prioritario {  
  order: 1;  
}
```