DESARROLLO WEB

LA WEB

¿Que se necesita para ser desarrollador web?

HTML5
Esqueleto/Piezas

CSS3
Piel/Ropa

JAVASCRIPT
Accesorios/Dinamismo visual

BACKEND
Órganos/Funcionalidad interna

:FLEXBOX:
GRID

Ing. Adrián Manzano


01
FLEXBOX

FLEXBOX

Flex es uno de los nuevos valores **CSS3** para la propiedad **display** que nos permite maquetar nuestras páginas web de una manera mucho más fácil de lo que se hacía con la forma tradicional

PROPIEDADES IMPORTANTES - PADRE

display

flex

flex-direction

row | column

justify-content

flex-start | flex-end
center | space-between
space-around | space-evenly

align-items

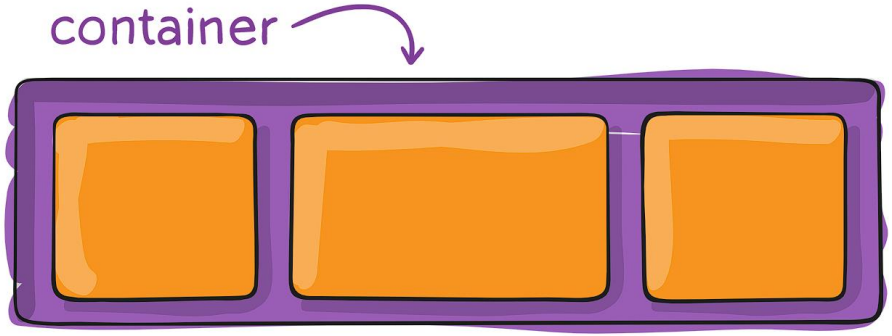
flex-start | flex-end
center | baseline
stretch

flex-wrap

nowrap | wrap
wrap-reverse

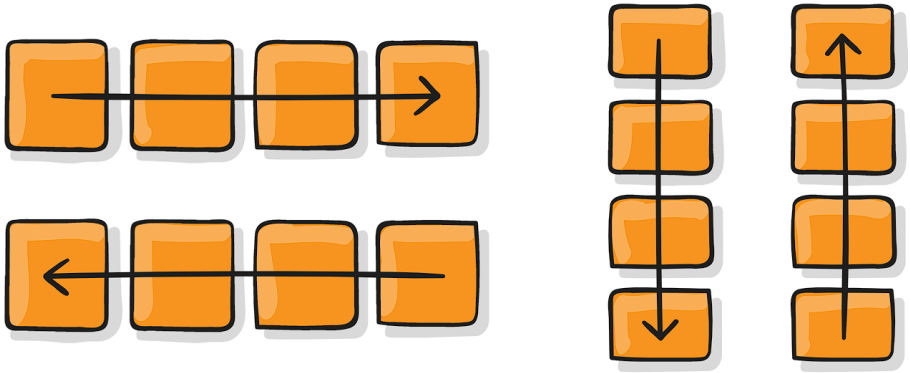
DISPLAY

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



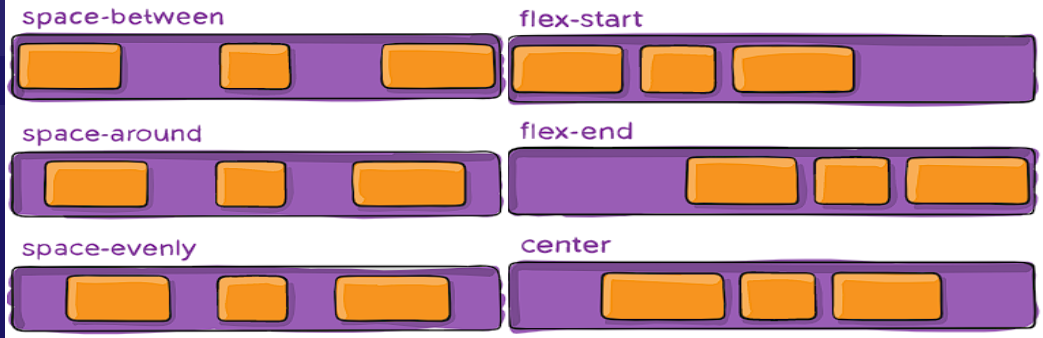
FLEX DIRECTION

```
.container {
  flex-direction:
        row         | row-reverse
        column      | column-reverse;
}
```



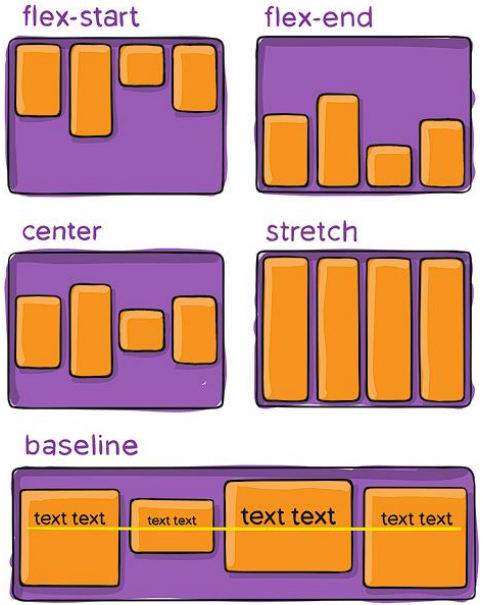
JUSTIFY CONTENT

```
.container {
  justify-content:
        flex-start      | flex-end
        space-between   | space-around
        center          | space-evenly;
}
```



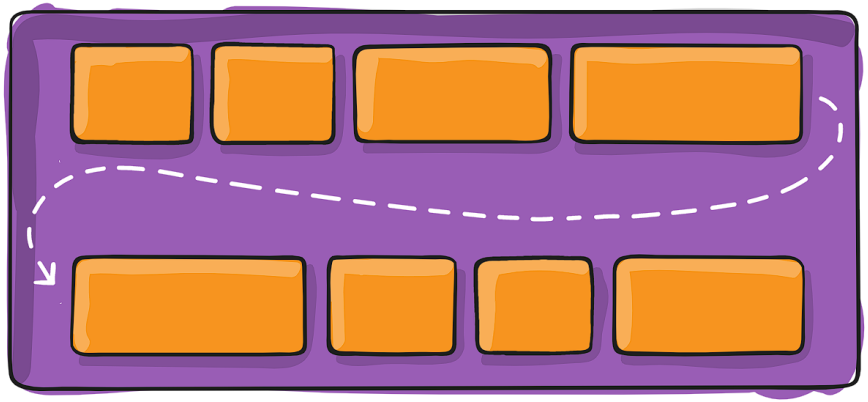
ALIGN ITEMS

```
.container {
  align-items:
        stretch | flex-start | flex-end
        center  | baseline;
}
```



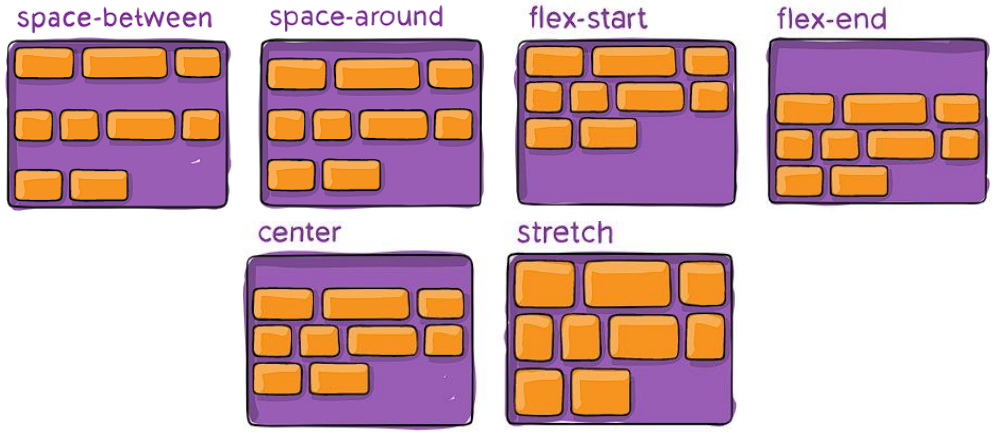
FLEX WRAP

```
.container {
  flex-wrap:
        nowrap | wrap | wrap-reverse;
}
```



ALIGN CONTENT

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



PROPIEDADES IMPORTANTES - HIJOS

order

<number>

flex

[<flex-grow><flex-shrink>||<flex-basis>]

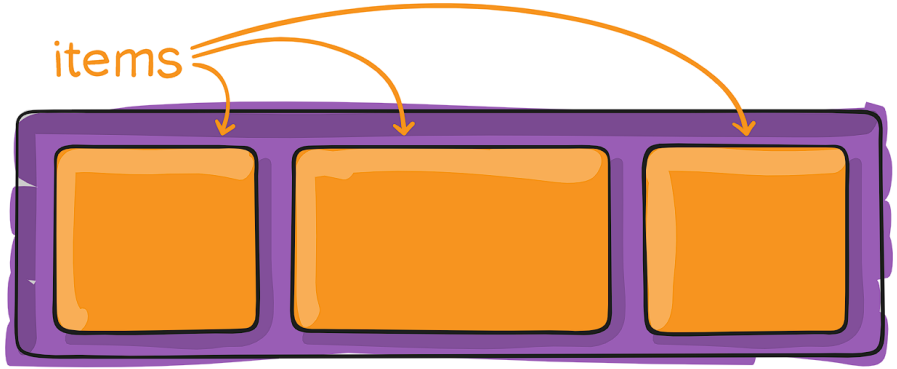
flex-grow: <number>;
flex-shrink: <number>;
flex-basis: <unit>;

align-self

auto | flex-start
Flex-end | center
Baseline | stretch

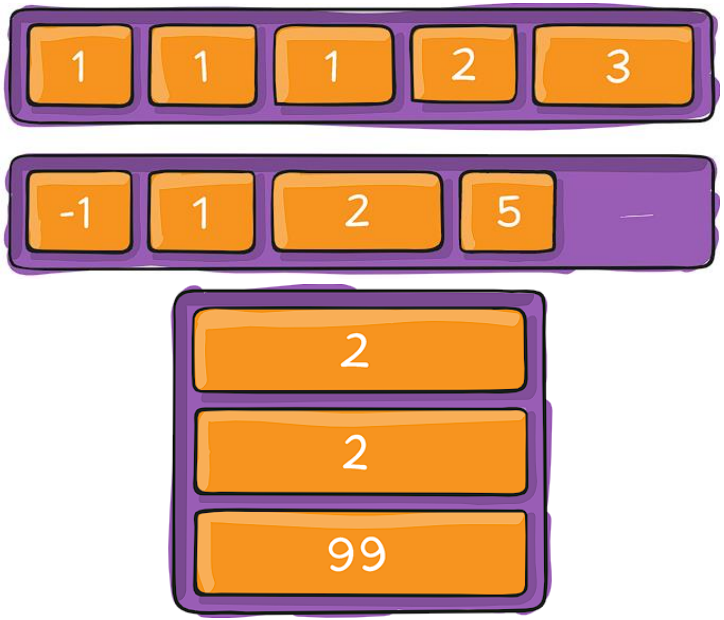
DISPLAY

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



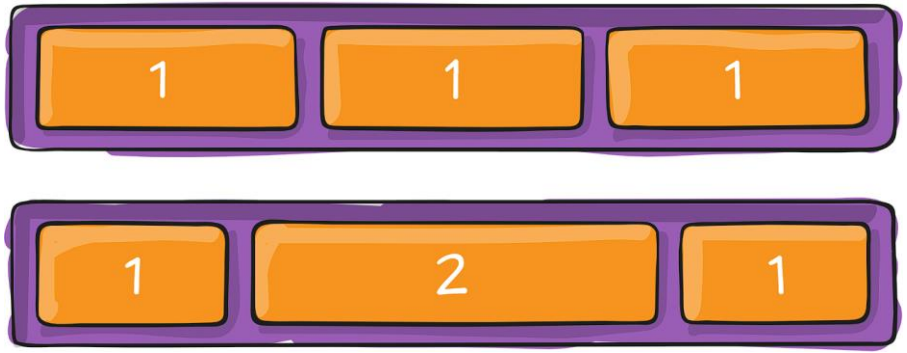
ORDER

```
.item {
  order: 5; /* default 0 */
}
```



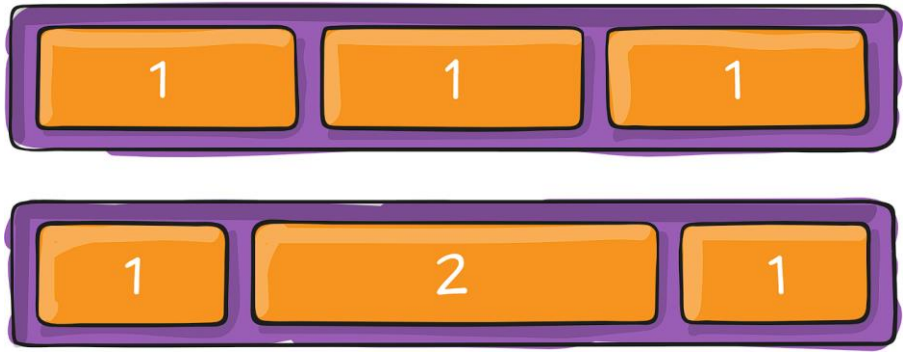
FLEX GROW

```
.item {
  flex-grow: 2; /* default 0 */
}
```



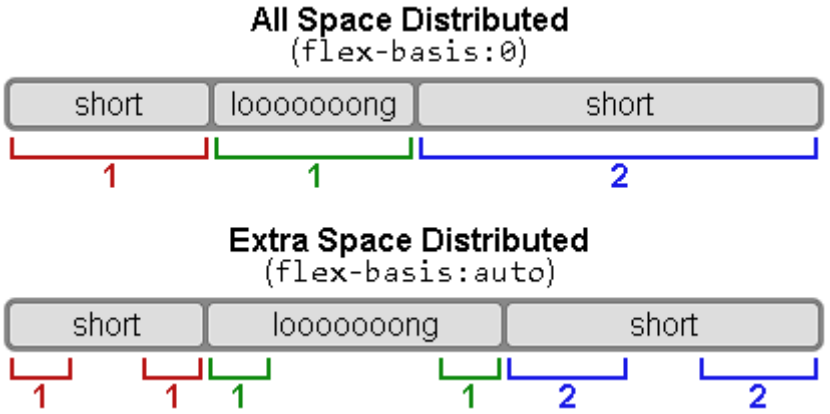
FLEX SHRINK

```
.item {
  flex-shrink: 3; /* default 1 */
}
```



FLEX BASIS

```
.item {
  flex-basis: auto; /* default auto */
}
```



FLEX

```
.item {
  flex:
        [ <'flex-grow'> <'flex-shrink'>?
          <'flex-basis'> ]
}
```


```
.tomate {
   flex: 2;
}
```

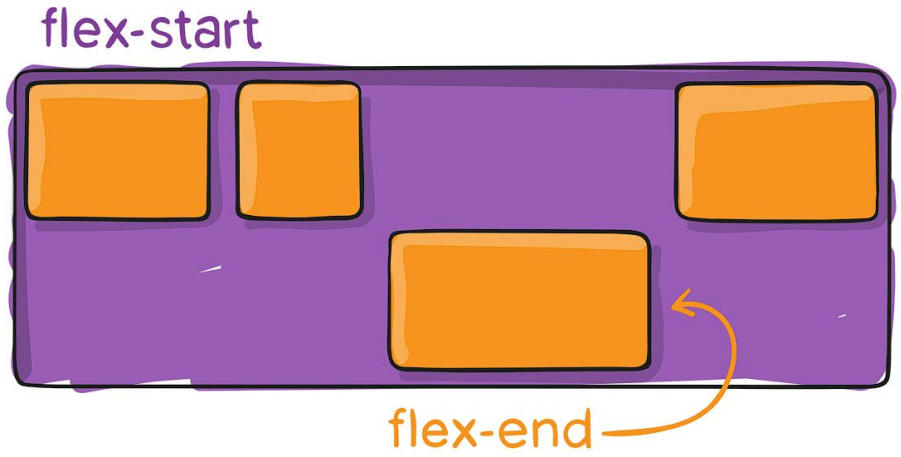
2/3

```
.oignon {
   flex: 1;
}
```

1/3

ALIGN SELF

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



FLEX

La propiedad flex es la abreviatura de tres propiedades **flex-grow**, **flex-shrink** y **flex-basis** cuyas características son:

- **flex-grow :** Capacidad de un elemento para estirarse en el espacio restante
- **flex-shrink :** Capacidad de un elemento para contraerse si es necesario,
- **flex-basis :** Tamaño inicial del elemento antes de distribuir el espacio restante.

Por defecto, los valores de estas propiedades son: **flex-grow: 0, flex-shrink: 1 y flex-basis: auto**.


02
GRID

GRID

CSS GRID es un mecanismo para maquetar en dos dimensiones, independiente del orden y si es o no flexible

PROPIEDADES IMPORTANTES - CONTAINER

display

grid

grid-template-rows

<track-size>

grid-template-columns

<track-size>

grid-template-areas

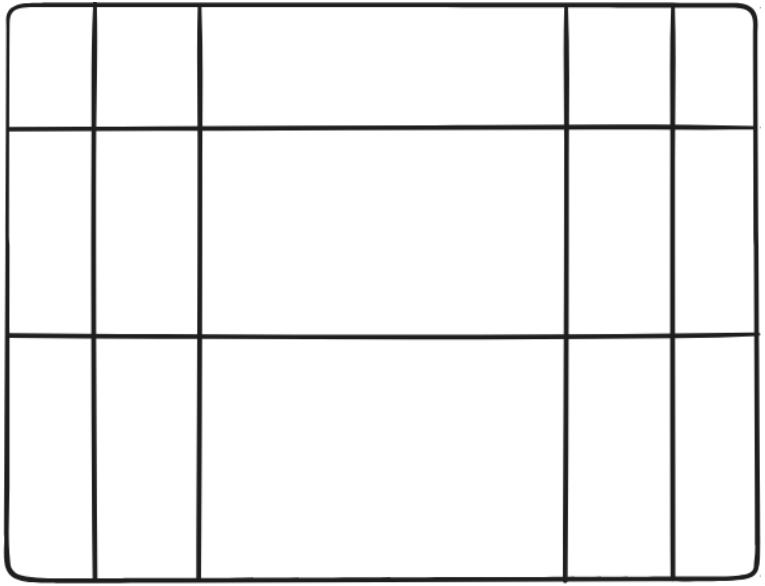
<grid-area-name>| ... | none |...

grid-gap

<size>

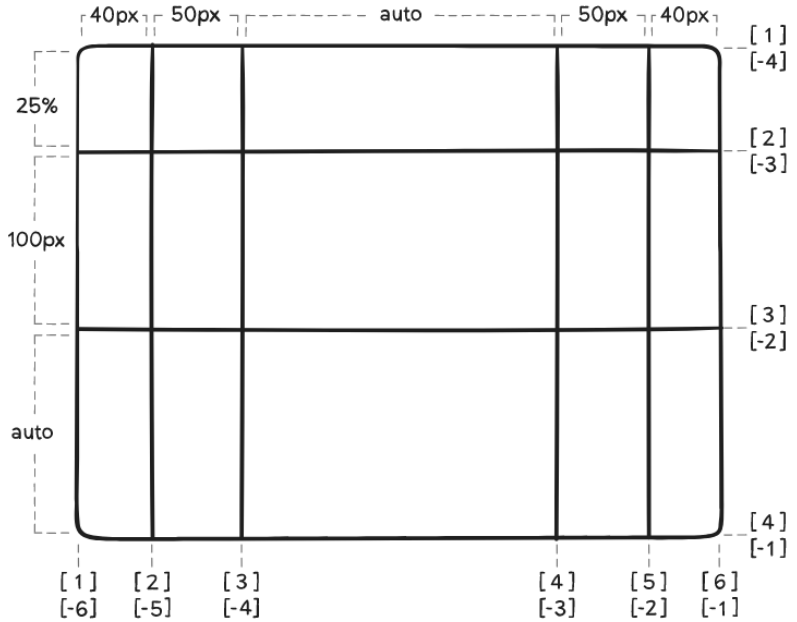
DISPLAY

```
.container {
  display: grid | inline-grid;
}
```



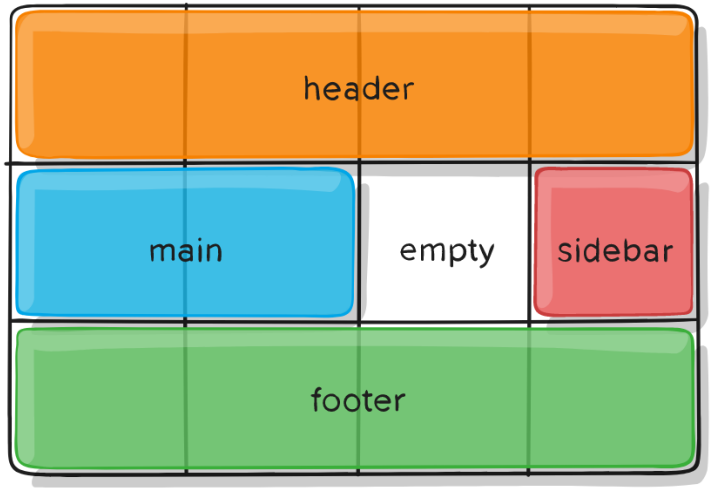
GRID TEMPLATE COLUMNS-ROWS

```
.container {
  grid-template-columns: 40px 50px auto 50px 40px;
  grid-template-rows: 25% 100px auto;
}
```



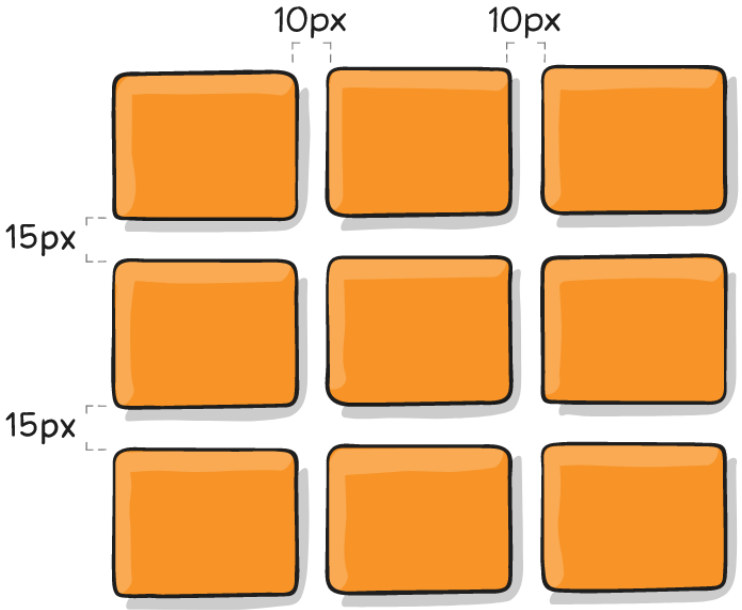
GRID TEMPLATE AREAS

```
.container {
  display: grid;
  grid-template-columns: 50px 50px 50px 50px;
  grid-template-rows: auto;
  grid-template-areas:
    "header header header header"
    "main main . sidebar"
    "footer footer footer footer";
}
```



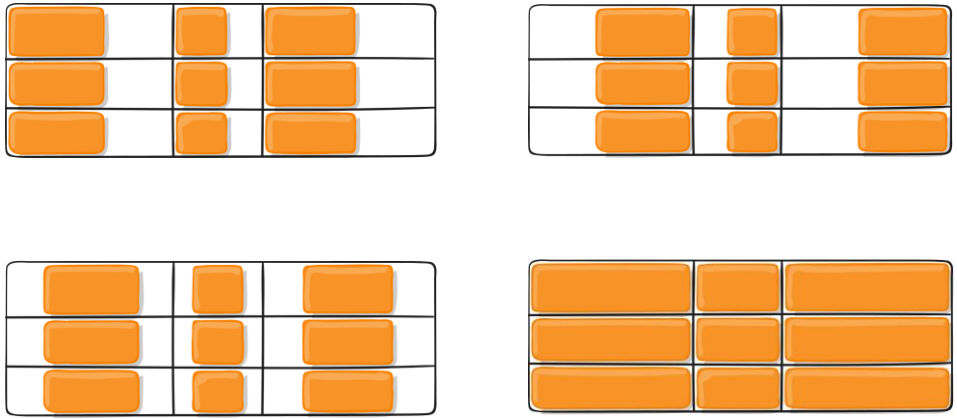
GRID GAP

```
.container {
  grid-template-columns: 100px 50px 100px;
  grid-template-rows: 80px auto 80px;
  gap: 15px 10px;
}
```



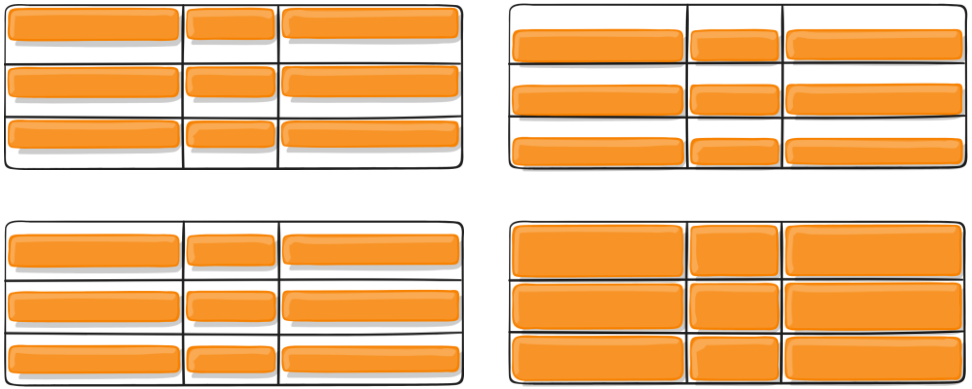
JUSTIFY ITEMS

```
.container {
  justify-items: start | end | center | stretch;
}
```



ALIGN ITEMS

```
.container {
  align-items: start | end | center | stretch;
}
```



PROPIEDADES IMPORTANTES - HIJOS

grid-column

<start-line> / <end-line>
<start-line> / span<value>

grid-row

<start-line> / <end-line>
<start-line> / span<value>

grid-area

<name>

justify-self

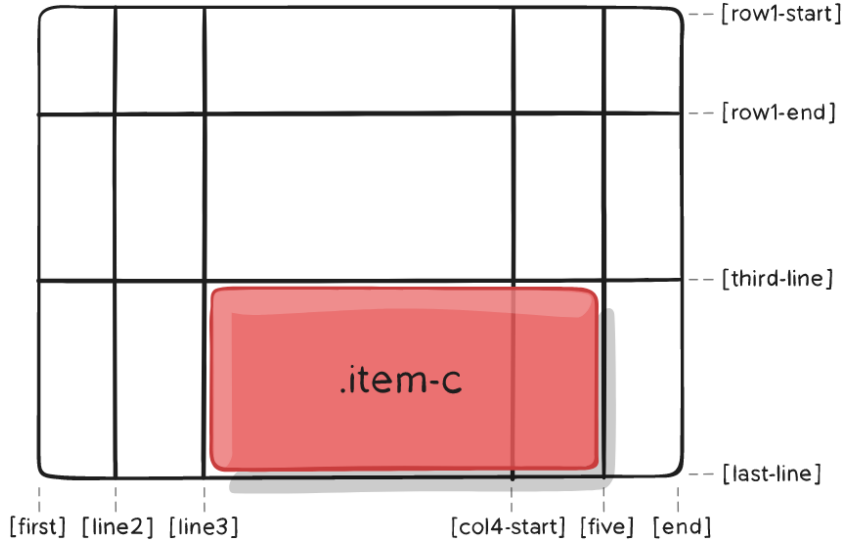
start | end
center | strech

align-self

start | end
center | strech

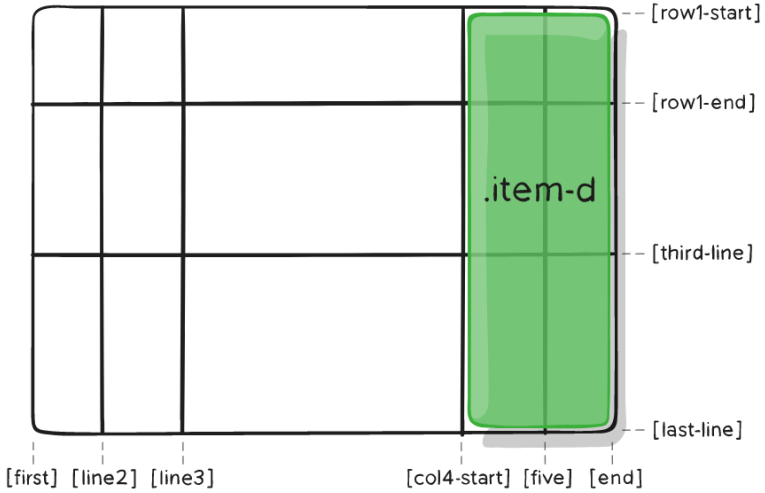
GRID COLUMN-ROW

```
.item-c {
  grid-column: 3 / span 2;
  grid-row: 3 / 4;
}
```



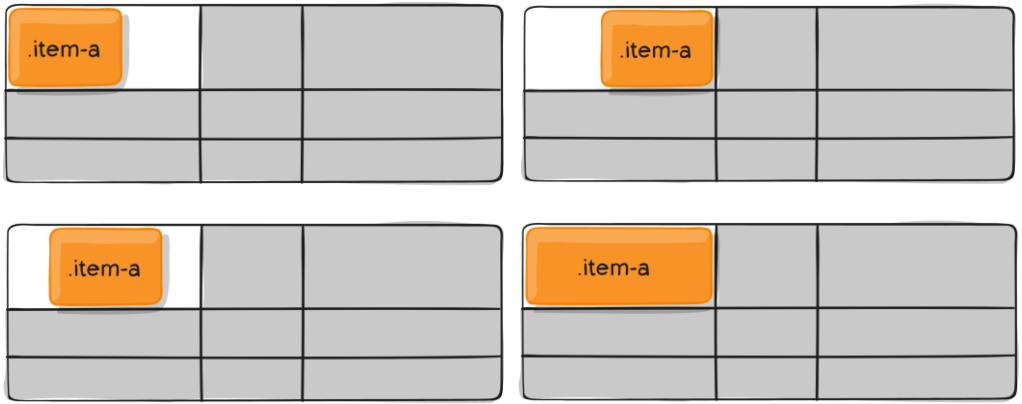
GRID AREA

```
.item {
  grid-area:
     <name> | <row-start> / <column-start>
     <row-end> / <column-end>;
}
.item-d {
  grid-area:
          aside 1 / col4-start / last-line / 6;
}
```



JUSTIFY SELF

```
.item {
  justify-self: start | end | center | stretch;
}
```

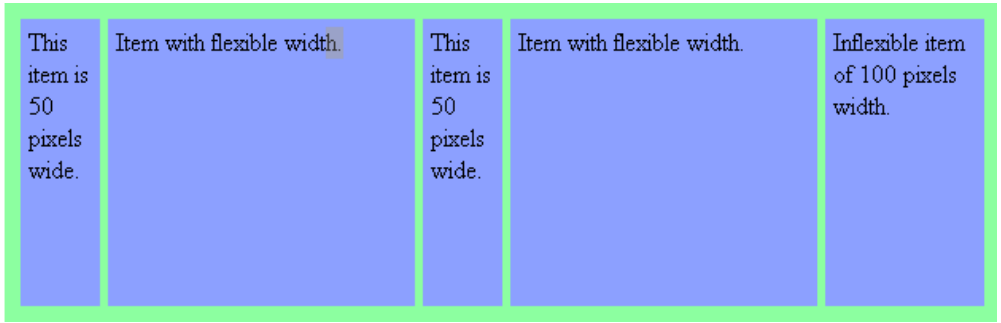


ALIGN SELF

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

FUNCION REPEAT

```
.container {
  display: grid;
  grid-template-columns: repeat(2, 50px 1fr) 100px;
}
```



GRACIAS!

¿Tienes más preguntas?

alex.amanz.95@gmail.com
55 1416 2446
atiwalkers.com.mx

https://learncssgrid.com/