

Bootcamp de Desarrollo Web

Clase 6

Flexbox



¿Qué es flexbox?

Es un sistema de elementos flexibles. Donde los elementos HTML se adaptan y colocan automáticamente siendo mucho más fácil personalizar los diseños.

Está pensado para diseños en una sola dimensión, es decir, o filas o columnas.

El Flexbox hace que se más fácil crear estructuras de layout responsive, sin usar posicionamiento o flotantes.





Contenedor y elementos flexibles

Para utilizar Flexbox, se debe convertir un contenedor en un contenedor flexible (**display: flex**).

Los elementos hijos directos de este contenedor se convierten en elementos flexibles (**flex items**) y se pueden manejar con propiedades específicas de Flexbox.

```
section {  
  display: flex;  
}
```



Flex Direction

Hay 4 valores para la propiedad **flex-direction**, que son:

- **row:** Indica que los elementos flexibles deben mostrarse en una fila de izquierda a derecha en el contexto del eje principal.
- **row-reverse:** Indica que los elementos flexibles deben mostrarse en una fila de derecha a izquierda en el contexto del eje principal.
- **column:** Indica que los elementos flexibles deben mostrarse en una columna de arriba hacia abajo en el contexto del eje principal.
- **column-reverse:** Indica que los elementos flexibles deben mostrarse en una columna de abajo hacia arriba en el contexto del eje principal.



Flex Direction

Flex Directions

flex-direction: row

1	2	3	
---	---	---	--

flex-direction: column

1
2
3

flex-direction: row-reverse

	3	2	1
--	---	---	---

flex-direction: column-reverse

3
2
1

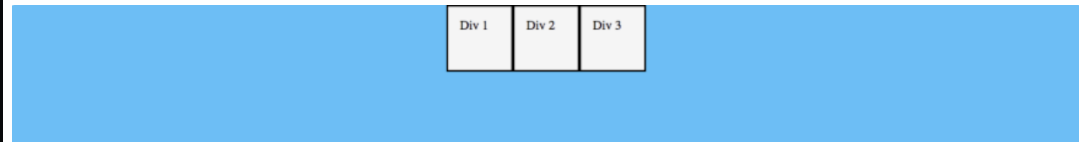


justify-content

La propiedad **justify-content** se utiliza para alinear horizontalmente los elementos dentro del contenedor **flexbox**.

justify-content: center hace exactamente lo que piensas: centra todos los elementos dentro del contenedor flexbox.

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: center; /* Add this line */  
}
```





justify-content: flex-start

flex-start alinea los elementos hacia la izquierda del contenedor flexbox.

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: flex-start; /* Change this line */  
}
```





justify-content: flex-end

flex-end es lo mismo que **flex-start**, excepto que alinea los elementos hacia el borde derecho (el 'final' del contenedor flexbox).

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: flex-end; /* Change this line */  
}
```

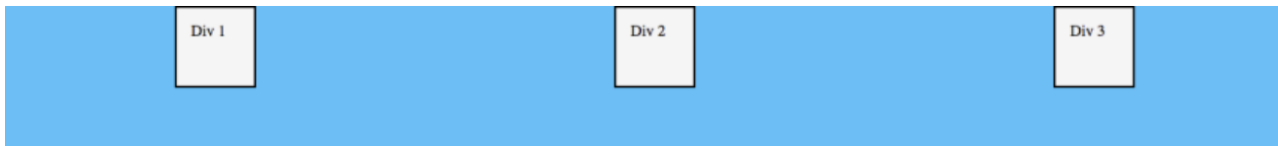




justify-content: space-around

Hace espacio en los elementos dentro del contenedor **flexbox** para que cada elemento tenga la misma cantidad de espacio alrededor.

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: space-around; /* Change this line */  
}
```

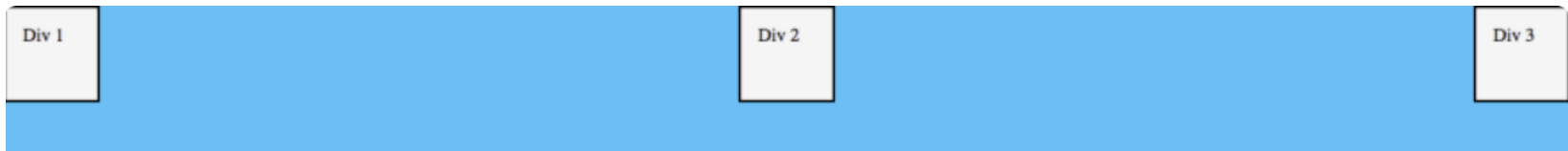




justify-content: space-between

Separa los elementos con una cantidad uniforme de espacio entre ellos en lugar de con una cantidad uniforme de espacio alrededor de ellos.

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: space-between; /* Change this line */  
}
```

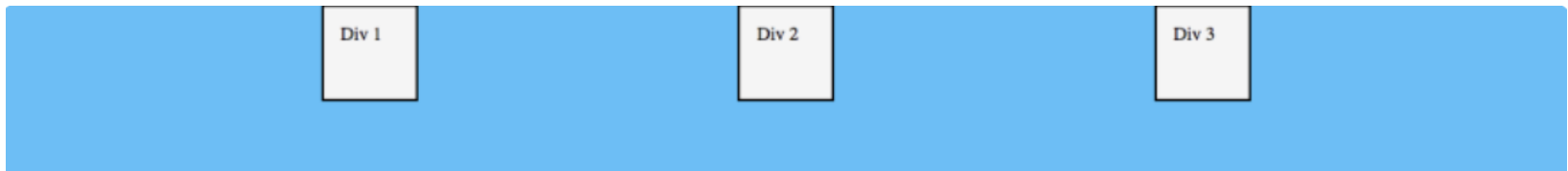




justify-content: space-evenly

El espacio entre cada elemento es el mismo, pero también es igual al espacio entre los elementos exteriores y los bordes

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: space-evenly; /* Change this line */  
}
```



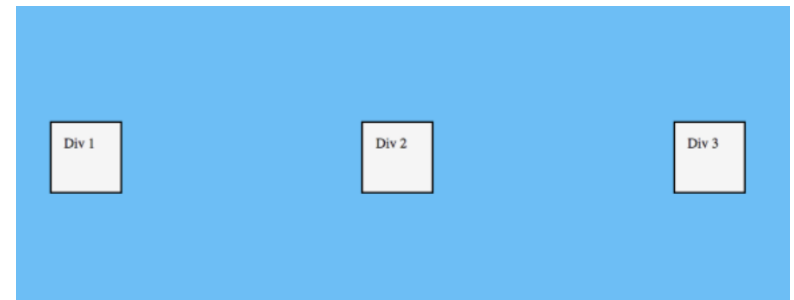


Propiedad align-items

Controla la alineación vertical de los elementos flexibles dentro del contenedor flexbox a lo largo del eje transversal (ya sea fila o columna, dependiendo de la dirección principal establecida por flex-direction).

Align Center:

```
#wrapper {  
  background-color: deepskyblue;  
  height: 50vh;  
  display: flex;  
  justify-content: space-evenly;  
  align-items: center; /* Add this line */  
}
```



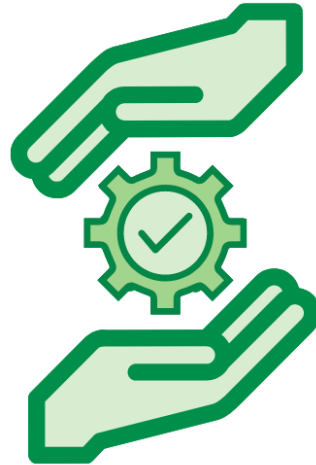


align-items:flex-start y align-items:flex-end

align-items: flex-start alinea los elementos flexibles al principio del contenedor flexbox a lo largo del eje transversal, ya sea en la parte superior si la dirección principal es una fila o a la izquierda si la dirección principal es una columna.

align-items: flex-end posiciona los elementos al final del contenedor flexbox a lo largo del eje transversal, es decir, en la parte inferior si la dirección principal es una fila o a la derecha si la dirección principal es una columna.





Bootcamp de Desarrollo Web

Clase 6

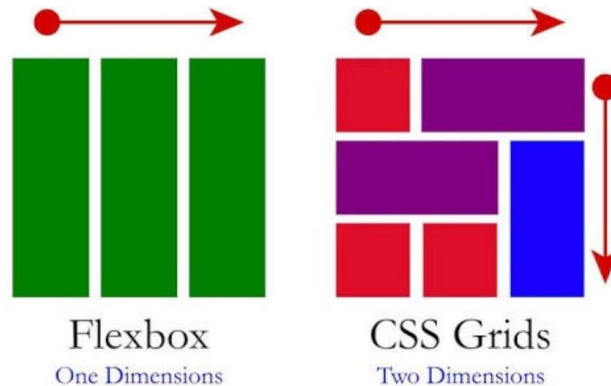
CSS Grid



¿Qué es CSS Grid?

CSS Grid es un sistema de diseño bidimensional en CSS que permite crear diseños de manera más estructurada y flexible que los modelos de diseño anteriores.

Se compone de un conjunto de líneas horizontales y verticales que definen filas y columnas en las que se colocan los elementos HTML.





Propiedad display: grid | inline-grid

Para crear la cuadrícula grid hay que definir sobre el elemento contenedor la propiedad display y especificar el valor grid o inline-grid.

```
<!DOCTYPE html>
<html>
<head>
<style>
.contenedor{
  display: grid;
}
</style>
</head>
<body>
<div class="contenedor"> <!-- contenedor padre-->
  <div class="rejilla">Item 1</div> <!-- Ítems del grid -->
  <div class="rejilla">Item 2</div>
  <div class="rejilla">Item 3</div>
  <div class="rejilla">Item 4</div>
</div>
</body>
</html>
```




Propiedad `display`: `grid` | `inline-grid`

Los valores **`inline-grid`** y **`grid`** indican cómo se comporta el contenedor con respecto al contenido exterior.

Con el valor **`inline-grid`** el contenedor aparece en línea con respecto al contenido exterior.

Con el valor **`grid`**, el contenedor aparece en bloque con respecto al contenido exterior.

`display: inline-grid;`

Contenido exterior Item 1
Item 2
Item 3
Item 4

`display: grid;`

Contenido exterior
Item 1
Item 2
Item 3



Propiedades `grid-template-columns` y `grid-template-rows`

Para definir el espaciado o tamaño de las columnas y las filas usamos las propiedades **`grid-template-columns`** y **`grid-template-rows`**.

```
.contenedor{  
  display: grid;  
  grid-template-rows: 200px; 200px  
  grid-template-columns: 200px; 200px; 200px;  
}
```

`grid-template-columns: auto auto auto auto;`

Item 1 Item 2 Item 3 Item 4

`grid-template-columns: 50px 200px 50px 500px;`

Item 1 Item 2 Item 3 Item 4

`grid-template-columns: 1fr 2fr 1fr 3fr;`

Item 1 Item 2 Item 3 Item 4

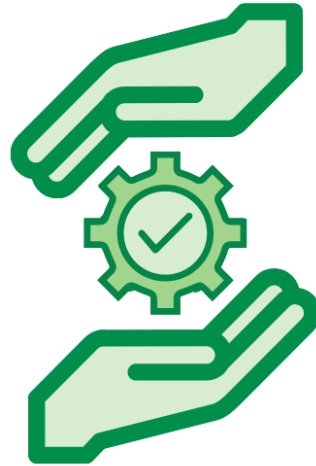


Propiedades row-gap y column-gap

Podemos definir el espaciado entre las rejillas mediante las propiedades **row-gap** y **column-gap**. El espaciado solo se crea entre las columnas/filas, no en los bordes exteriores.

```
.contenedor{  
  display: grid;  
  grid-template-rows: 200px; 200px  
  grid-template-columns: 200px; 200px; 200px;  
  column-gap: 20px;  
  row-gap: 10px;  
}
```





Bootcamp de Desarrollo Web

Clase 6

CSS