



## Contenedor Flex (display: flex)

Primero tienes que declarar un contenedor como flex:

CSS

CopiarEditar

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

---



## Dirección de los ítems: flex-direction

Controla cómo se alinean los hijos:

CSS

CopiarEditar

```
.contenedor {  
  flex-direction: row;    /* Horizontal (por defecto) */  
  flex-direction: row-reverse; /* Horizontal invertido */  
  flex-direction: column; /* Vertical */  
  flex-direction: column-reverse; /* Vertical invertido */  
}
```

---



## Ajuste de línea: flex-wrap

¿Los elementos se ajustan o no cuando no caben?

CSS

CopiarEditar

```
.contenedor {  
  flex-wrap: nowrap; /* Todos en una línea (por defecto) */  
  flex-wrap: wrap; /* Se ajustan a la siguiente línea si no caben */  
  flex-wrap: wrap-reverse; /* Igual que wrap pero en orden inverso */  
}
```



## Dirección + ajuste: flex-flow

Atajo para flex-direction y flex-wrap:

CSS

CopiarEditar

```
.contenedor {  
  flex-flow: row wrap;  
}
```



## Alineación en el eje principal: justify-content

Alinea horizontalmente si row, o verticalmente si column:

CSS

CopiarEditar

```
.contenedor {  
  justify-content: flex-start; /* Izquierda (por defecto) */  
  justify-content: flex-end; /* Derecha */  
  justify-content: center; /* Centrado */  
  justify-content: space-between; /* Espaciado entre */  
  justify-content: space-around; /* Espaciado con márgenes */  
  justify-content: space-evenly; /* Espacio igual entre todos */  
}
```



## Alineación en el eje cruzado: align-items

Alinea verticalmente si row, u horizontalmente si column:

CSS

CopiarEditar

```
.contenedor {  
  align-items: stretch; /* Estira para que llenen el alto (por defecto) */
```

```
align-items: flex-start; /* Arriba */
align-items: flex-end; /* Abajo */
align-items: center; /* Centrado */
align-items: baseline; /* Alinea por texto/base */
}
```

---



## Alineación múltiple: align-content

Solo aplica si hay varias filas/columnas (wrap activado):

CSS

CopiarEditar

```
.contenedor {
  align-content: flex-start;
  align-content: flex-end;
  align-content: center;
  align-content: space-between;
  align-content: space-around;
  align-content: stretch;
}
```

---



## Tamaño y crecimiento individual: flex (en los hijos)

Esto se pone en los elementos hijos del contenedor flex:

CSS

CopiarEditar

```
.item {
  flex: 1; /* Crece para ocupar el espacio disponible equitativamente */
}
```

Forma completa:

CSS

CopiarEditar

```
.item {  
  flex: grow shrink basis;  
}
```

/\* Ejemplo práctico \*/

```
.item {  
  flex: 1 1 200px; /* crece, encoge, tamaño base */  
}
```

---



## Alineación individual: align-self

Permite que un hijo tenga una alineación diferente al resto:

CSS

CopiarEditar

```
.item {  
  align-self: center;  
  align-self: flex-end;  
  align-self: flex-start;  
  align-self: stretch;  
}
```

---

### ✂ Ejemplo completo

HTML

CopiarEditar

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

CSS

CopiarEditar

```
.contenedor {  
  display: flex;  
  flex-direction: row;  
  flex-wrap: wrap;  
  justify-content: space-around;  
  align-items: center;  
  height: 200px;  
}
```

```
.item {  
  flex: 1 1 100px;  
  background: lightblue;  
  margin: 10px;  
  text-align: center;  
}
```

## ¿Cómo funciona?

### 1. Creas un contenedor Flex:

Ese será el que controla cómo se organizan sus hijos.

css

CopiarEditar

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

### 2. Los hijos del contenedor se convierten en ítems flex:

Estos son los que **se alinean automáticamente**, se ajustan, se distribuyen, crecen o encogen.

html

CopiarEditar

```
<div class="contenedor">

  <div class="item">1</div>

  <div class="item">2</div>

  <div class="item">3</div>

</div>
```

---

## ¿Qué puedes controlar desde el contenedor?

- Dirección de los ítems (flex-direction)
  - Si se ajustan o no (flex-wrap)
  - Alineación horizontal (justify-content)
  - Alineación vertical (align-items)
  - Alineación de múltiples líneas (align-content)
- 

### ✂ Y desde los hijos (.item) puedes controlar:

- Cuánto crecen o encogen (flex)
  - Alineación individual (align-self)
  - Tamaño base (flex-basis)
  - Orden de aparición (order)
- 

### Mini resumen visual:

plaintext

CopiarEditar

```
<div class="contenedor">  ← Contenedor (display: flex)

  └─ <div class="item">1</div>  ← Hijo (ítem flex)

  └─ <div class="item">2</div>

  └─ <div class="item">3</div>

</div>
```

## Ejemplo HTML + CSS:

html

CopiarEditar

```
<!DOCTYPE html>

<html lang="es">

<head>

  <meta charset="UTF-8">

  <title>Ejemplo Flexbox</title>

  <style>

    body {

      font-family: sans-serif;

      background-color: #f0f0f0;

      padding: 30px;

    }

    .contenedor {

      display: flex;          /* Activamos Flexbox */

      flex-direction: row;    /* Dirección horizontal */

      flex-wrap: wrap;        /* Que se ajusten si no caben */

      justify-content: space-around; /* Espacio alrededor de los hijos */

      align-items: center;     /* Centrado vertical */

      background-color: #333;

      padding: 20px;

      height: 300px;

      border-radius: 10px;

    }

    .item {

      background-color: #4CAF50;

      color: white;

      padding: 20px;
```

```

margin: 10px;

text-align: center;

flex: 1 1 150px; /* Que crezcan, se encojan, y tamaño base */

border-radius: 8px;

font-size: 1.2em;
}
</style>
</head>
<body>

<h1>Ejemplo de Flexbox en acción 🍌</h1>

<div class="contenedor">

  <div class="item">Elemento 1</div>

  <div class="item">Elemento 2</div>

  <div class="item">Elemento 3</div>

  <div class="item">Elemento 4</div>

</div>

</body>
</html>

```

Para poner un navegador (<nav>) a la izquierda, el contenido principal (<main>) a su derecha, y un <aside> también dentro del mismo contenedor **usando Flexbox**, puedes hacerlo así:

## Estructura HTML

html

CopiarEditar

```

<div class="contenedor">

  <nav>Navegación</nav>

```



```
<main>Contenido principal</main>

<aside>Contenido secundario</aside>

</div>
```

## CSS con Flex

CSS

CopiarEditar

```
.contenedor {

  display: flex;

  flex-direction: row; /* Por defecto ya es row, pero lo dejamos claro */

}
```

```
nav {

  width: 200px;

  background-color: #ccc;

  padding: 10px;

}
```

```
main {

  flex: 1;

  background-color: #eee;

  padding: 10px;

}
```

```
aside {

  width: 200px;

  background-color: #ddd;

  padding: 10px;

}
```

 **Explicación rápida**

- display: flex en el contenedor principal hace que los hijos (nav, main, aside) se coloquen en fila.
- nav y aside tienen ancho fijo.
- main ocupa el espacio restante con flex: 1.