

MINI GUÍA DISPLAY FLEX

```
#contenedor{
```

```
display: flex; /* Esto hace toda la "magia":
                los "hijos" se distribuirán dentro de este container */
```

```
flex-flow: row ó column nowrap ó wrap;
```

```
/* Es un shorthand de flex-direction y flex-wrap */
```

```
/* flex-direction: puede ser row, column, row-reverse y column-reverse */
```

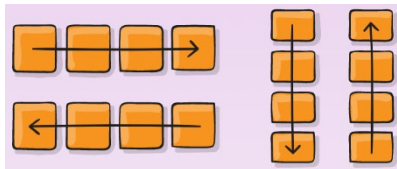
```
/* flex-wrap: si "hijos" son obligados a permanecer en una misma línea o
pueden fluir en varias líneas*/
```

```
/* flex-flow por defecto es row nowrap, pero conviene ponerlo */
```

```
justify-content: space-around|space-evenly|... ;
/* distribución en el eje primario */
```

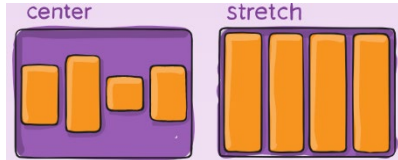
```
align-items: center|flex-start|flex-end|stretch;
/* alineación en eje secundario */
```

```
}
```

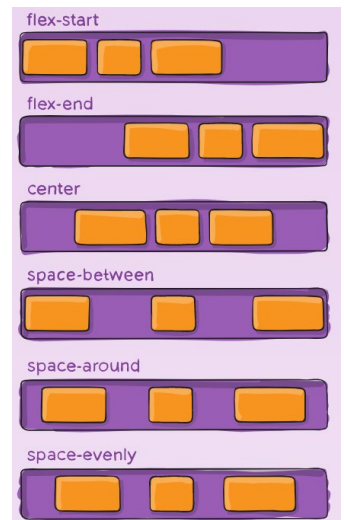


Si flex-flow es row, el eje primario es X,
y el secundario el Y.

Si flex-flow es column, el eje primario es Y,
y el secundario el X.



Por defecto, y si los hijos no tienen tamaño,
el align-items por defecto es stretch



```
#itemX{
```

```
flex: 2 0 20em (unidades o %); /* Shorthand de 3 propied.: grow, shrink, basis */
```

```
/* flex-grow: es la porción del tamaño disponible que coge este container (def. 0)*/
```

```
/* flex-shrink: rapidez con la que decrecerá container cuando minimicemos (def. 1)*/
```

```
/* flex-basis: es tamaño de container en eje primario (actúa como width o height) */
```

Casi siempre a flex le daremos un valor: si valor tiene unidad será basis, si no será grow

```
align-self: center|flex-start|flex-end|stretch;
```

```
/* si este ítem queremos que no obedezca a align-items */
```

```
order: 3; /* Podemos cambiar el orden del elemento en el flex */
```

```
}
```

