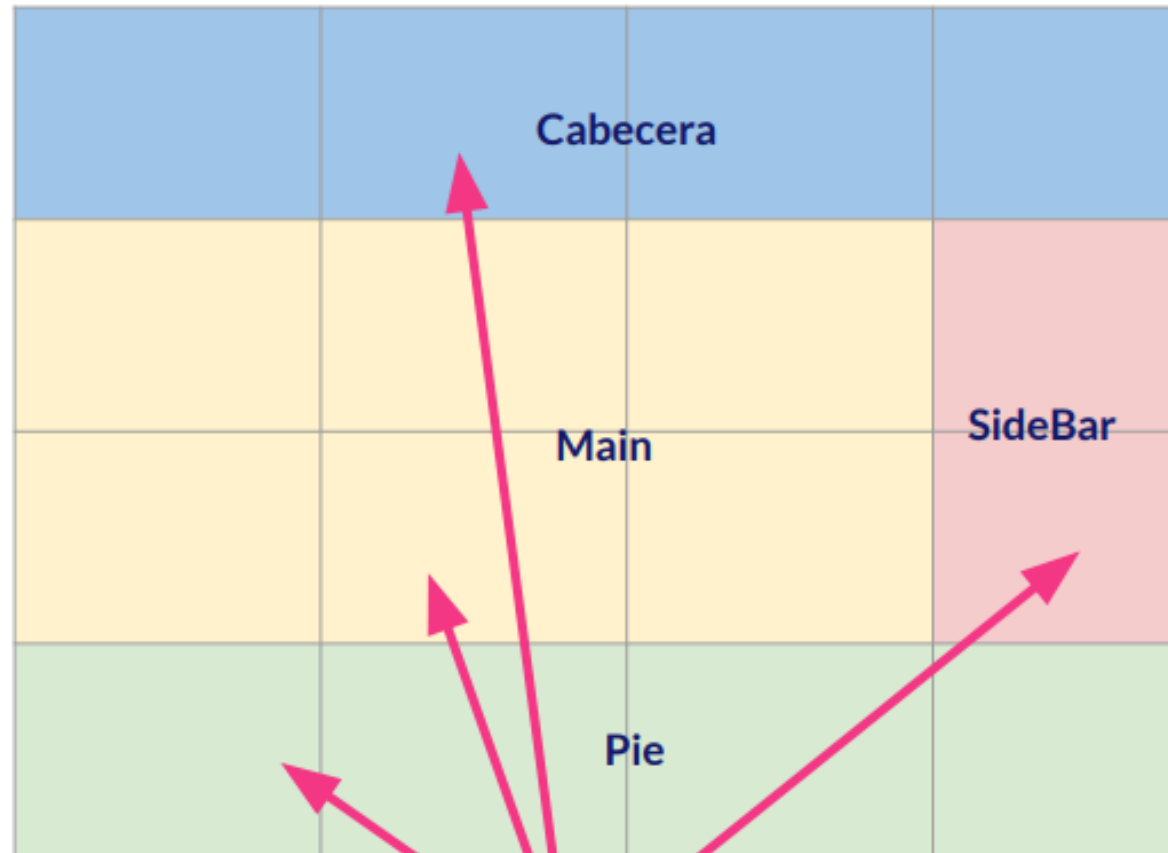


El contenedor GRID (`display:grid`)



Elementos del Grid

Contenedor con
`display: grid / inline-grid`

PARA DEFINIR LA ESTRUCTURA

- `grid-template-columns` : Número de columnas del grid y tamaño de las mismas
- `grid-template-rows`: Número de filas del grid y tamaño de las mismas.
- `grid-row-gap`: Separación entre las filas.
- `grid-column-gap`: Separación entre las columnas

COLUMNAS

`grid-template-columns`:

Debemos de poner tantos valores de anchura como columnas quiero que tenga el GRID.

`grid-template-columns: 20% 50% 30%`

`grid-template-columns: 100px auto 100px 100px`

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

`grid-template-columns : [id] 100px [nombre] 300px [apellidos] auto`

PARA DEFINIR LA ESTRUCTURA

- `grid-template-columns` : Número de columnas del grid y tamaño de las mismas
- `grid-template-rows`: Número de filas del grid y tamaño de las mismas.
- `grid-row-gap`: Separación entre las filas.
- `grid-column-gap`: Separación entre las columnas

COLUMNAS

`grid-template-columns`:

Debemos de poner tantos valores de anchura como columnas quiero que tenga el GRID.

`grid-template-columns: 20% 50% 30%`

`grid-template-columns: 100px auto 100px 100px`

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

`grid-template-columns : [id] 100px [nombre] 300px [apellidos] auto`

FILAS

- `grid-template-rows`: Debemos de poner tantos valores de altura como filas quiero que tenga el GRID.

`grid-template-rows: 20% 50% 30%`

`grid-template-rows: 100px auto 100px 100px`

`grid-template-rows: auto auto auto auto auto`

`grid-template-rows : [uno] 100px [dos] 300px [tres] auto`

REPETIR VALORES Y UNIDAD FR

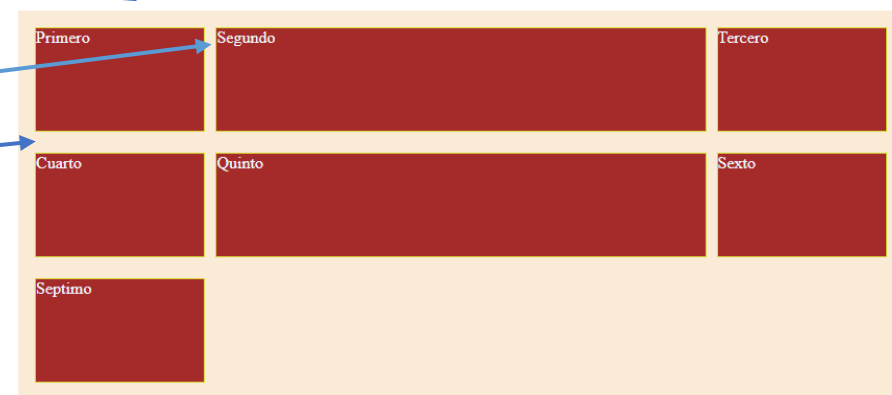
`grid-template-column: repeat(3, 20% [col-start]) auto`

`grid-template-column: 2fr 100px 1fr 2f`

Ejemplo

```
* {  
  box-sizing: border-box;  
}  
  
.container {  
  background-color: antiquewhite;  
  display: grid;  
  grid-column-gap: 10px;  
  grid-row-gap: 20px;  
  grid-template-columns: 20% auto 20%;  
  grid-template-rows: repeat(3, 100px);  
  margin: 20px auto;  
  padding: 1em;  
  width: 80%;  
}  
  
.container > div {  
  background-color: brown;  
  border: 1px solid rgb(215, 218, 55);  
  color: aliceblue;  
}
```

grid-template-columns: 20% auto 20%

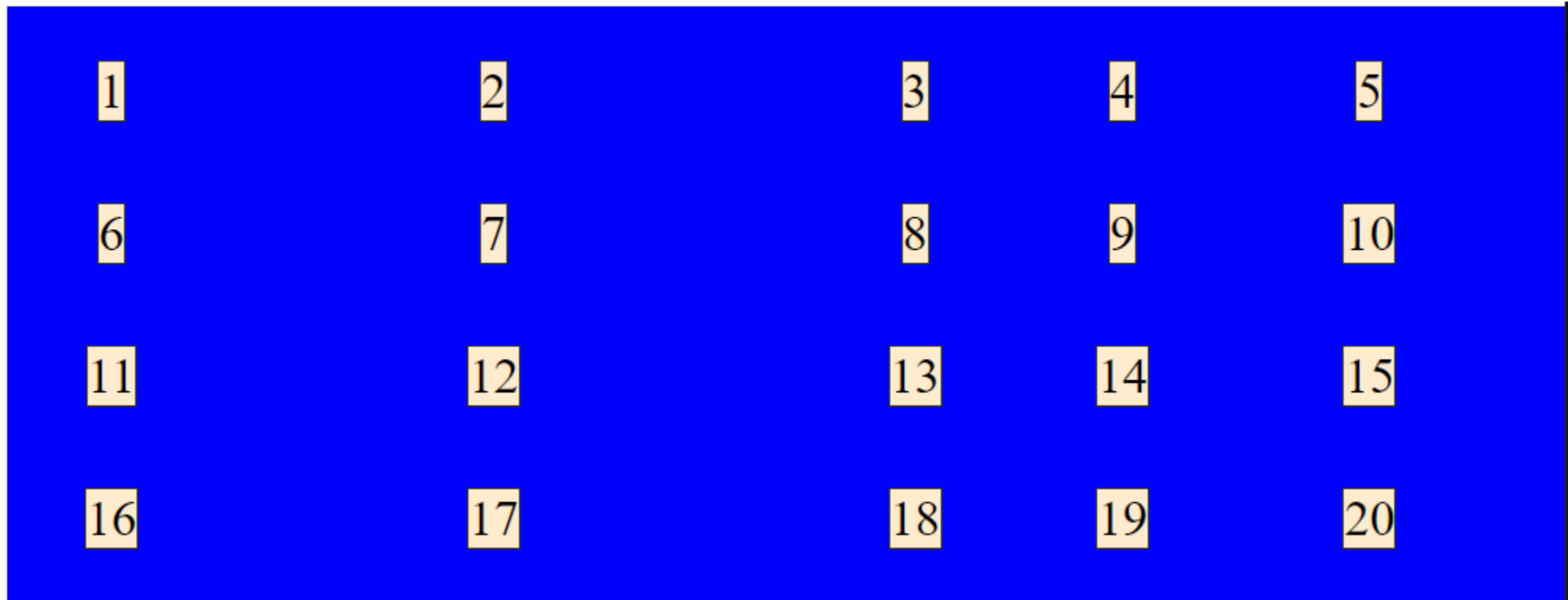


grid-template-rows: repeat(3, 100px);

```
justify-items: end; /*alineacion horizontal /start-center-end*/  
align-items: end ; /*alineacion vertical / start-center-end*/
```

Estas opciones nos sirven para alinear los contenidos

Valores por defecto `stretch`



```
justify-content: center ;
```

```
/*a linear el contenedor horizontalmente start/center/end*/
```

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

Para alinear horizontalmente el contenido que hay dentro de cada celda usaré la propiedad CSS justify-items

Para alinear verticalmente el contenido que hay dentro de cada celda usaré la propiedad CSS align-items

start

end

start

end

center

stretch

center

stretch


```
justify-items: center;
```

```
align-items: center;
```

Primero

Segundo

Tercero

Cuarto

Quinto

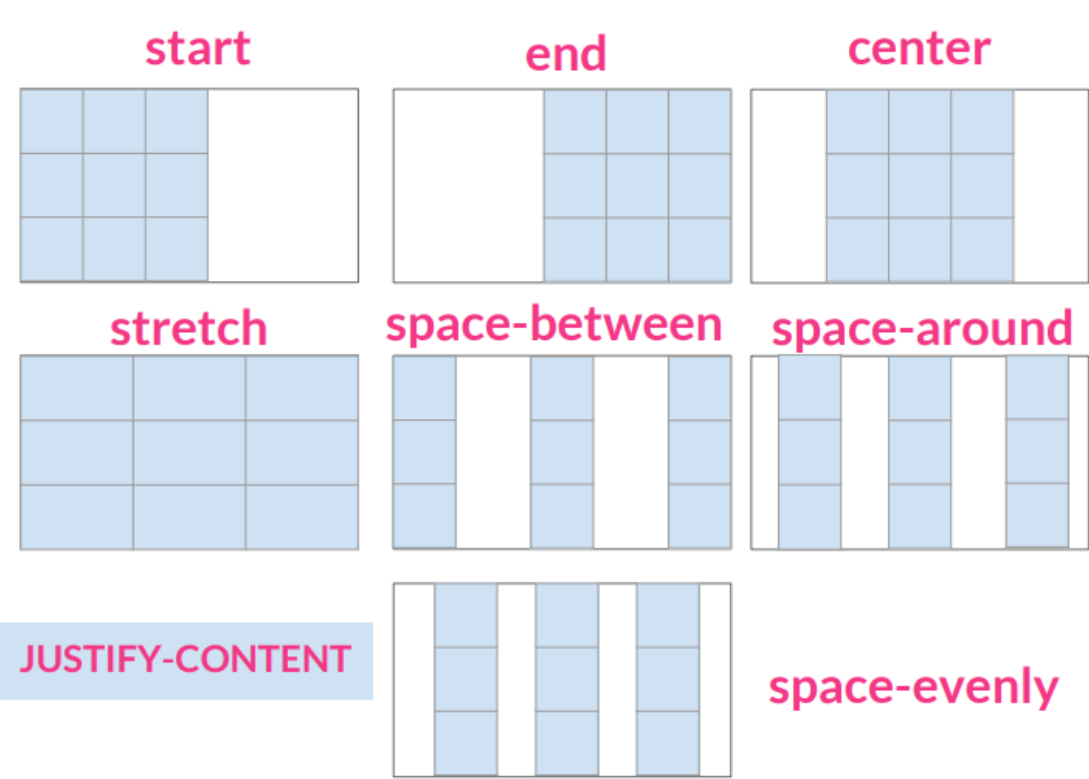
Sexto

Septimo

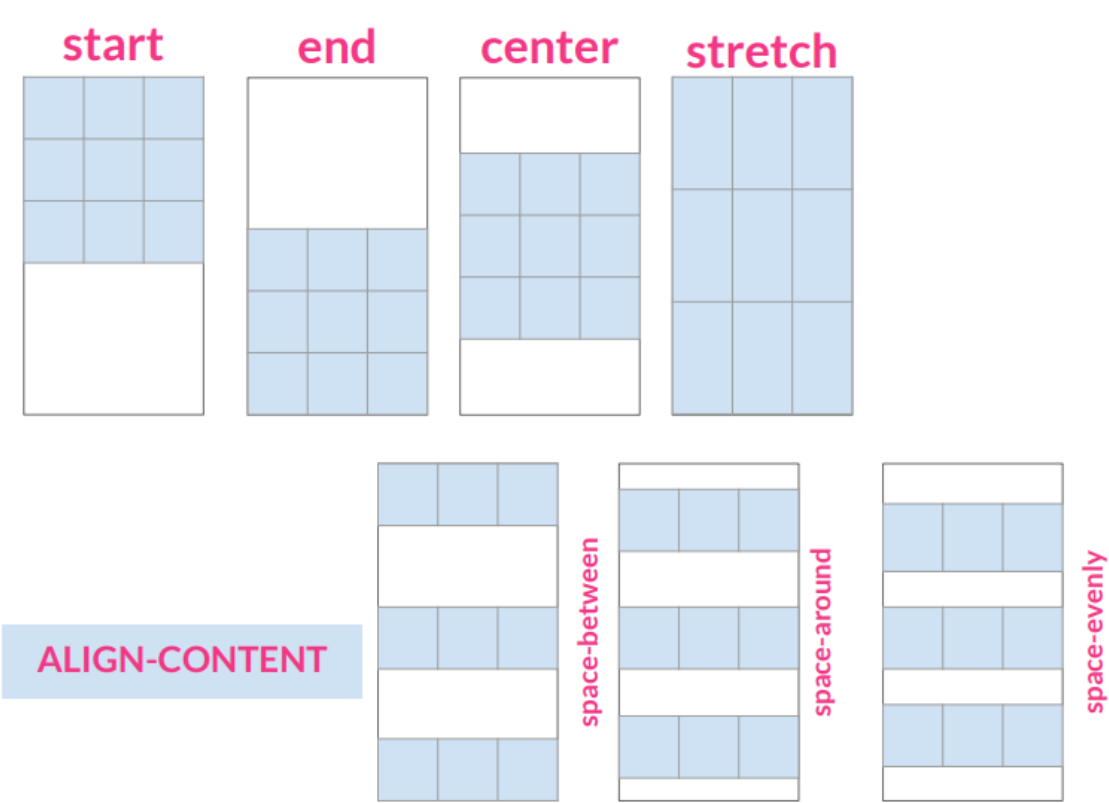
Septimo

En el caso de que el GRID no ocupe todo el contenedor GRID podemos distribuirlo con las propiedades CSS

justify-content (horizontal)

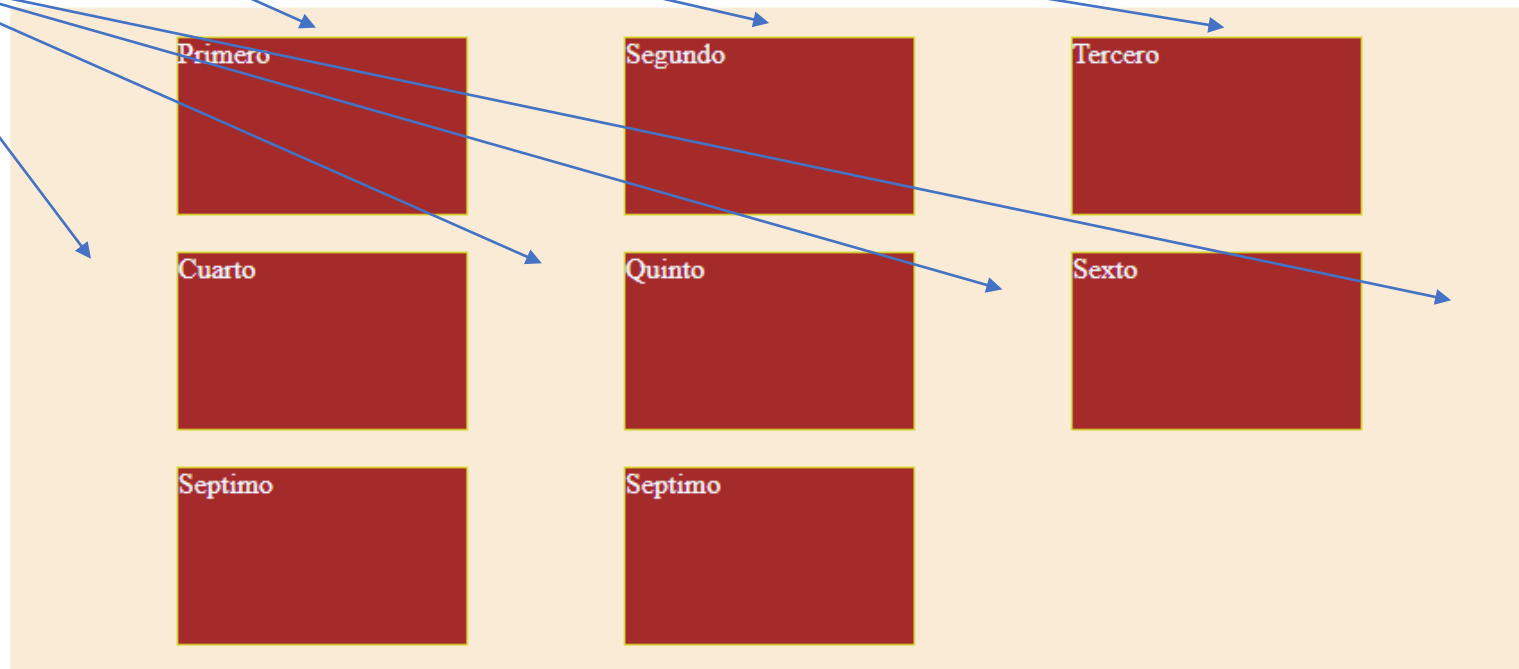


align-content (vertical)



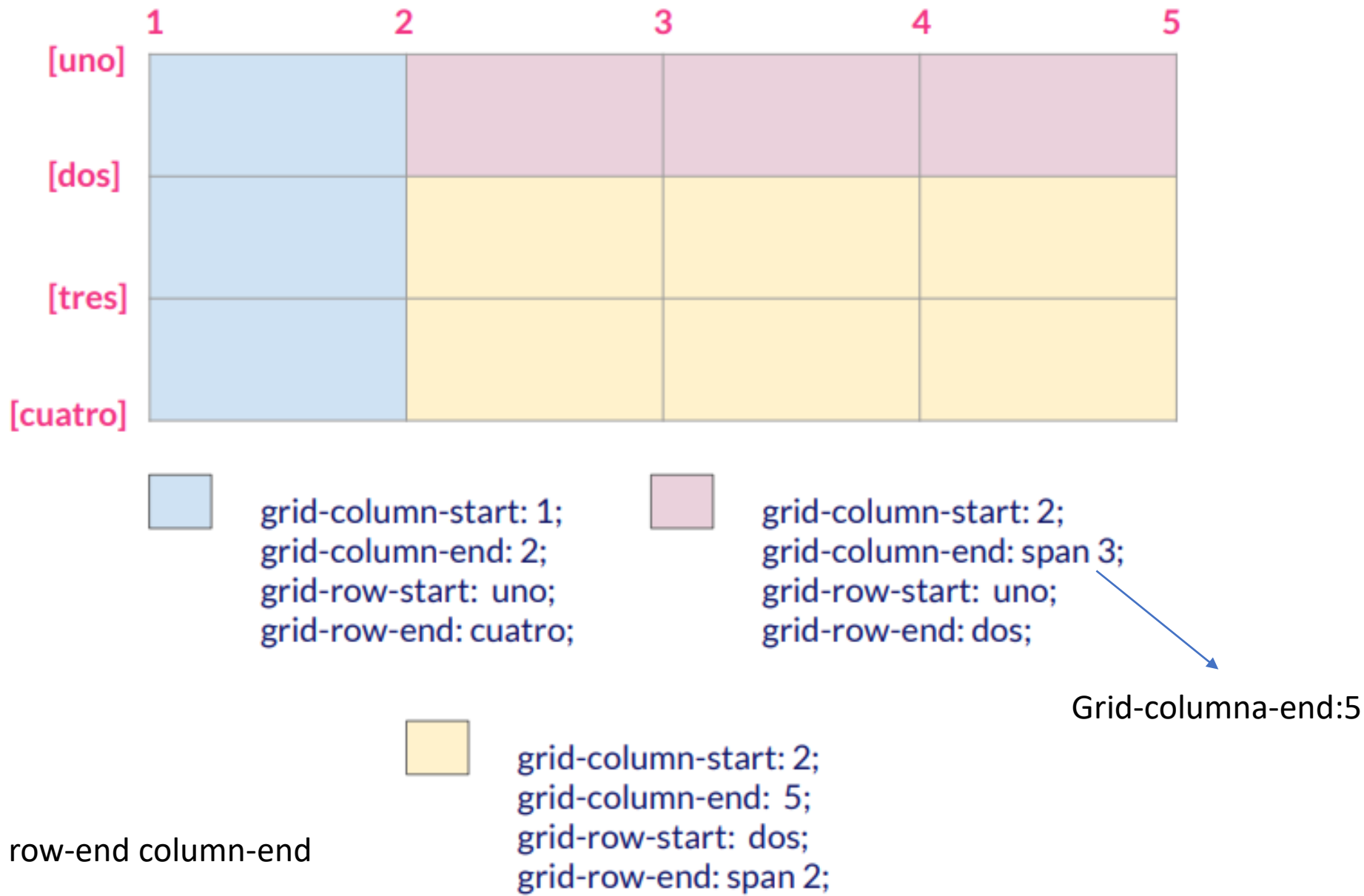
Ejemplo

```
.container {  
  background-color: antiquewhite;  
  display: grid;  
  grid-column-gap: 10px;  
  grid-row-gap: 20px;  
  grid-template-columns: 20% 20% 20%;  
  grid-template-rows: repeat(3, 100px);  
  justify-content: space-evenly;  
  margin: 20px auto;  
  padding: 1em;  
  width: 80%;  
}
```



PARA ESPECIFICAR EL AREA

- grid-column-start
- grid-column-end
- grid-row-start
- grid-row-end



grid-column: start / end
grid-row: start / end
grid: row-start column-start row-end column-end

```

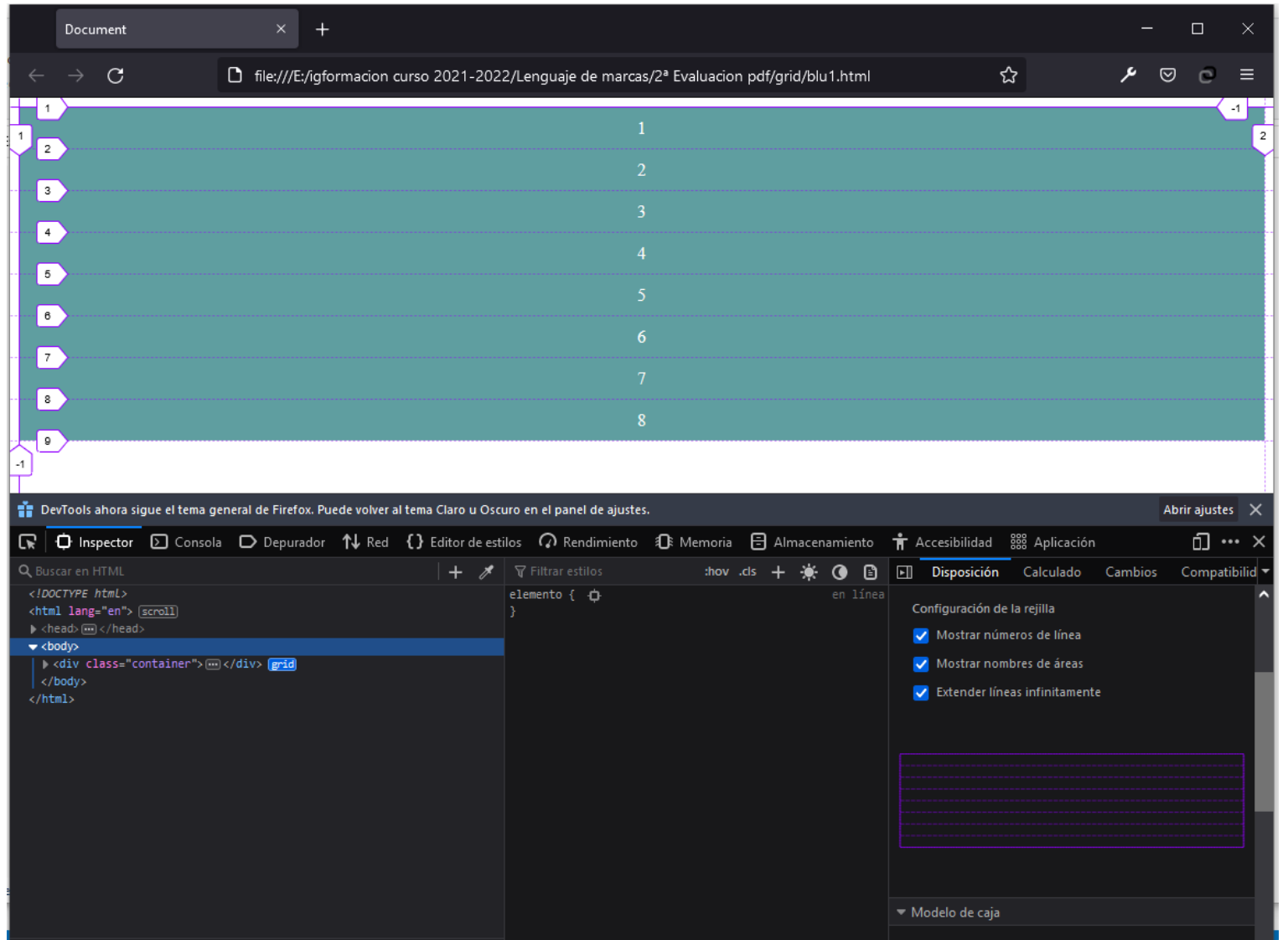
<div class="container">
  <div class="cajas">1</div>
  <div class="cajas">2</div>
  <div class="cajas">3</div>
  <div class="cajas">4</div>
  <div class="cajas">5</div>
  <div class="cajas">6</div>
  <div class="cajas">7</div>
  <div class="cajas">8</div>
</div>

```

```

.container {
  display: grid;
}
.cajas {
  background-color: cadetblue;
  padding: 10px;
  color: #fff;
  text-align: center;
}

```



```
.container {
  display: grid;
  /*grid-template-columns: 100px 100px 100px;*/anchos en pixeles
  /*grid-template-columns: 50% 30% 20%;*/anchos en porcentajes
  /*grid-template-columns: 1fr 300px 3fr;*/anchos pixeles y fracciones
  grid-template-columns: repeat(3,1fr);
  /*grid-template-rows: 2fr 1fr 3fr;*/
  grid-template-rows: repeat(3,1fr);
  /*grid-auto-rows: 200px Todas las cajas adicionales*/
  grid-gap: 15px; /*Distancia entre filas y columnas*/
  /*height: 100vh; altura de la ventana*/
  /*column-gap: 15px;
  row-gap: 5px;*/
}

.cajas {
  background-color: cadetblue;
  padding: 10px;
  color: #fff;
  text-align: center;
}
```

```
<div class="container">
  <div class="cajas">1</div>
  <div class="cajas">2</div>
  <div class="cajas">3</div>
  <div class="cajas">4</div>
  <div class="cajas">5</div>
  <div class="cajas">6</div>
  <div class="cajas">7</div>
  <div class="cajas">8</div>
</div>
```

```

<div class="grid-container">
  <div class="item1">1</div>
  <div class="item2">2</div>
  <div class="item3">3</div>
  <div class="item4">4</div>
  <div class="item5">5</div>
  <div class="item6">6</div>
  <div class="item7">7</div>
  <div class="item8">8</div>
  <div class="item9">9</div>
  <div class="item10">10</div>
  <div class="item11">11</div>
  <div class="item12">12</div>
  <div class="item13">13</div>
  <div class="item14">14</div>
  <div class="item15">15</div>
</div>

```

```

.grid-container {
  display: grid;
  grid-template-columns: auto auto auto auto auto auto;
  gap: 10px;
  background-color: #2196F3;
  padding: 10px;
}

```

```

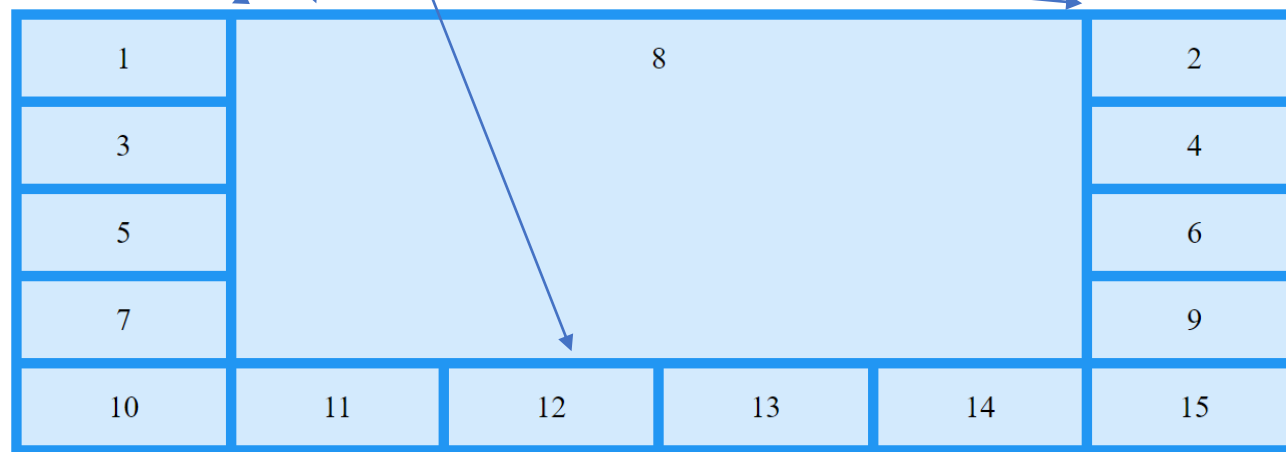
.grid-container > div {
  background-color: rgba(255, 255, 255, 0.8);
  text-align: center;
  padding: 20px 0;
  font-size: 30px;
}

```

```

.item8 {
  grid-area: 1 / 2 / 5 / 6;
}

```



(1765) CSS Grid -  Curso desde cero  Tutorial en Español - YouTube

CSS Grid Layout (w3schools.com)