

PUERTA 18

# FLEXBOX

curso

<Front End>

# Qué ofrece CSS flexbox

Antes de Flexbox, para distribuir “componentes” en la horizontal o en la vertical, de forma diferente a lo predeterminado, se usaban subsistemas de propiedades css a partir de `table`, `float`, `position`, `display: inline-block`...

Eran esencialmente “hacks” (“emparches”).

Dejaban de lado funcionalidades importantes (distribución vertical, por ejemplo).

# Qué ofrece CSS flexbox

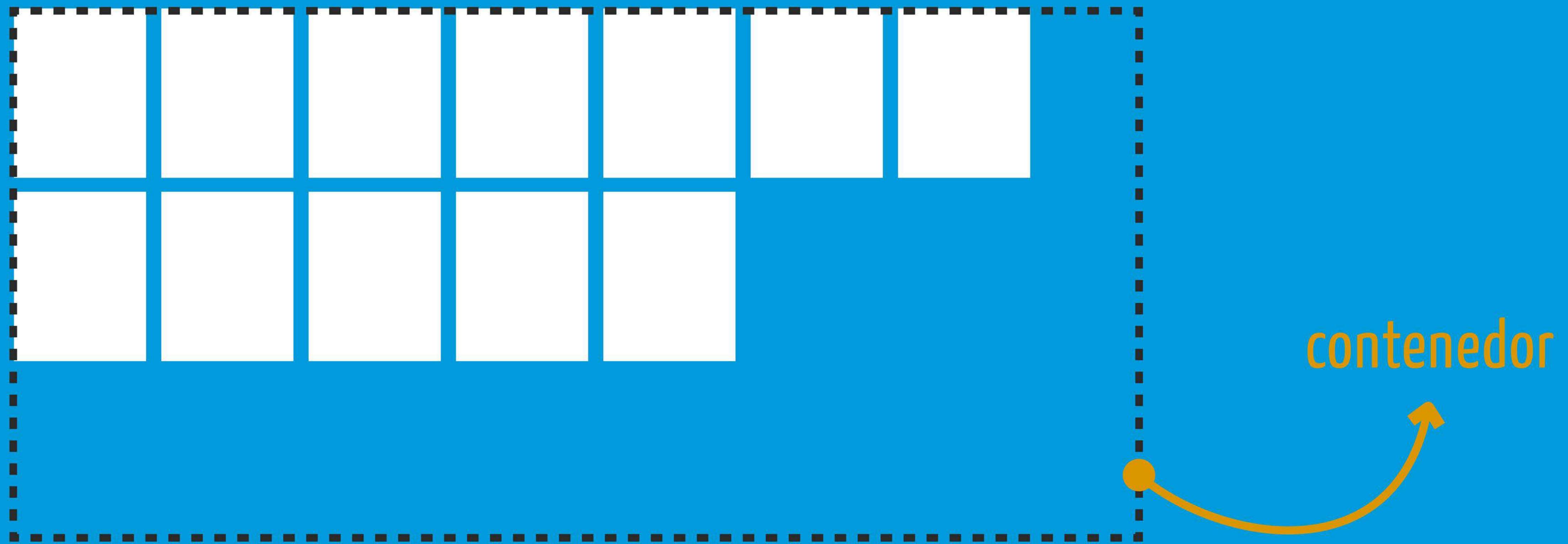
**Flexbox es el primer subsistema CSS  
creado específicamente para resolver  
los problemas de diseño que hemos  
estado “emparchando” desde que se  
comenzaron a diseñar sitios web**

# Cómo funciona CSS flexbox

- Se aplica CSS Flexbox a un módulo contenedor que tenga elementos contenidos;
- Es una relación entre “padres” e “hijos” (“nietos”, “bisnietos” etc. pueden participar de esta relación de forma indirecta y con “efectos colaterales”);
- Estas relaciones también pueden anidarse para incluir “nietos”, “bisnietos”, etc.;
- Los elementos contenidos pueden colocarse en cualquier orden mediante CSS;

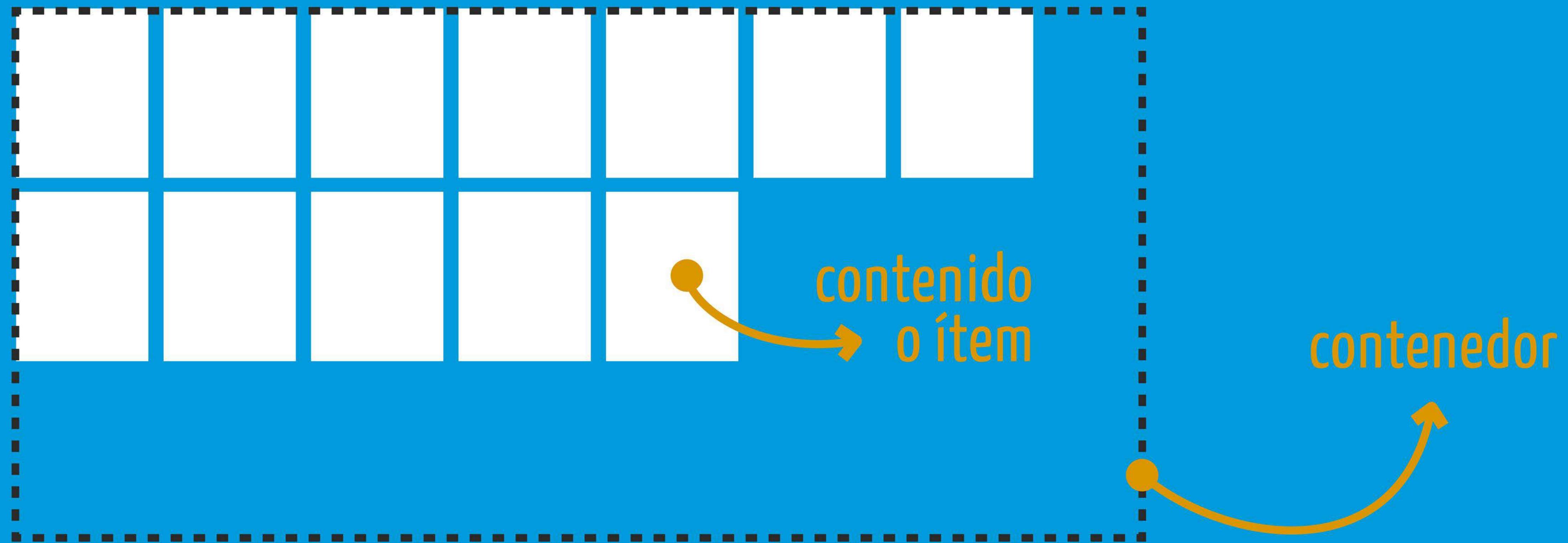
# Algunos conceptos, para ponernos de acuerdo:

Una “caja flexible” es un contenedor que organiza sus elementos “hijos” según un eje principal y un eje secundario



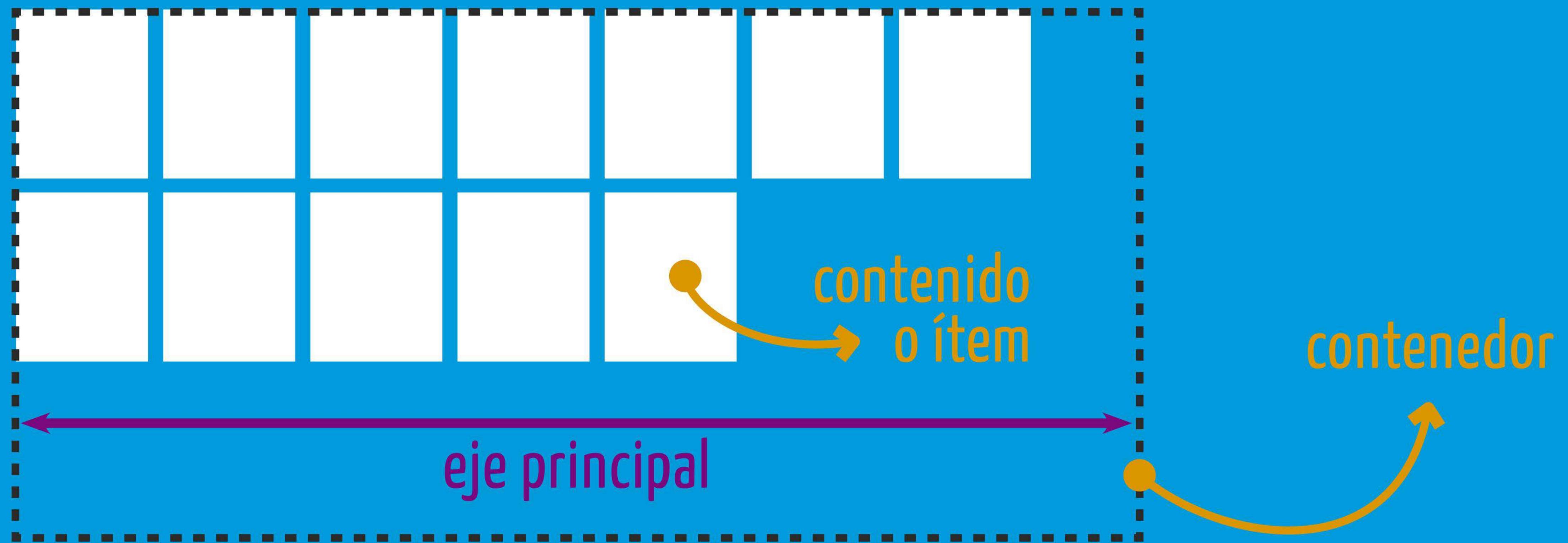
# Algunos conceptos, para ponernos de acuerdo:

Una “caja flexible” es un contenedor que organiza sus elementos “hijos” según un eje principal y un eje secundario



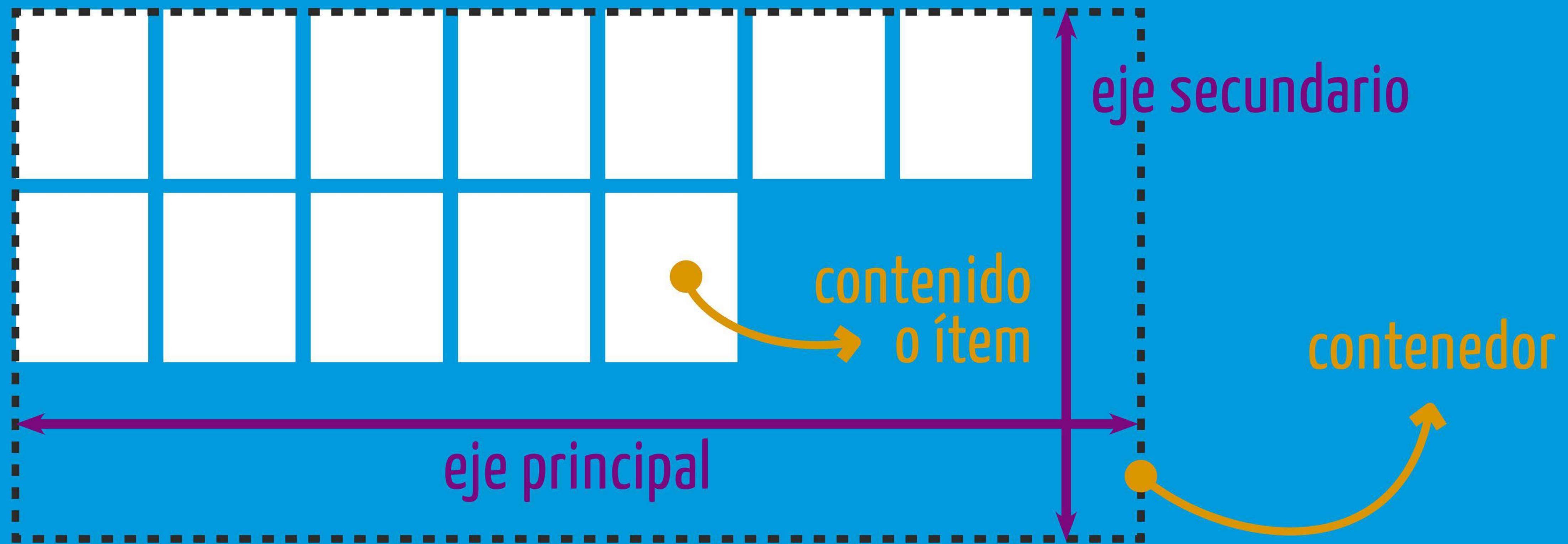
# Algunos conceptos, para ponernos de acuerdo:

Una “caja flexible” es un contenedor que organiza sus elementos “hijos” según un eje principal y un eje secundario



# Algunos conceptos, para ponernos de acuerdo:

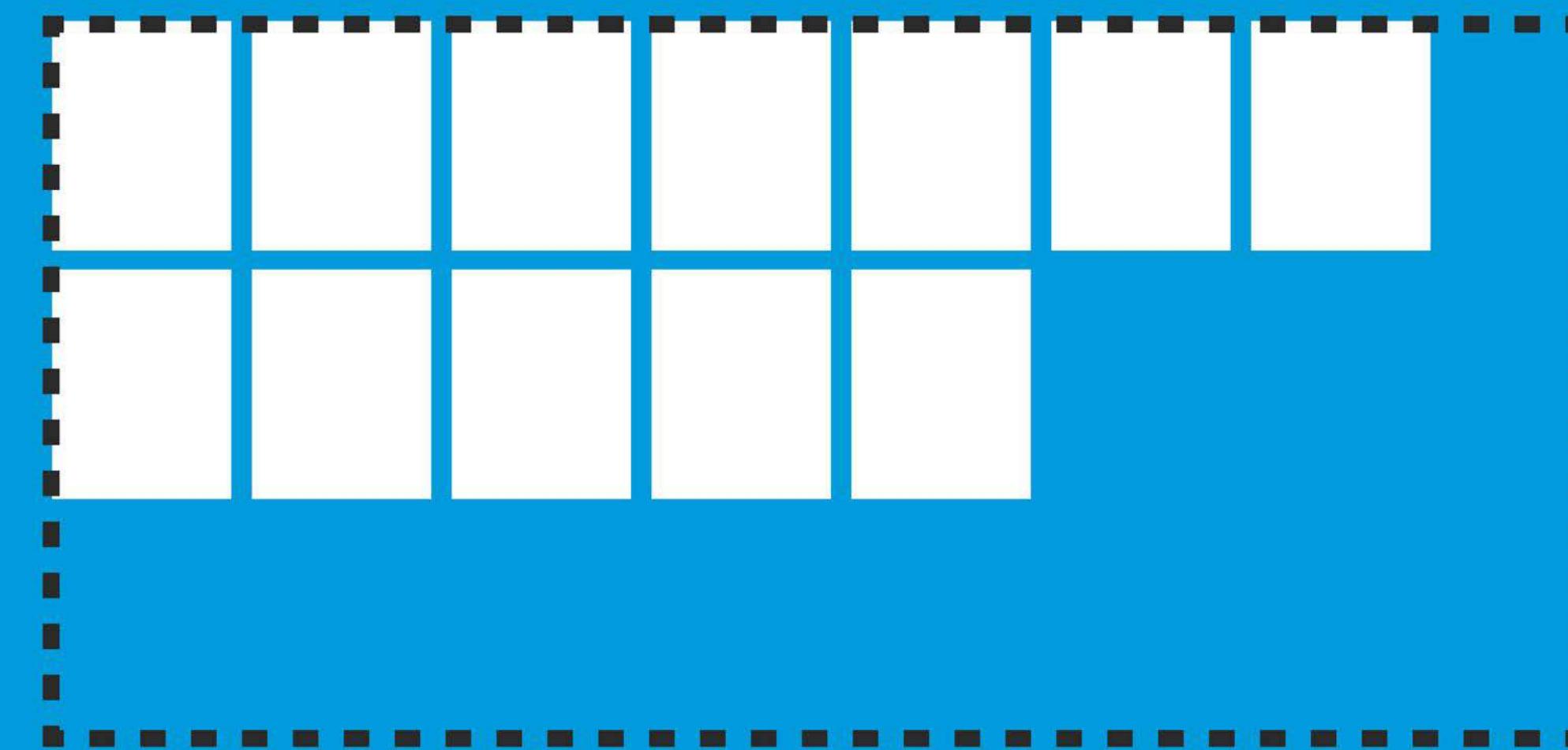
Una “caja flexible” es un contenedor que organiza sus elementos “hijos” según un eje principal y un eje secundario



# Algunos conceptos, para ponernos de acuerdo:

Los elementos “hijos” de una “caja flexible” se convierten en “cajitas inline”:

- su ancho pasa a depender de su contenido y ya no ocupan el 100% del espacio disponible como las “cajas normales”;
- en principio, se posicionan horizontalmente, diferente a las “cajas normales”;
- se expanden o achican para llenar el espacio libre.



# Propiedades CSS para el MÓDULO CONTENEDOR

- `display`
- `flex-direction`
- `flex-wrap`
- `justify-content`
- `align-items`
- `align-content`
- `gap`

## Propiedades CSS para el CONTENEDOR:

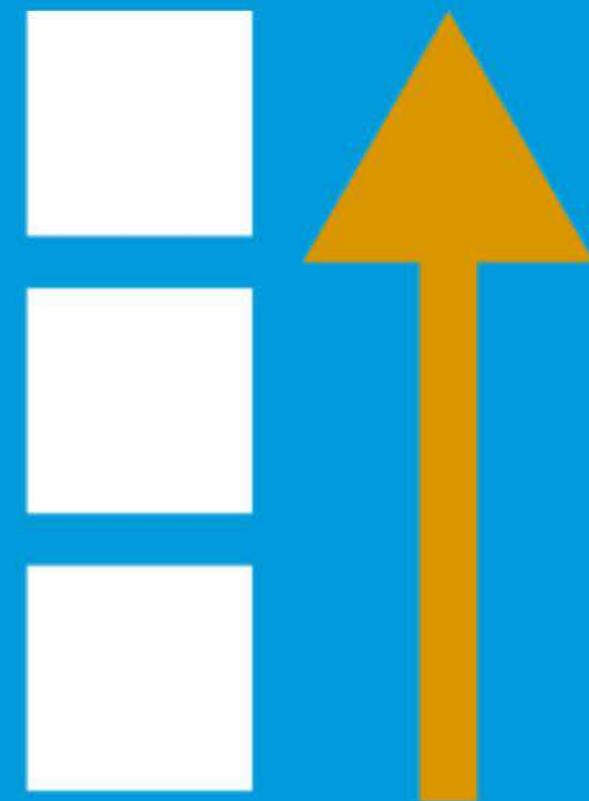
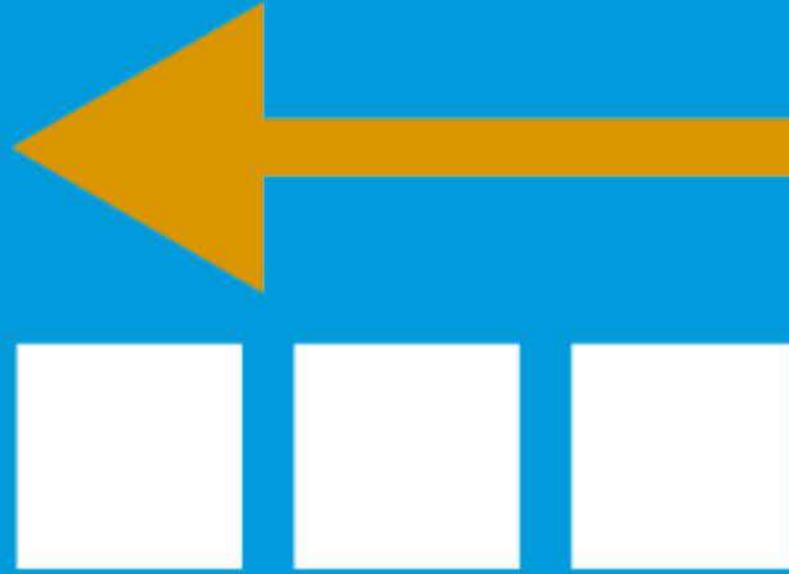
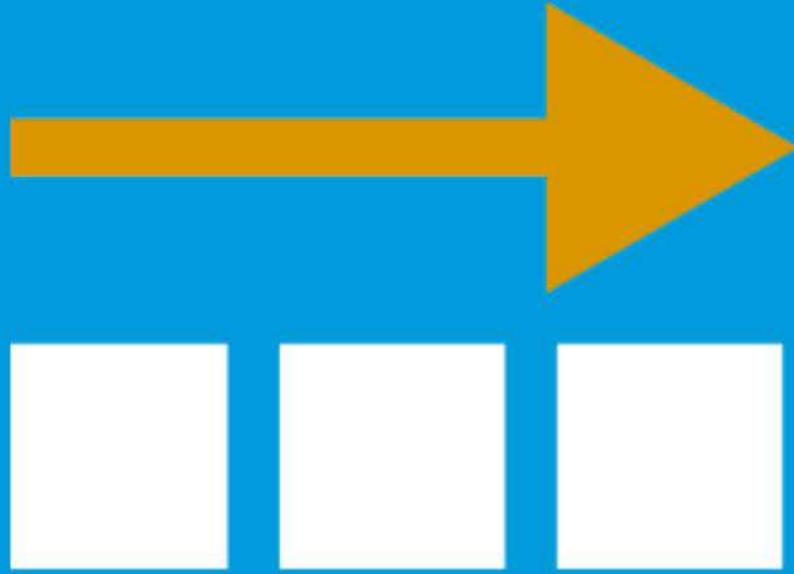
**display**

Antes de configurar otras propiedades,  
hay que indicarle al contenedor que sea una “caja flexible”.

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

Propiedades CSS para el CONTENEDOR:

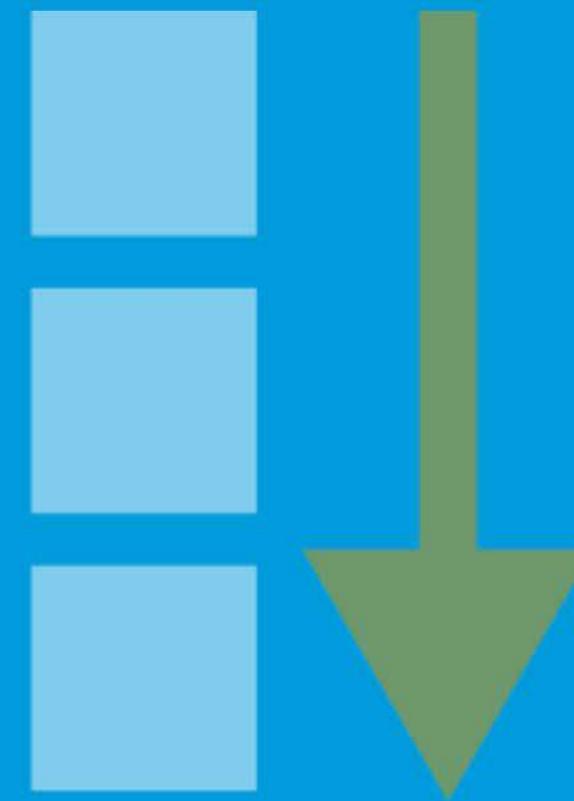
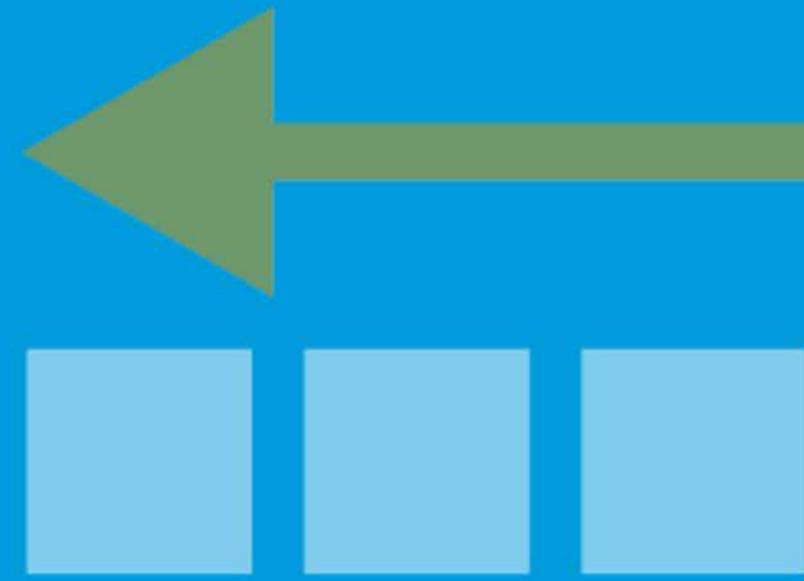
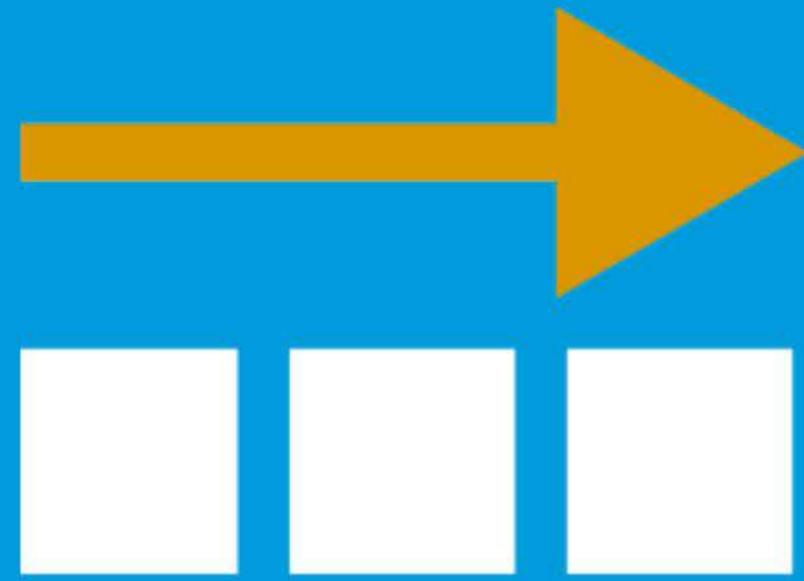
**flex-direction**



Modifica la dirección y el sentido del eje principal  
del contenedor y por ende, del eje secundario  
El valor predeterminado es **row**

Propiedades CSS para el CONTENEDOR:

**flex-direction**



**flex-direction: row;**

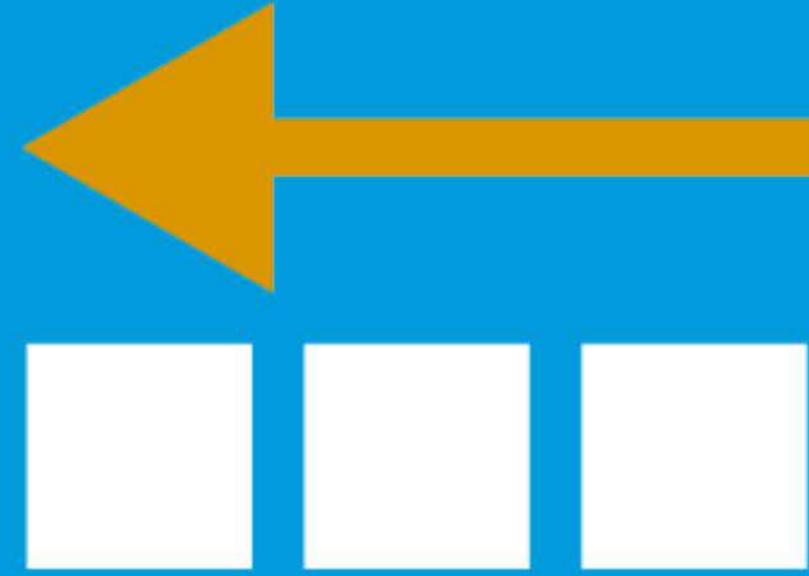
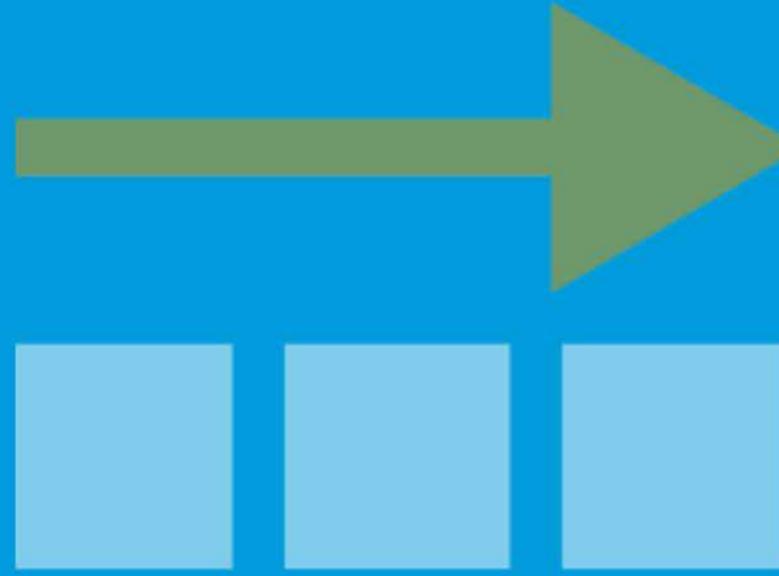
**flex-direction: row-reverse;**

**flex-direction: column;**

**flex-direction: column-reverse;**

Propiedades CSS para el CONTENEDOR:

**flex-direction**



**flex-direction: row;**

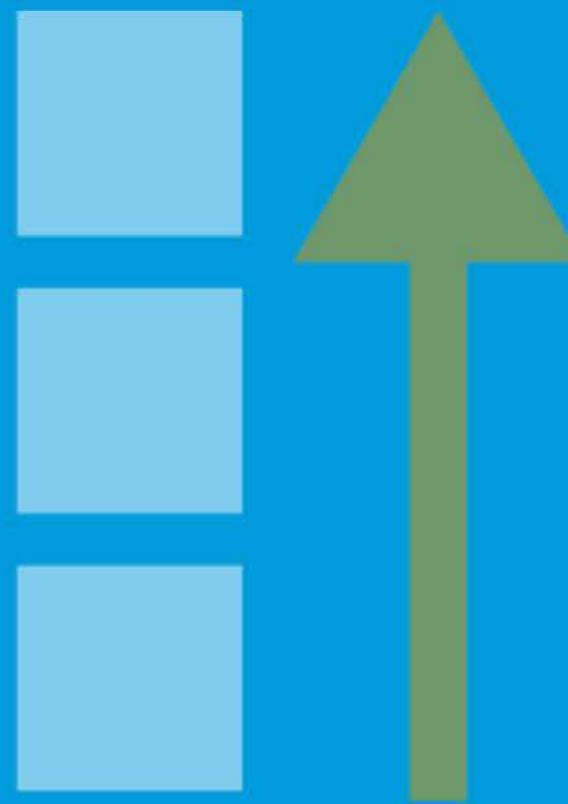
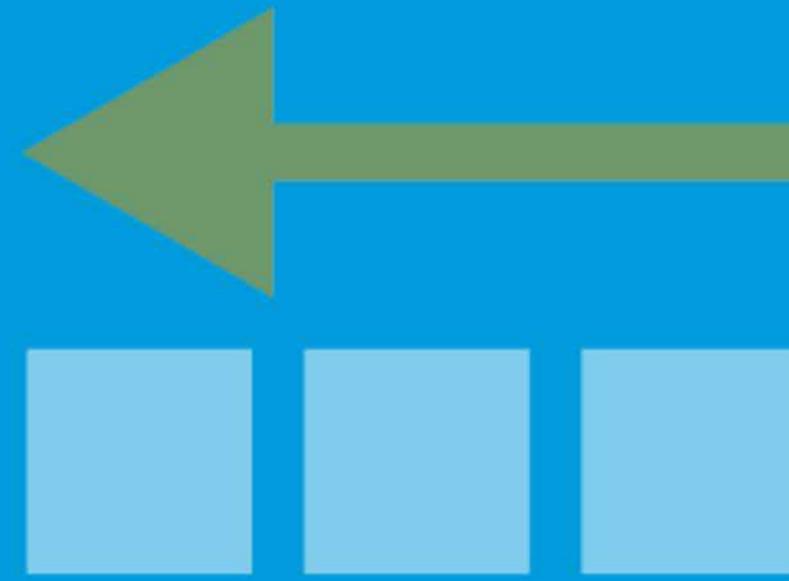
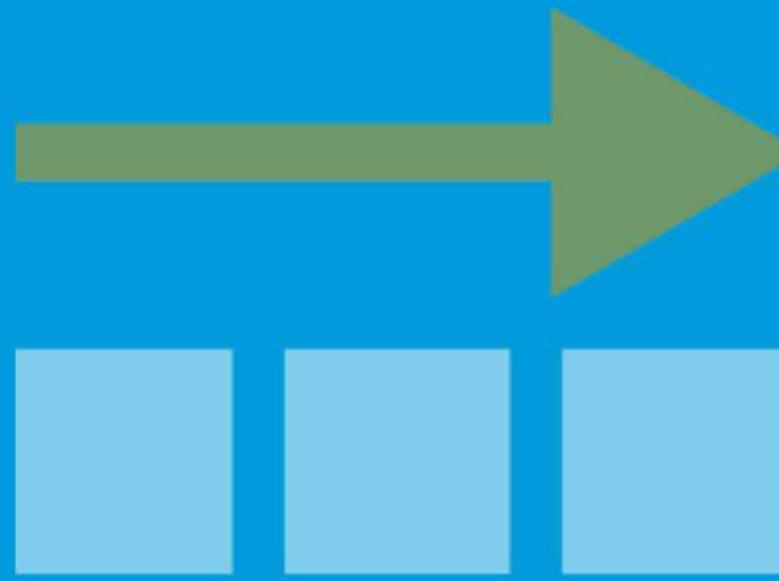
**flex-direction: row-reverse;**

**flex-direction: column;**

**flex-direction: column-reverse;**

Propiedades CSS para el CONTENEDOR:

**flex-direction**



**flex-direction: row;**

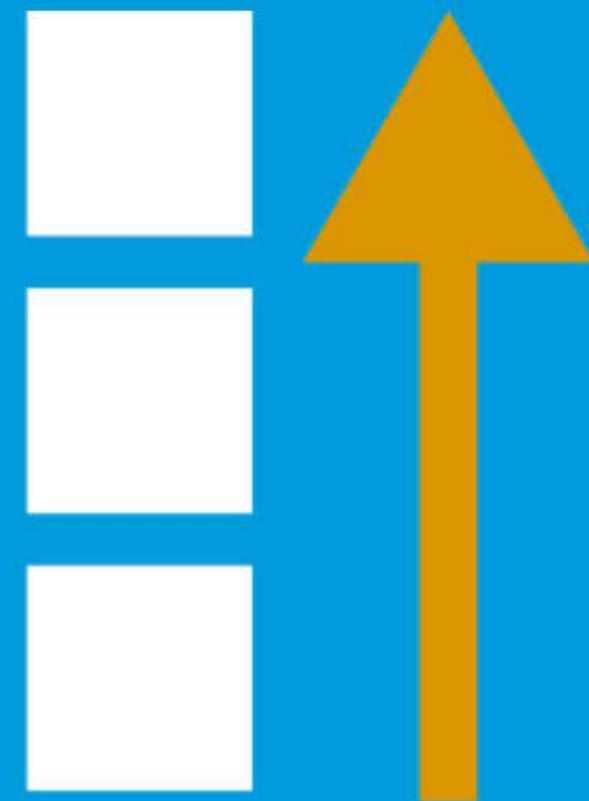
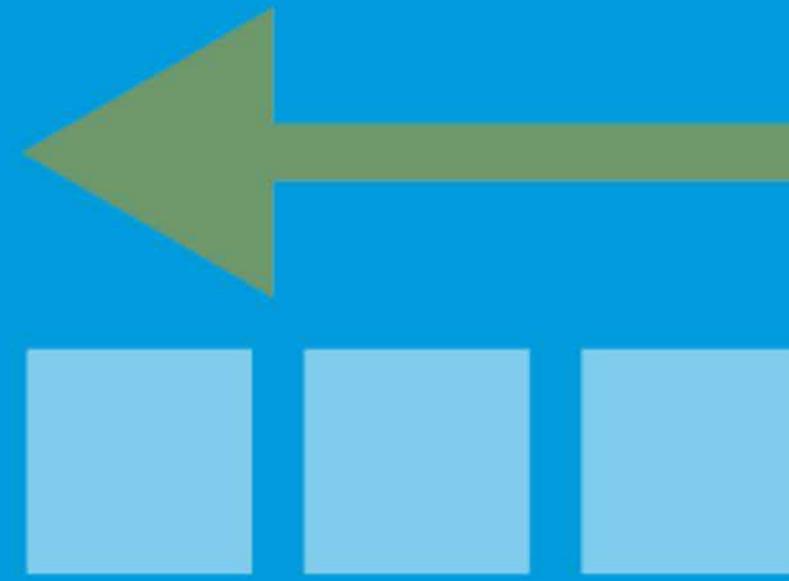
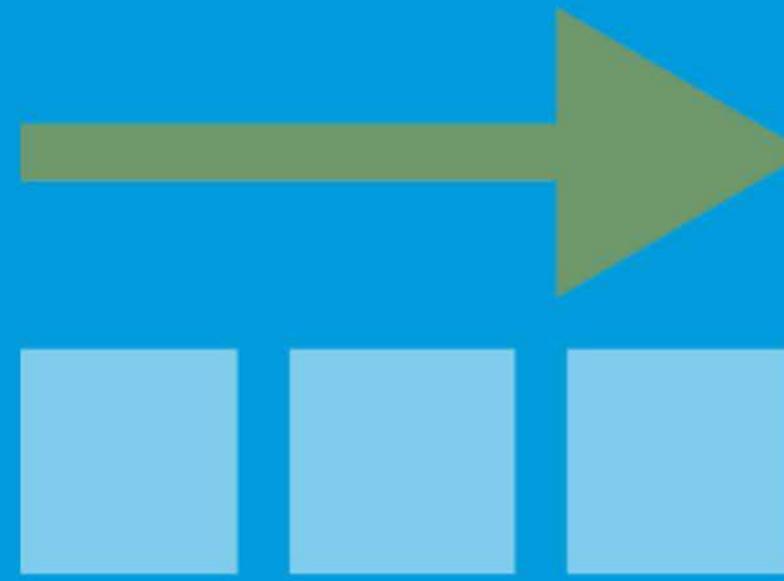
**flex-direction: row-reverse;**

**flex-direction: column;**

**flex-direction: column-reverse;**

Propiedades CSS para el CONTENEDOR:

**flex-direction**



**flex-direction: row;**

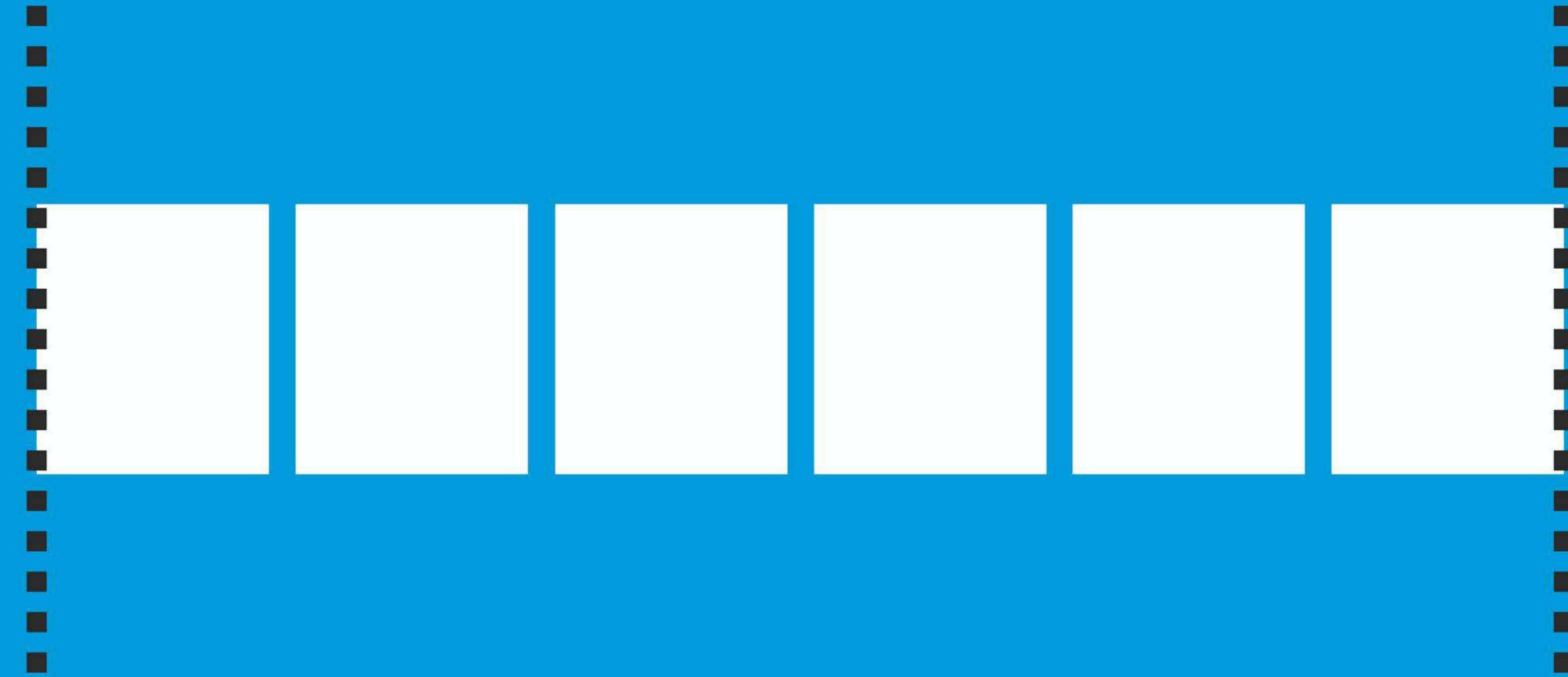
**flex-direction: row-reverse;**

**flex-direction: column;**

**flex-direction: column-reverse;**

## Propiedades CSS para el CONTENEDOR:

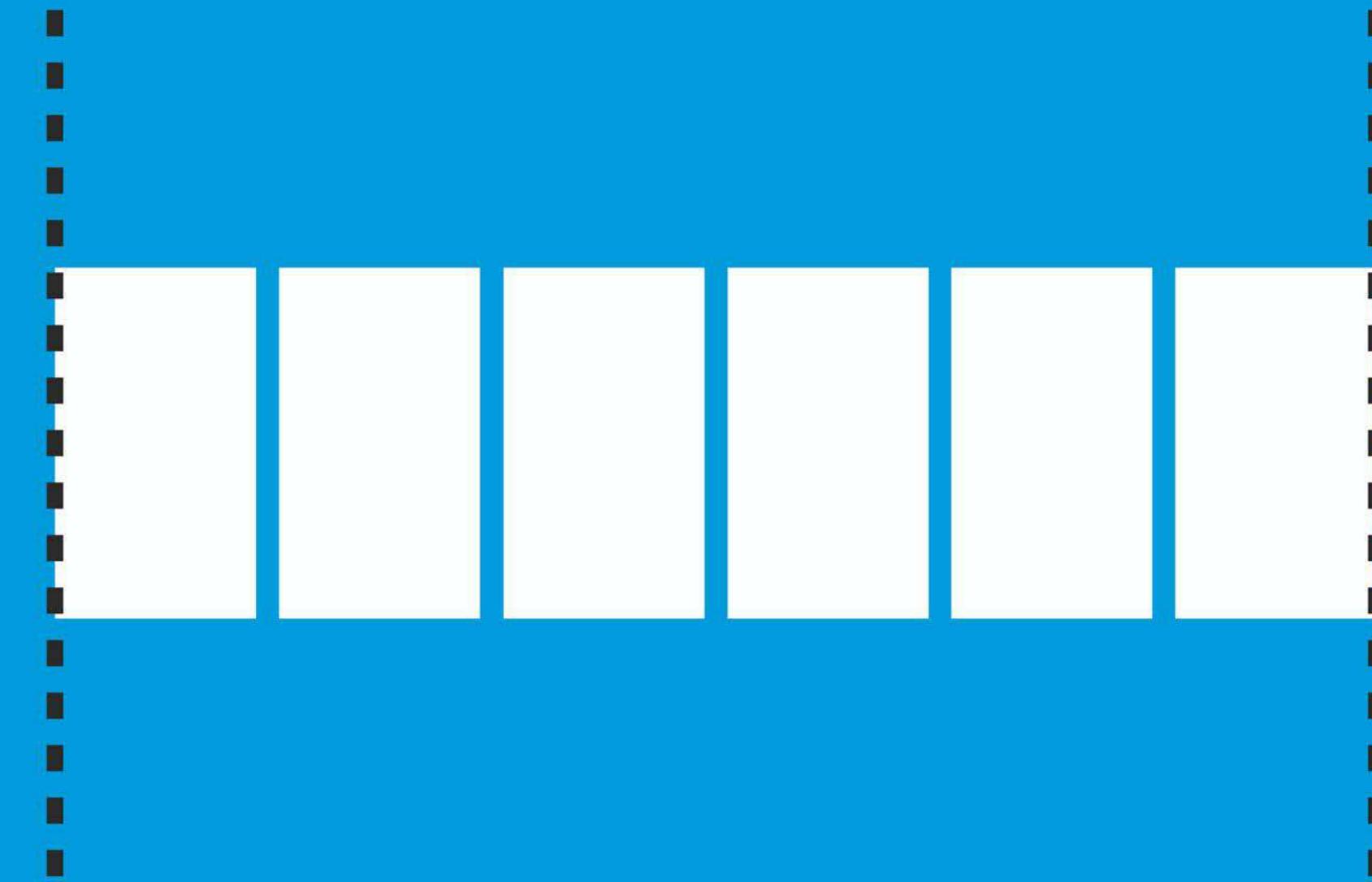
**flex-wrap**



Especifica el comportamiento del contenedor respecto al desborde o no desborde de los elementos cuando los mismos no caben en una sola línea de ítems  
El valor predeterminado es nowrap

Propiedades CSS para el CONTENEDOR:

**flex-wrap**



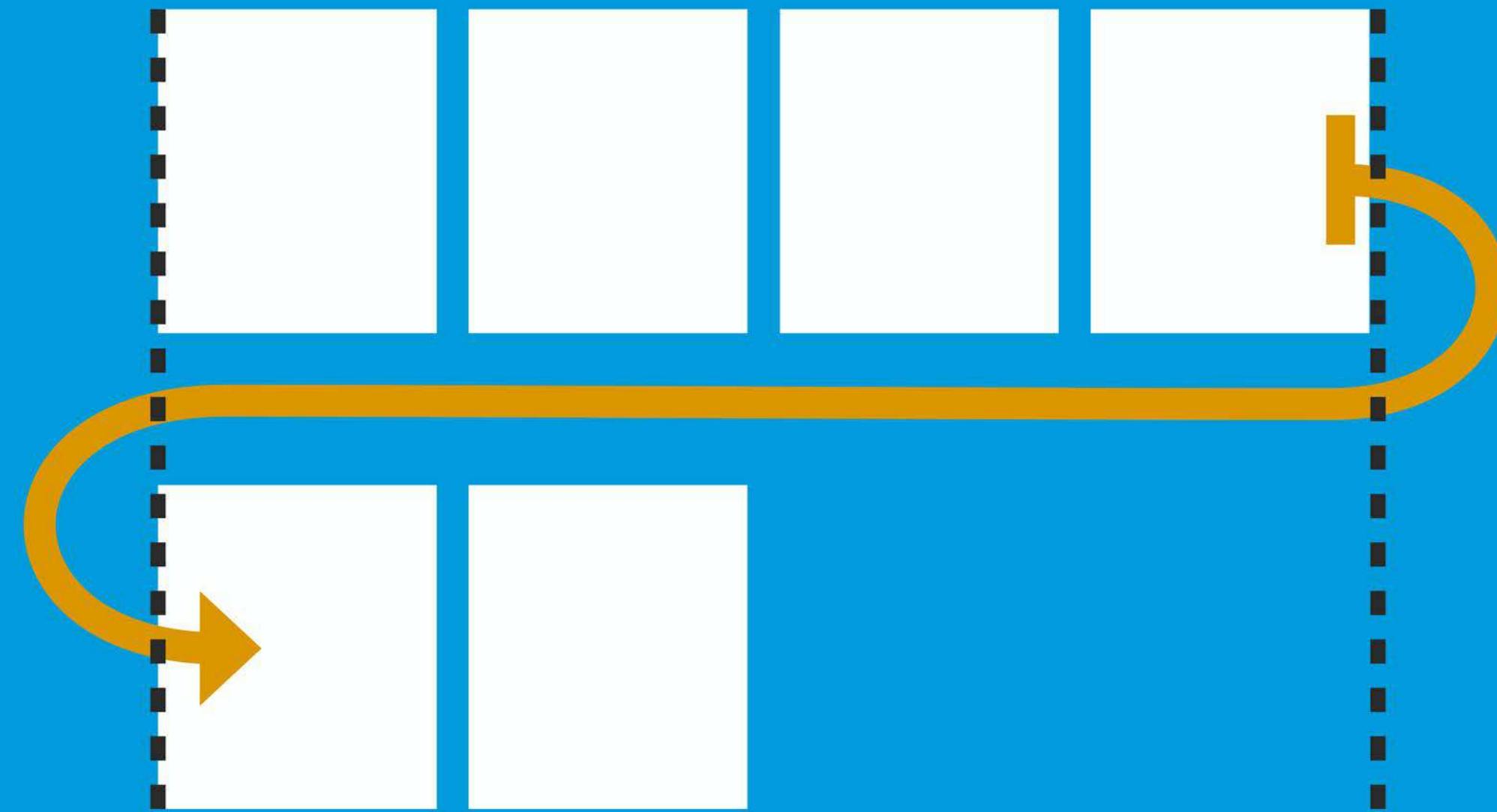
**flex-wrap: nowrap;**

**flex-wrap: wrap;**

**flex-wrap: wrap-reverse;**

Propiedades CSS para el CONTENEDOR:

**flex-wrap**



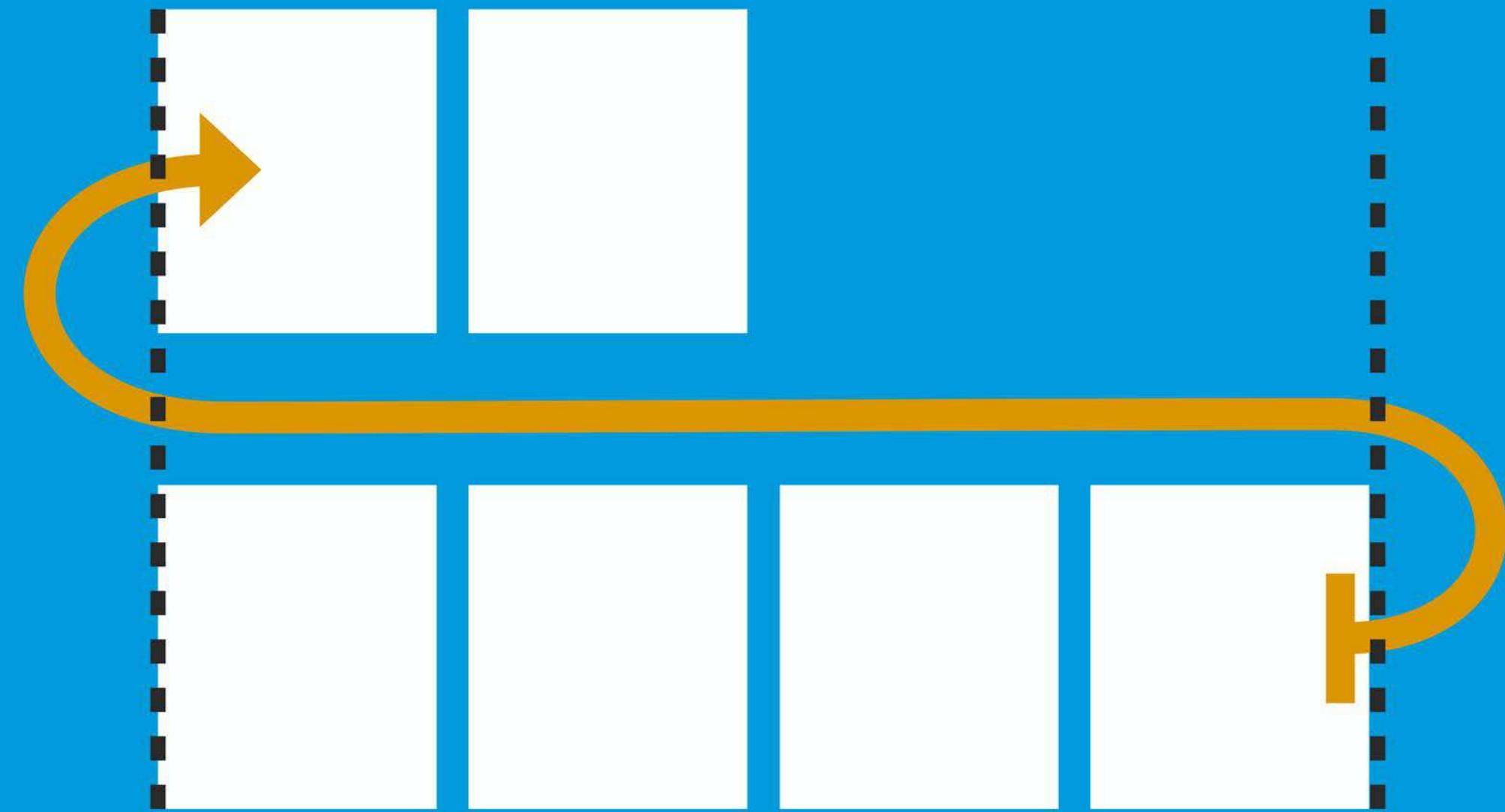
**flex-wrap: nowrap;**

**flex-wrap: wrap;**

**flex-wrap: wrap-reverse;**

Propiedades CSS para el CONTENEDOR:

**flex-wrap**



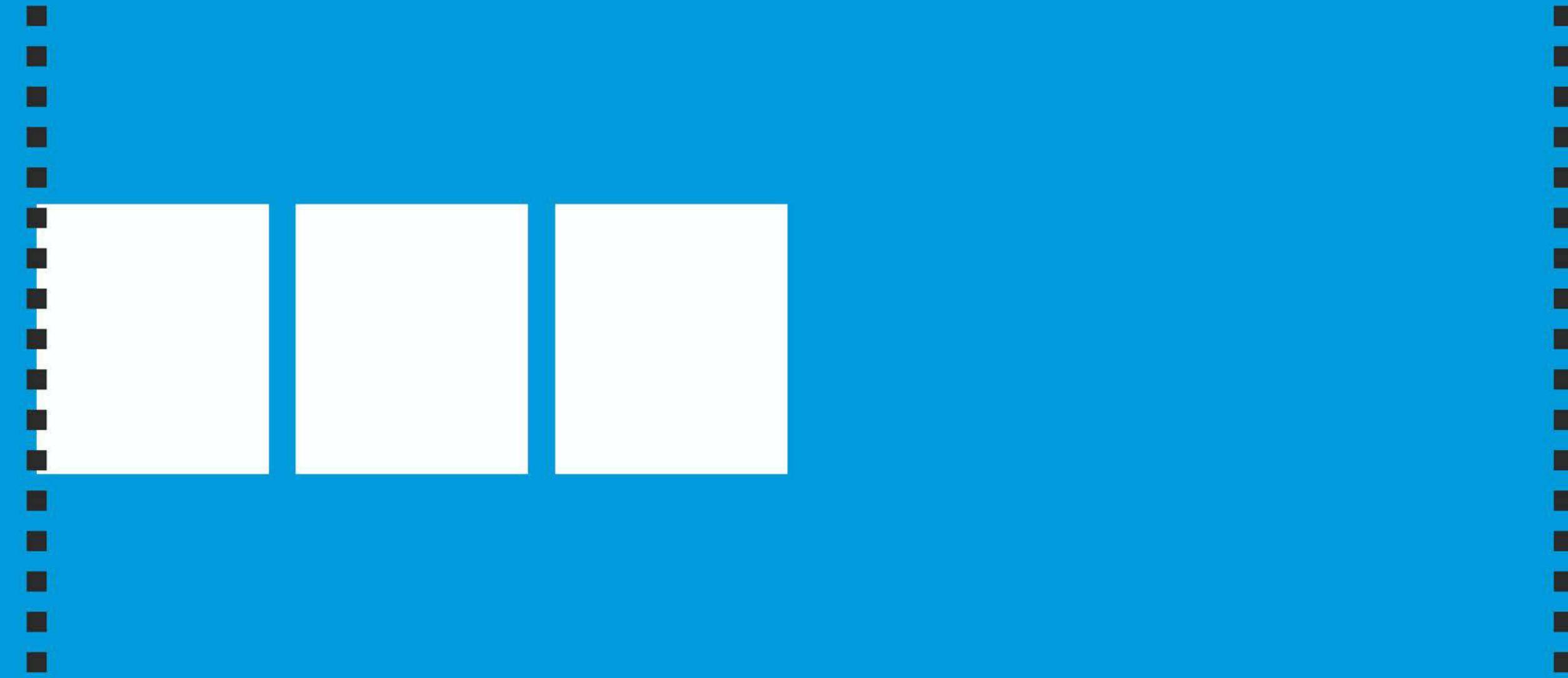
`flex-wrap: nowrap;`

`flex-wrap: wrap;`

`flex-wrap: wrap-reverse;`

Propiedades CSS para el CONTENEDOR:

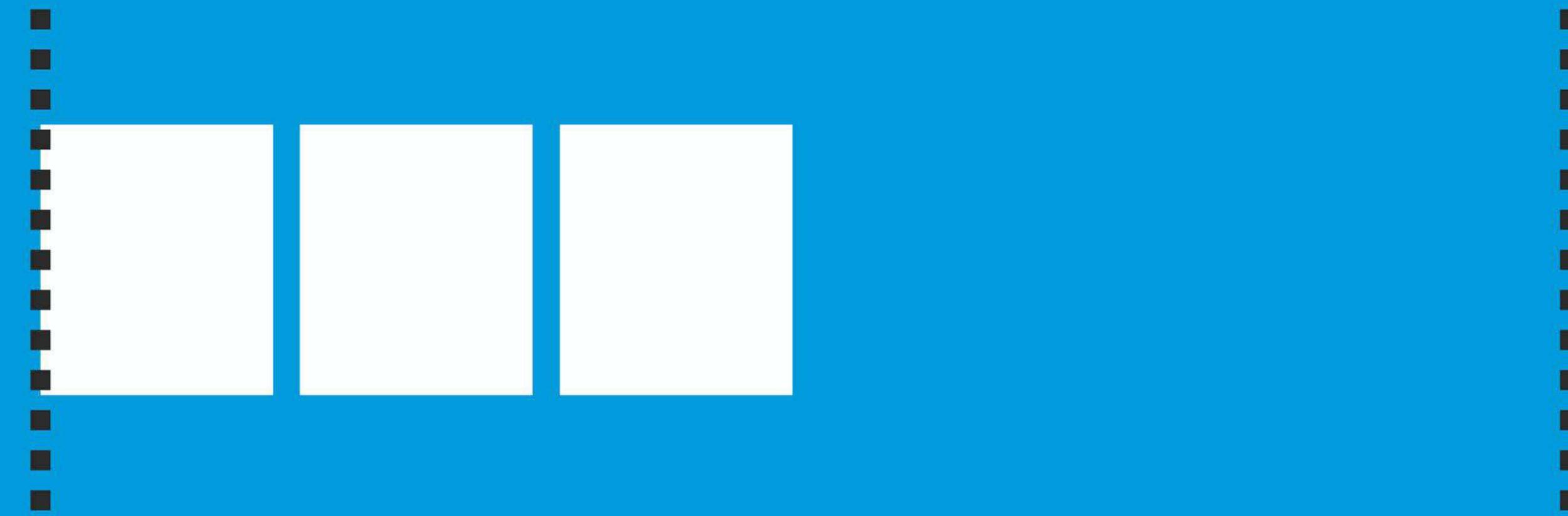
**justify-content**



Especifica la distribución de los contenidos según el eje principal  
El valor predeterminado es flex-start

Propiedades CSS para el CONTENEDOR:

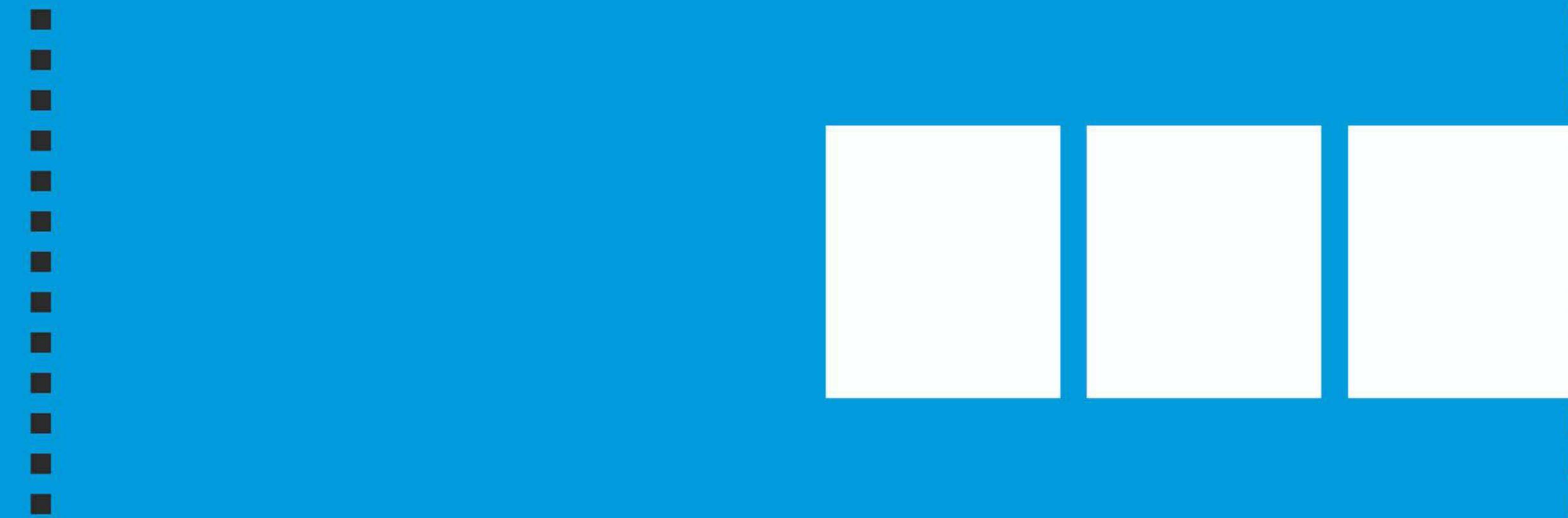
**justify-content**



**justify-content: flex-start;**  
**justify-content: flex-end;**  
**justify-content: center;**  
**justify-content: space-between;**  
**justify-content: space-around;**  
**justify-content: space-evenly;**

Propiedades CSS para el CONTENEDOR:

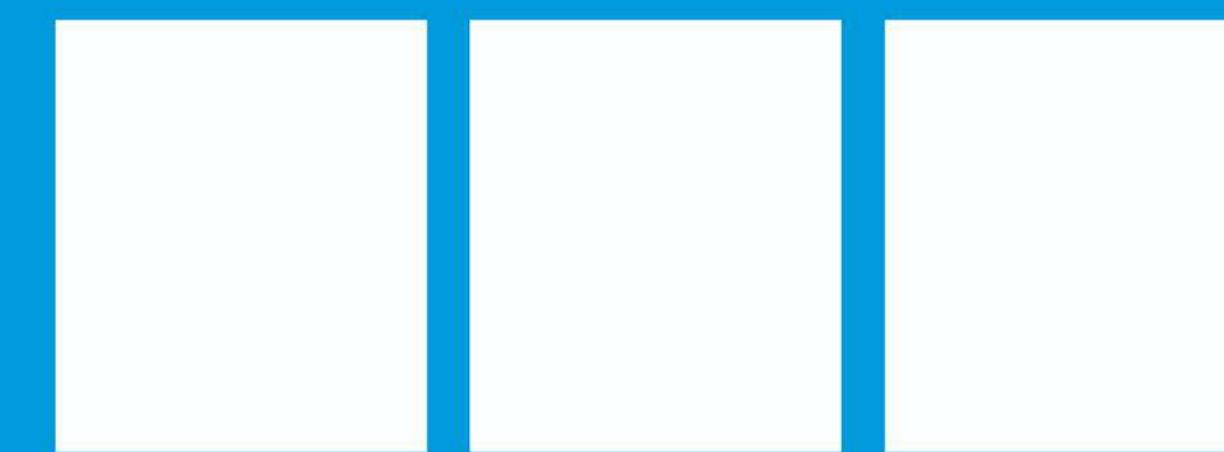
**justify-content**



**justify-content: flex-start;**  
**justify-content: flex-end;**  
**justify-content: center;**  
**justify-content: space-between;**  
**justify-content: space-around;**  
**justify-content: space-evenly;**

Propiedades CSS para el CONTENEDOR:

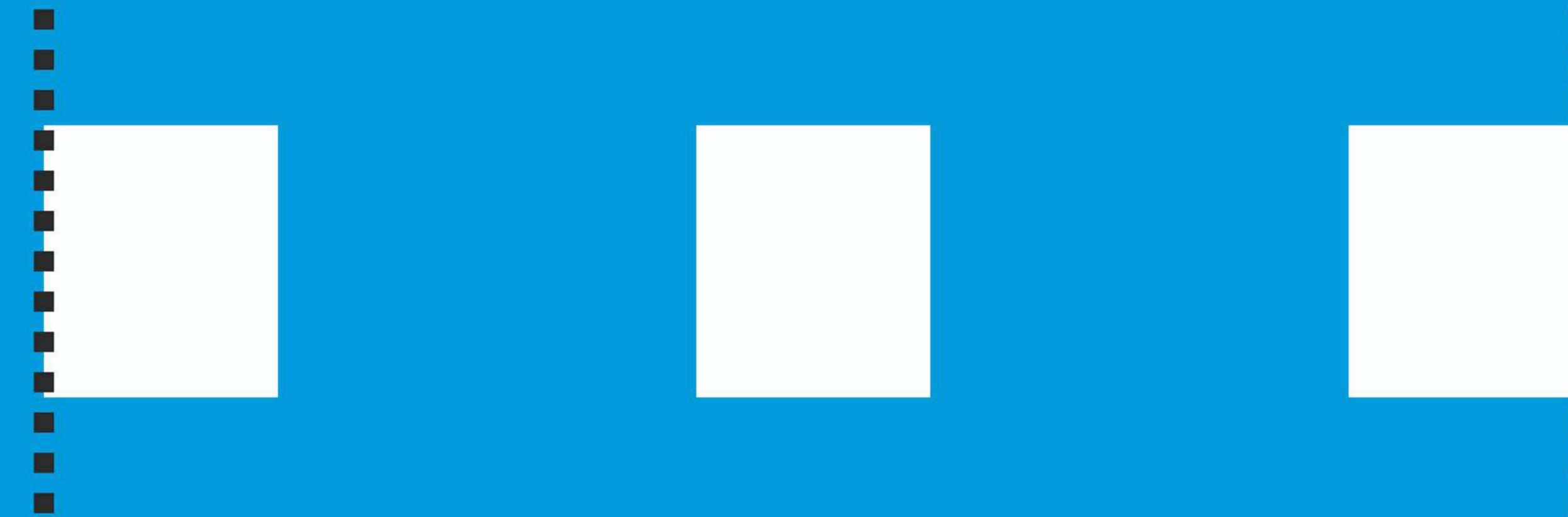
**justify-content**



**justify-content: flex-start;**  
**justify-content: flex-end;**  
**justify-content: center;**  
**justify-content: space-between;**  
**justify-content: space-around;**  
**justify-content: space-evenly;**

Propiedades CSS para el CONTENEDOR:

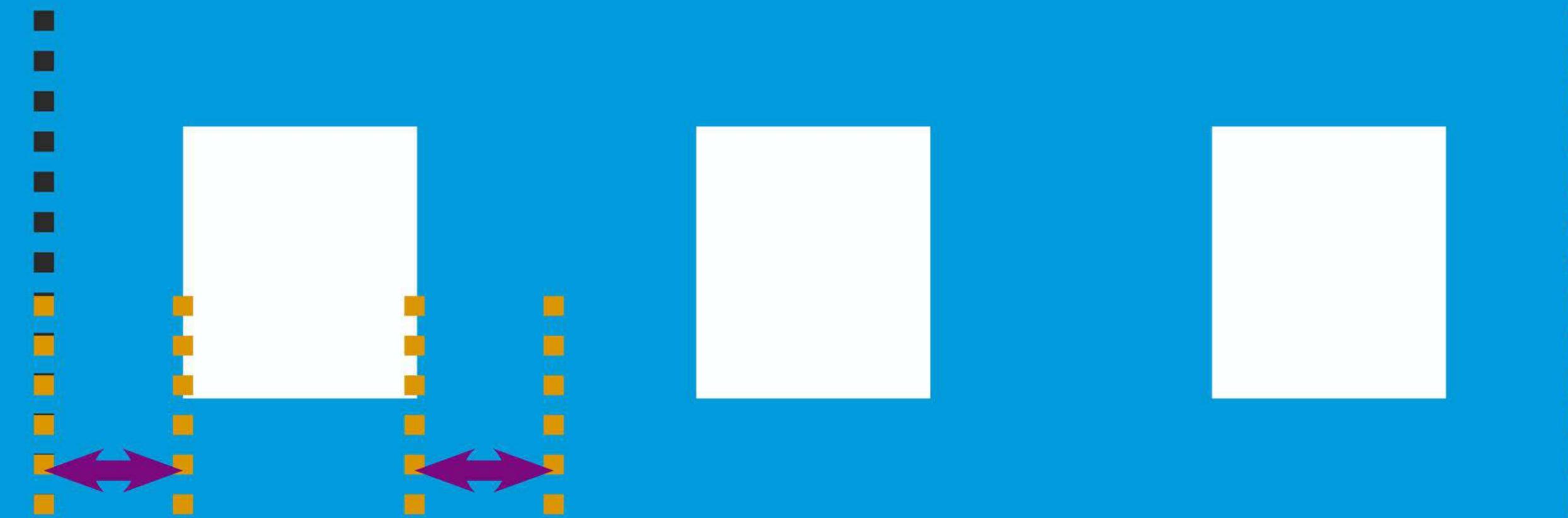
**justify-content**



**justify-content: flex-start;**  
**justify-content: flex-end;**  
**justify-content: center;**  
**justify-content: space-between;**  
**justify-content: space-around;**  
**justify-content: space-evenly;**

Propiedades CSS para el CONTENEDOR:

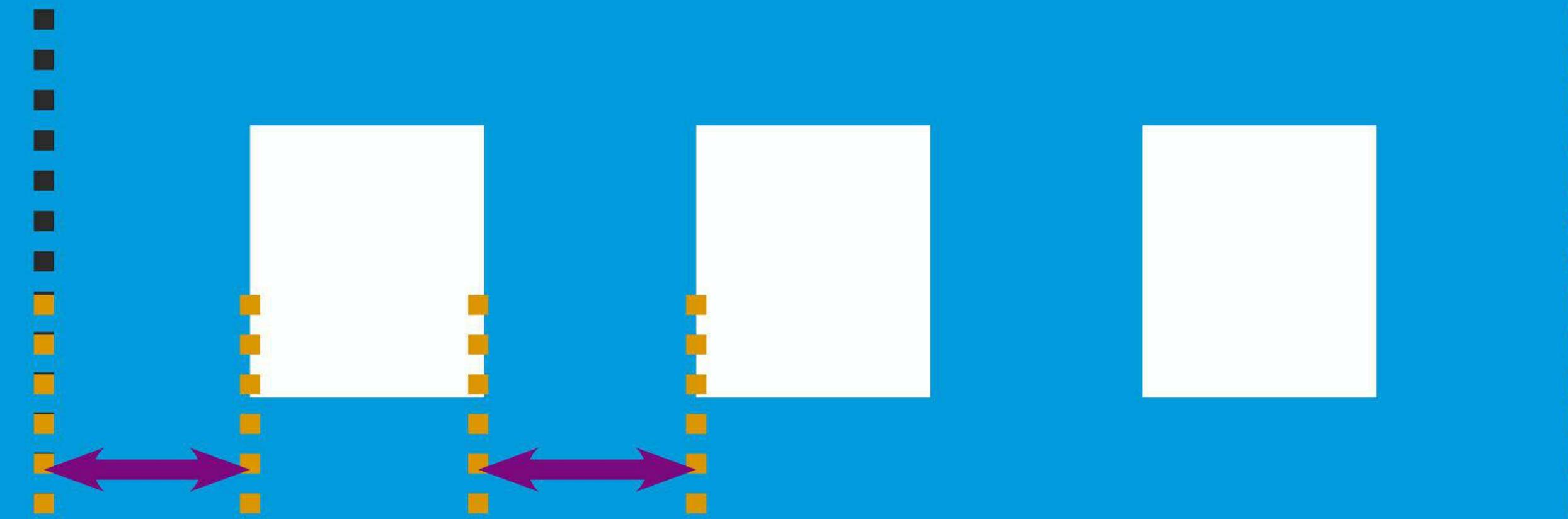
**justify-content**



**justify-content: flex-start;**  
**justify-content: flex-end;**  
**justify-content: center;**  
**justify-content: space-between;**  
**justify-content: space-around;**  
**justify-content: space-evenly;**

Propiedades CSS para el CONTENEDOR:

**justify-content**



`justify-content: flex-start;`  
`justify-content: flex-end;`  
`justify-content: center;`  
`justify-content: space-between;`  
`justify-content: space-around;`  
`justify-content: space-evenly;`

Propiedades CSS para el CONTENEDOR:

**align-items**



Especifica la distribución de los contenidos de  
cada línea según el eje secundario  
El valor predeterminado es **stretch**

Propiedades CSS para el CONTENEDOR:

**align-items**



**align-items: flex-start;**  
**align-items: flex-end;**  
**align-items: center;**  
**align-items: baseline;**  
**align-items: stretch;**

Propiedades CSS para el CONTENEDOR:

**align-items**



**align-items: flex-start;**

**align-items: flex-end;**

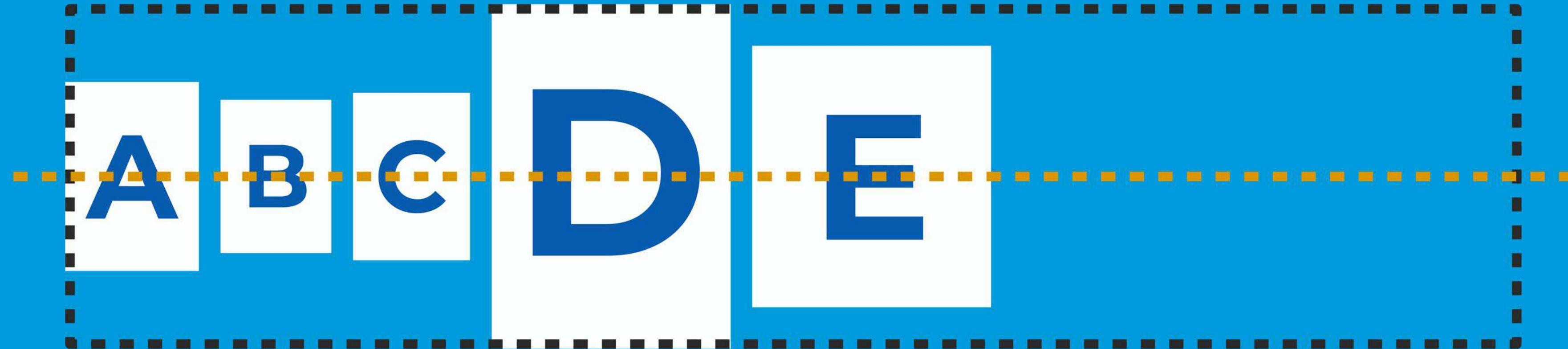
**align-items: center;**

**align-items: baseline;**

**align-items: stretch;**

Propiedades CSS para el CONTENEDOR:

**align-items**



**align-items: flex-start;**

**align-items: flex-end;**

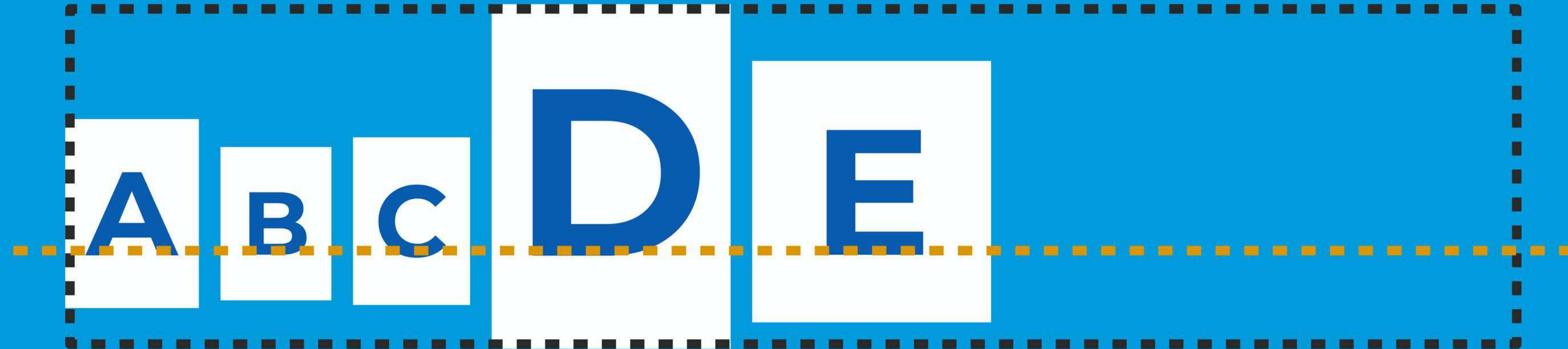
**align-items: center;**

**align-items: baseline;**

**align-items: stretch;**

Propiedades CSS para el CONTENEDOR:

**align-items**



**align-items: flex-start;**

**align-items: flex-end;**

**align-items: center;**

**align-items: baseline;**

**align-items: stretch;**

Propiedades CSS para el CONTENEDOR:

**align-items**



**align-items: flex-start;**

**align-items: flex-end;**

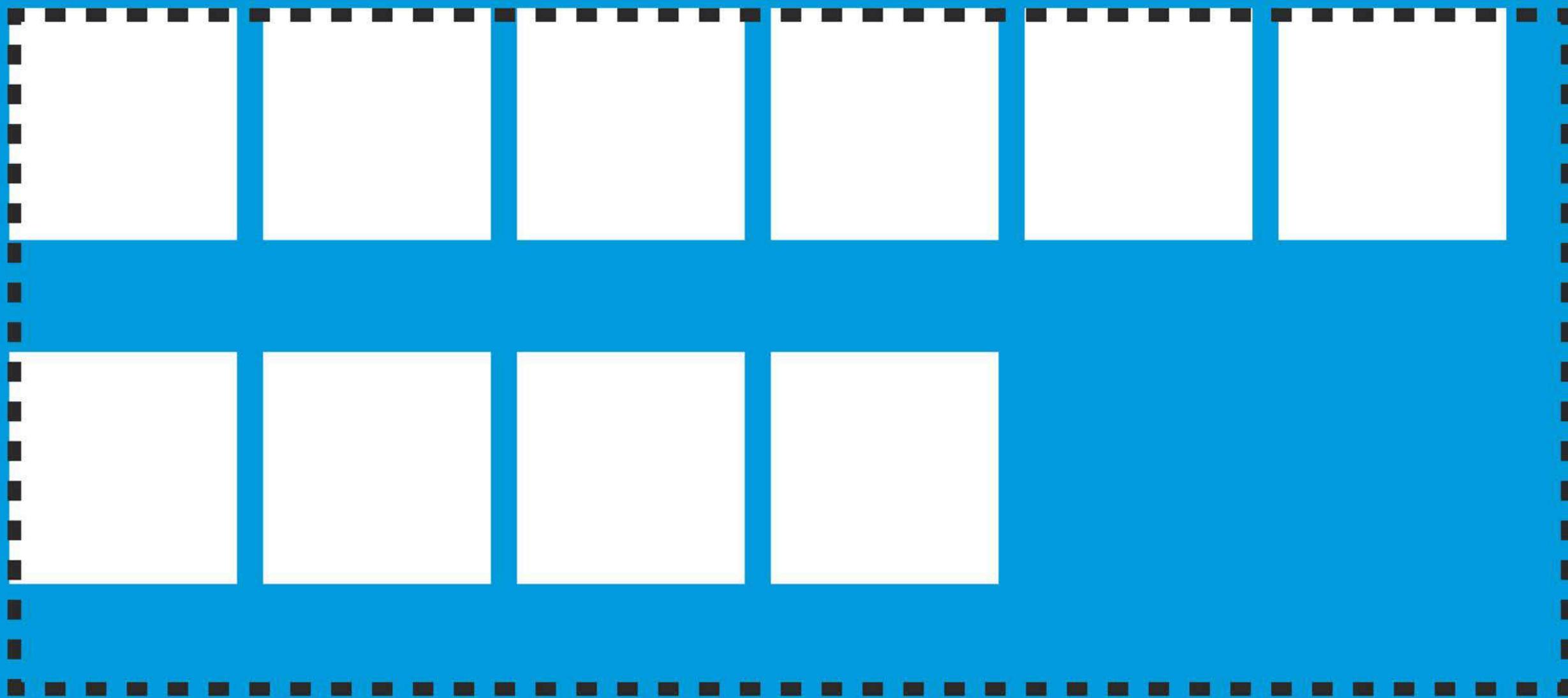
**align-items: center;**

**align-items: baseline;**

**align-items: stretch;**

Propiedades CSS para el CONTENEDOR:

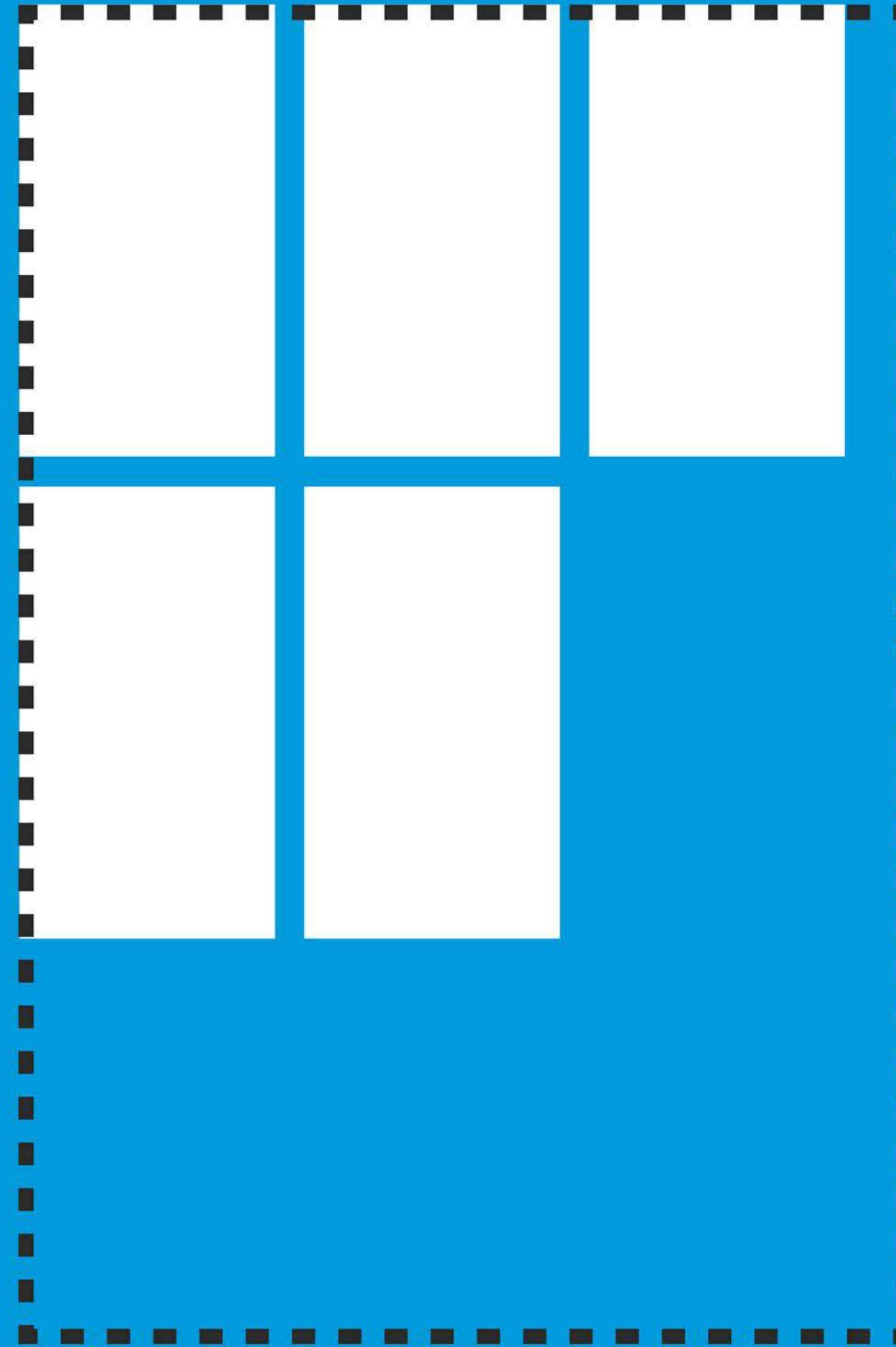
**align-content**



En contenedores con más de una línea de ítems, especifica la distribución de todos los contenidos según el eje secundario  
El valor predeterminado es **stretch**

Propiedades CSS para el CONTENEDOR:

## align-content



**align-content: flex-start;**

**align-content: flex-end;**

**align-content: center;**

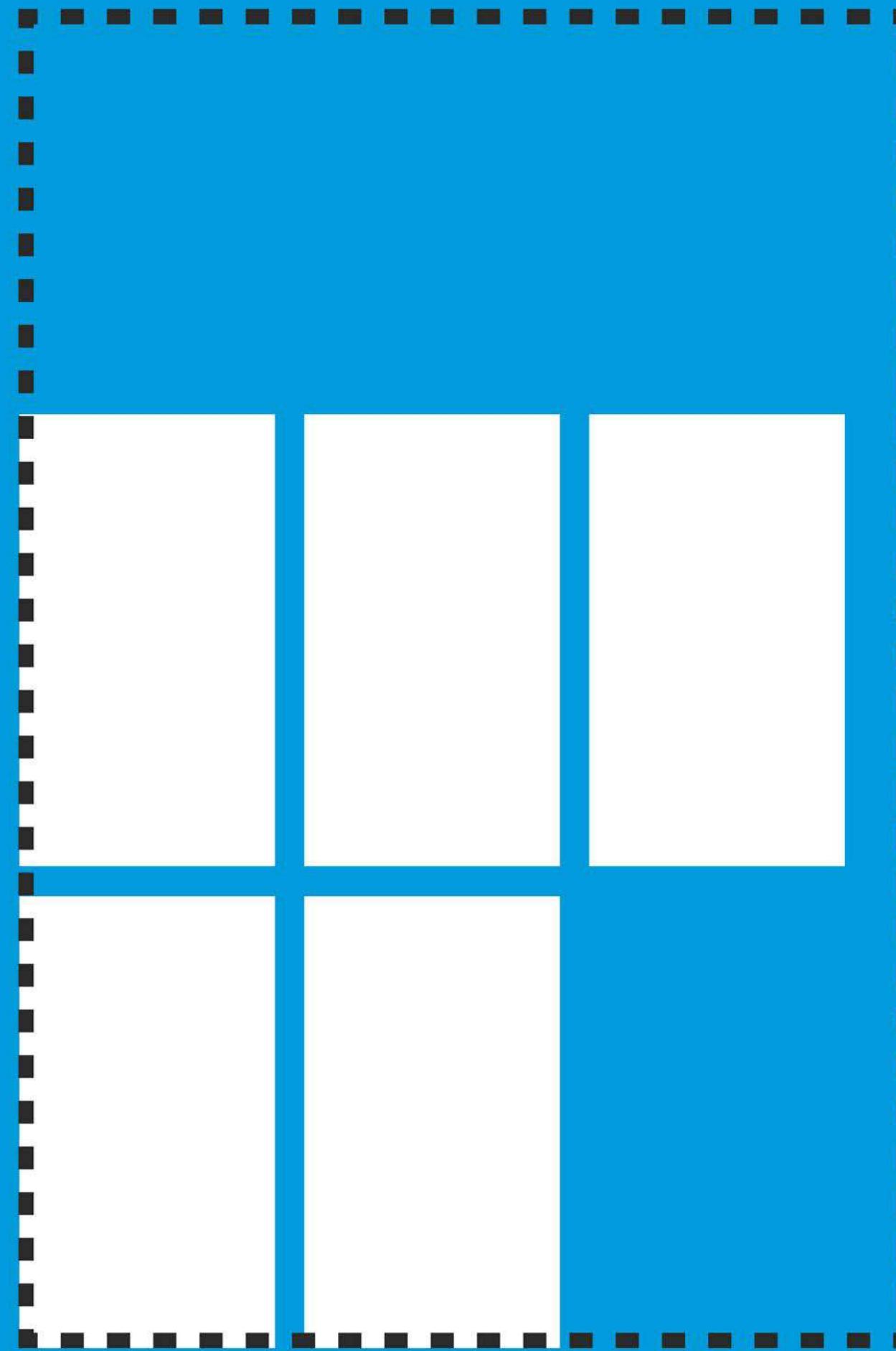
**align-content: stretch;**

**align-content: space-between;**

**align-content: space-around;**

Propiedades CSS para el CONTENEDOR:

**align-content**



`align-content: flex-start;`

`align-content: flex-end;`

`align-content: center;`

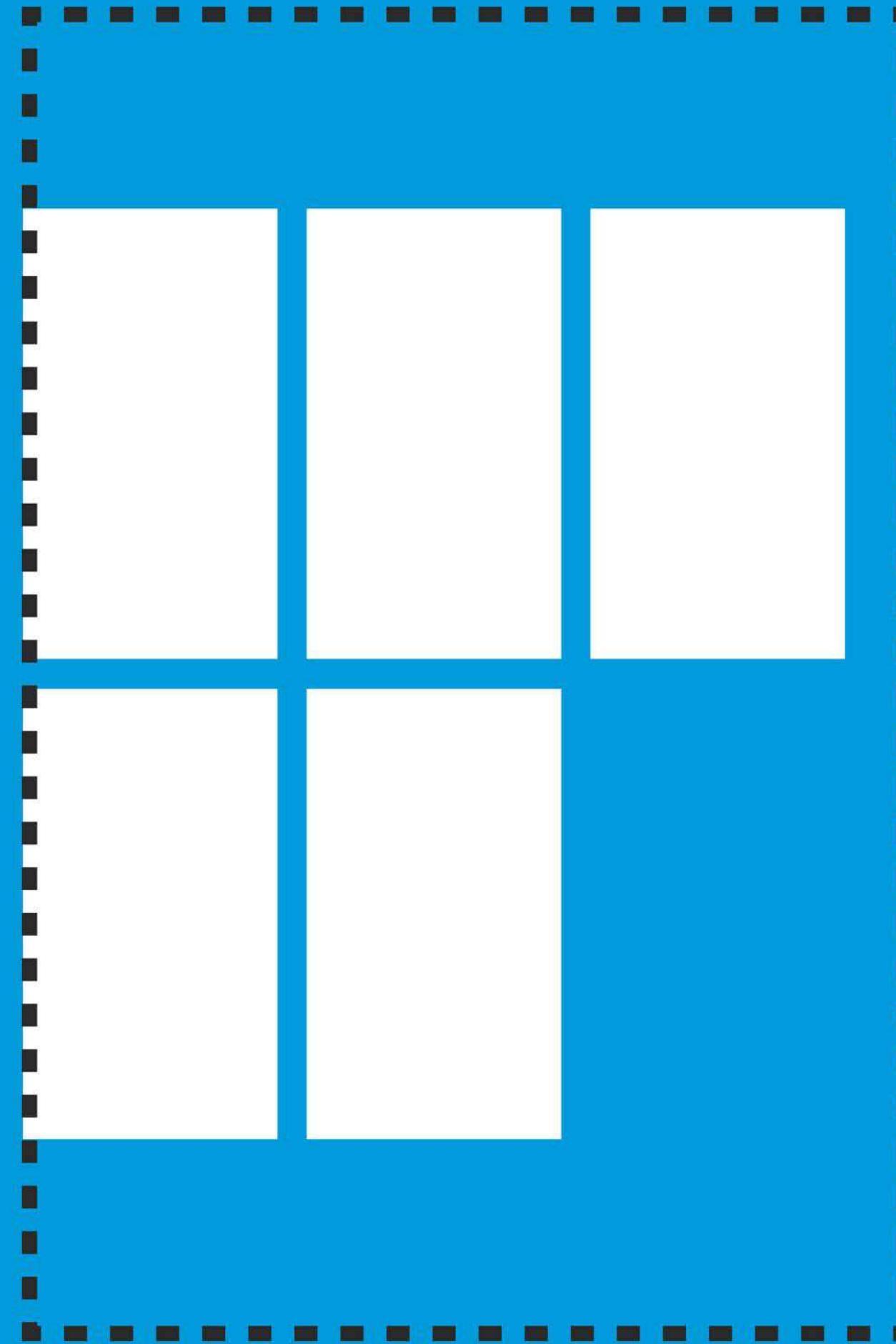
`align-content: stretch;`

`align-content: space-between;`

`align-content: space-around;`

Propiedades CSS para el CONTENEDOR:

## align-content



**align-content: flex-start;**

**align-content: flex-end;**

**align-content: center;**

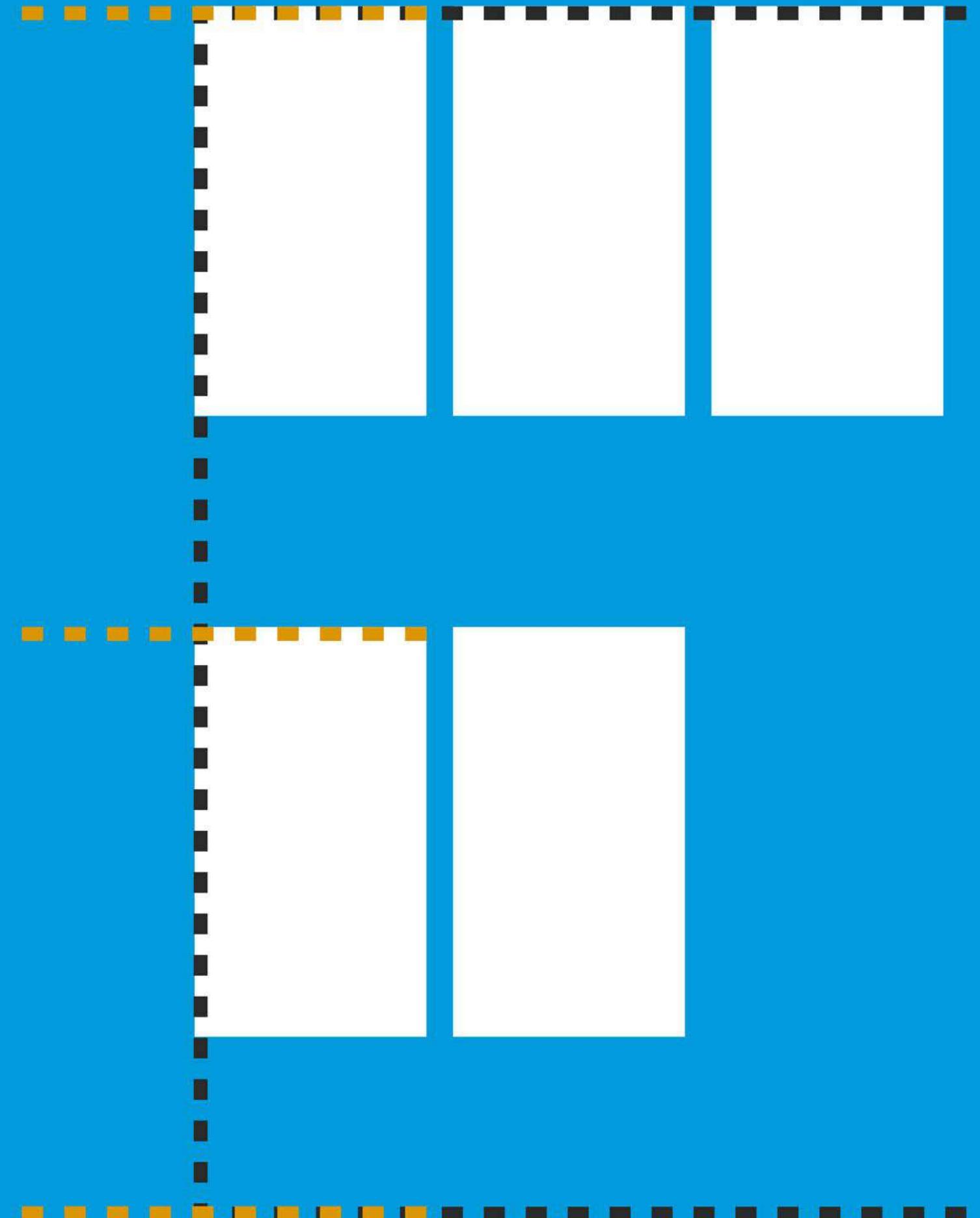
**align-content: stretch;**

**align-content: space-between;**

**align-content: space-around;**

Propiedades CSS para el CONTENEDOR:

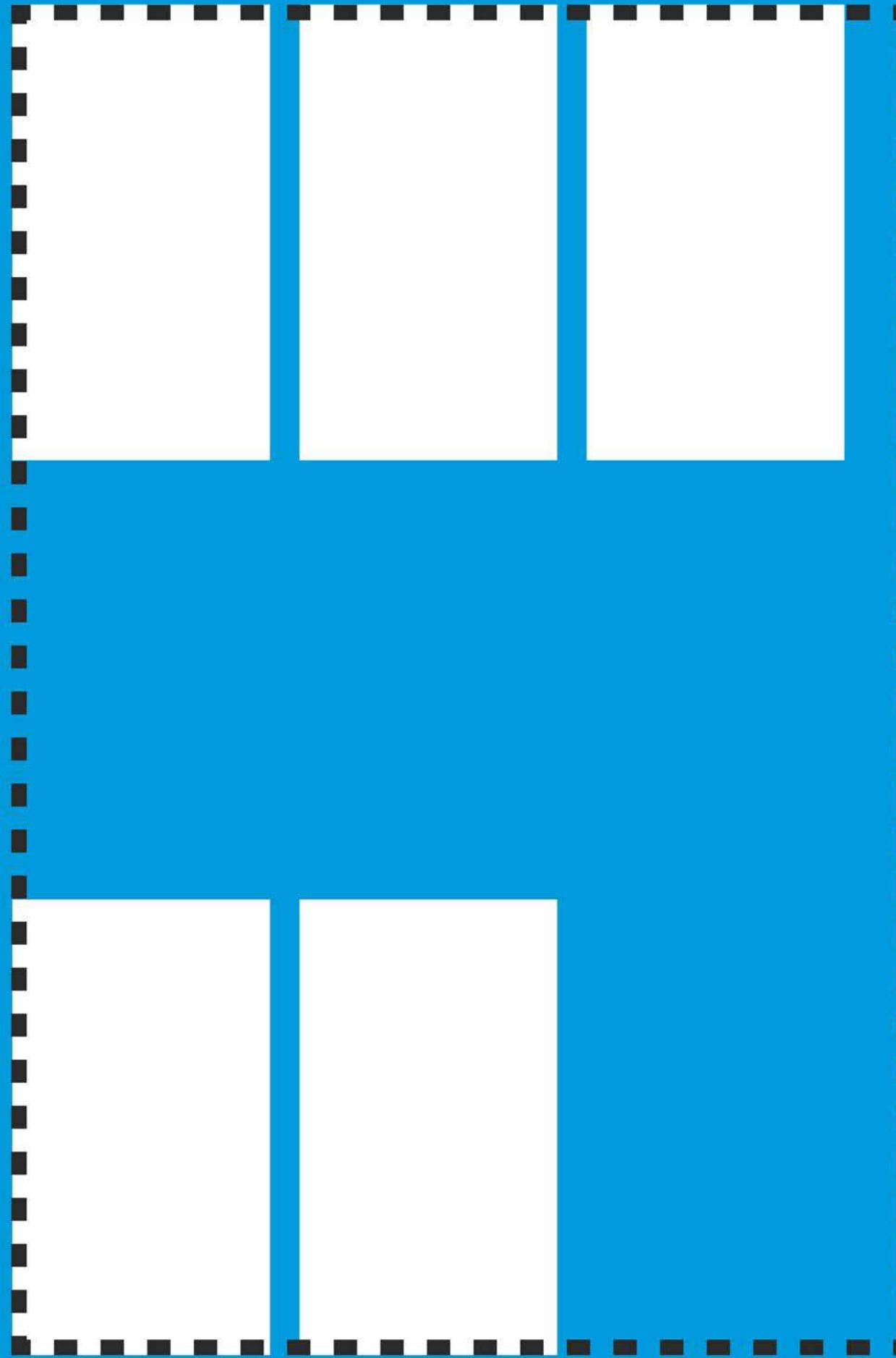
## align-content



`align-content: flex-start;`  
`align-content: flex-end;`  
`align-content: center;`  
`align-content: stretch;`  
`align-content: space-between;`  
`align-content: space-around;`

Propiedades CSS para el CONTENEDOR:

**align-content**



`align-content: flex-start;`

`align-content: flex-end;`

`align-content: center;`

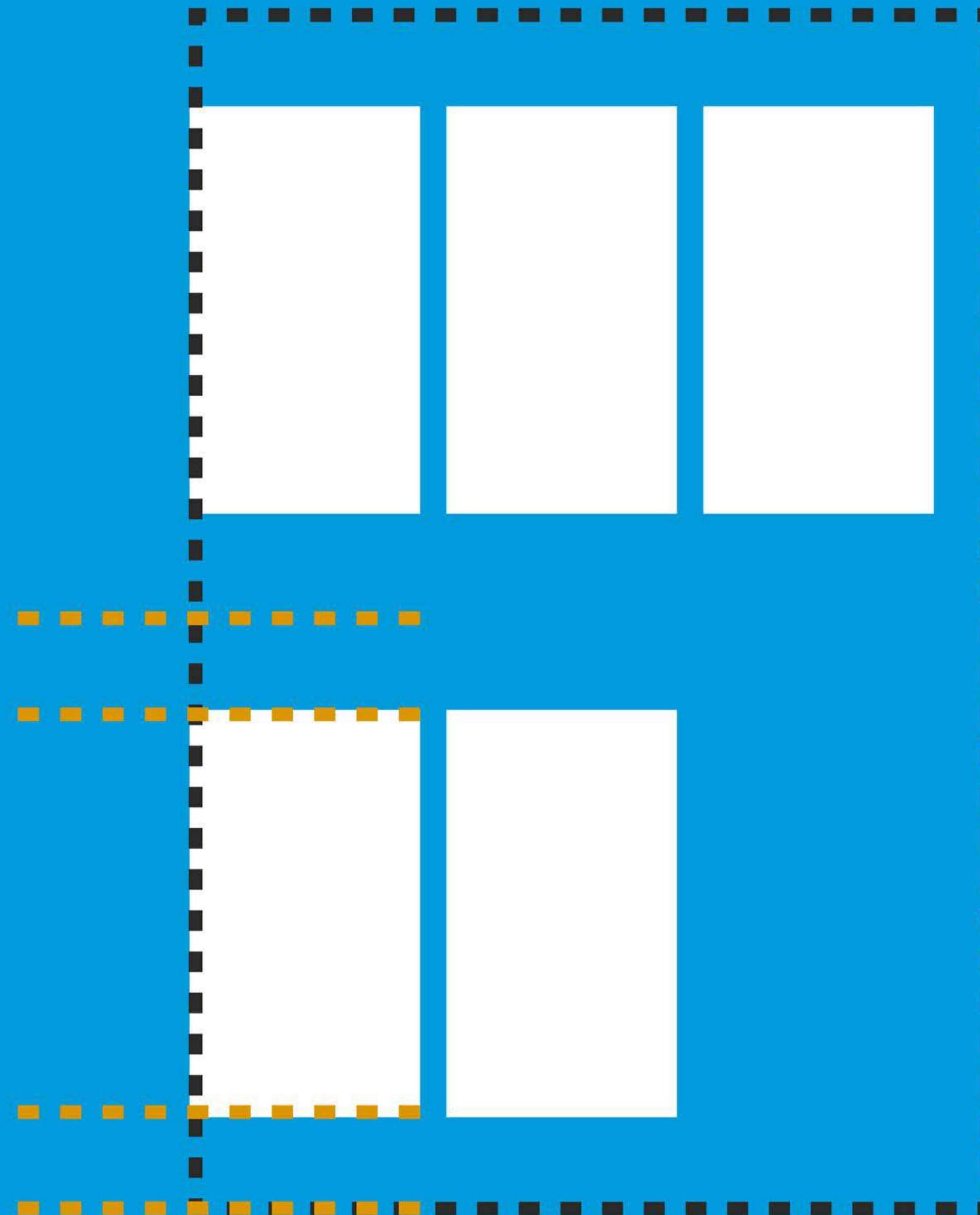
`align-content: stretch;`

`align-content: space-between;`

`align-content: space-around;`

Propiedades CSS para el CONTENEDOR:

## align-content



`align-content: flex-start;`

`align-content: flex-end;`

`align-content: center;`

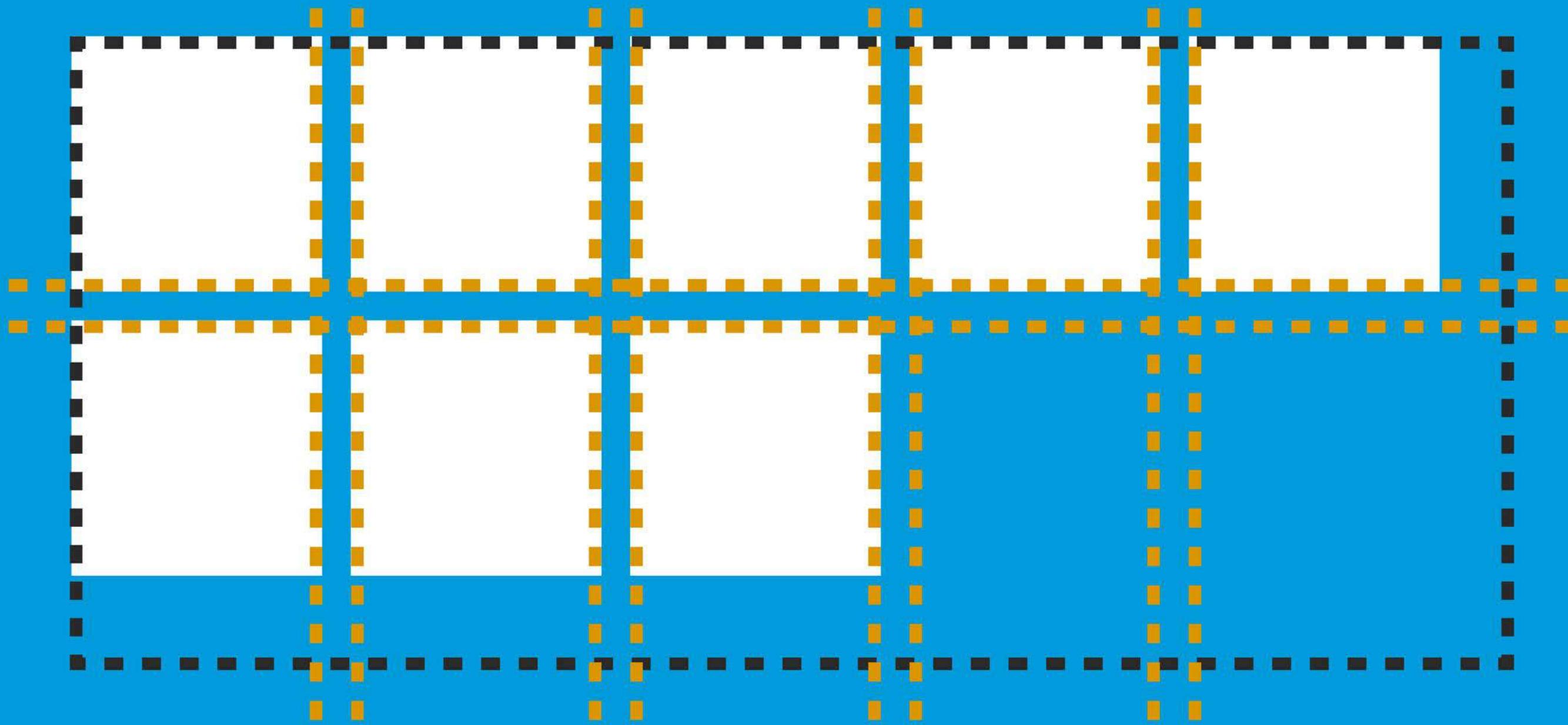
`align-content: stretch;`

`align-content: space-between;`

`align-content: space-around;`

## Propiedades CSS para el CONTENEDOR:

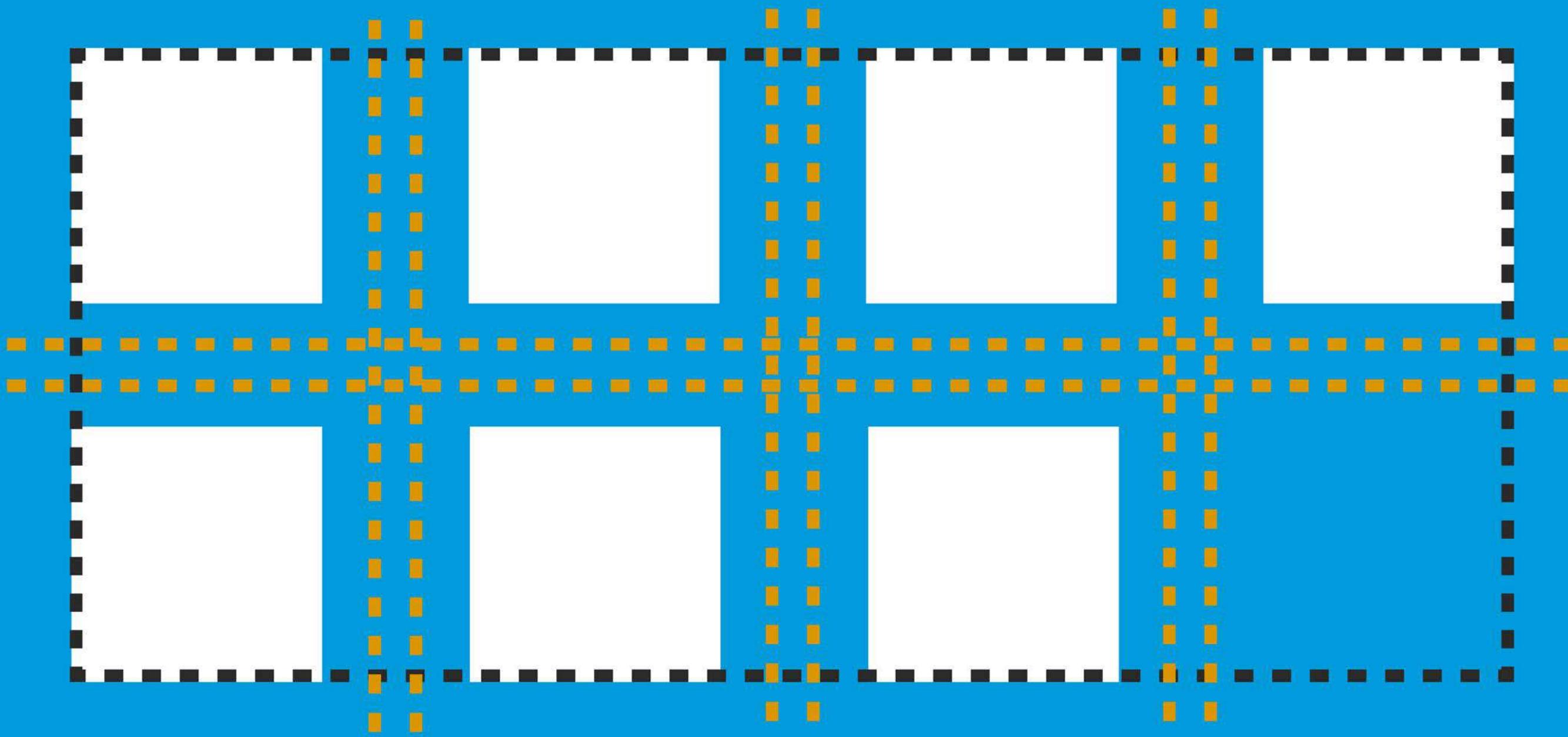
gap



Controla específicamente el **espacio** entre los elementos de la caja flexible  
Se aplica sólo entre elementos, no en los bordes exteriores  
La propiedad **gap** resume las propiedades **row-gap** y **column-gap**

## Propiedades CSS para el CONTENEDOR:

gap



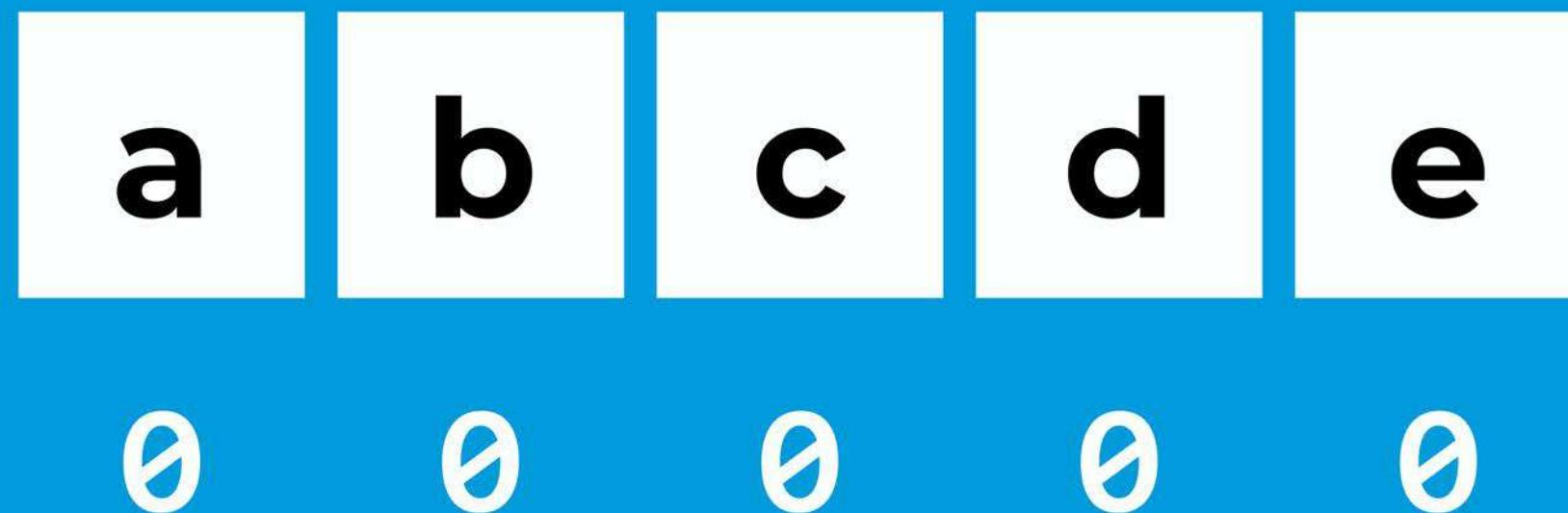
gap podría pensarse como una calle mínima:  
si la calle fuera más grande de lo indicado en la propiedad,  
puede ser que se haya aplicado justify-content: space-between; por ejemplo

# Propiedades CSS para los ÍTEMS/ELEMENTOS CONTENIDOS

- `order`
- `flex-grow`
- `flex-shink`
- `flex-basis`
- `align-self`

## Propiedades CSS para los CONTENIDOS:

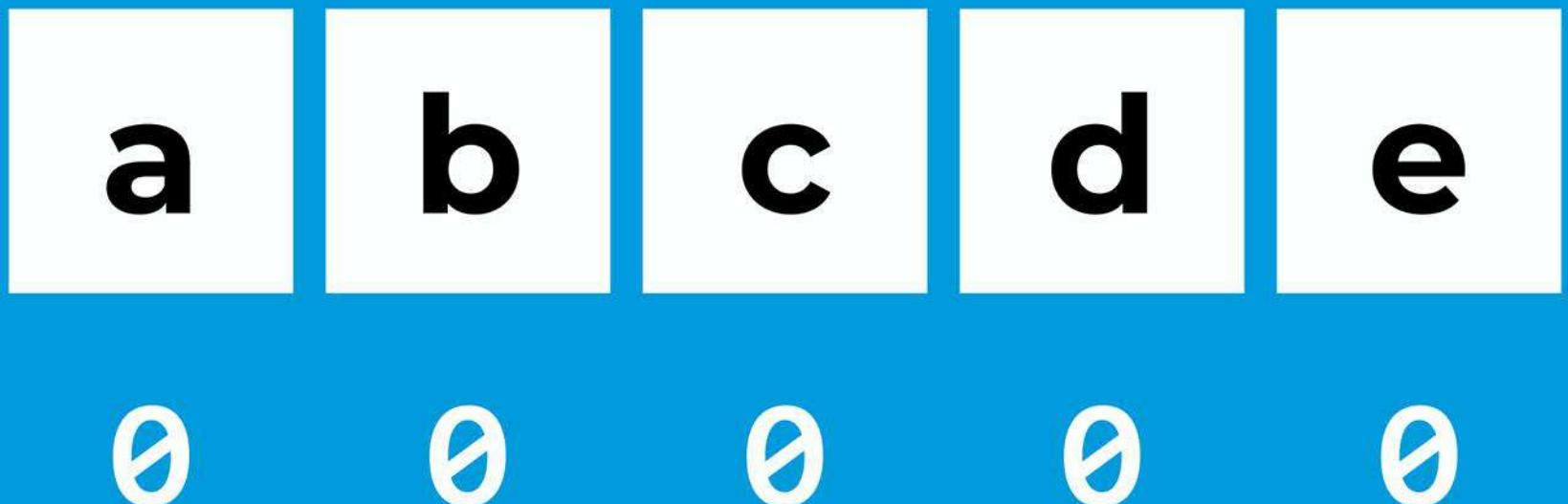
order



Especifica el **orden** de los ítems usando números enteros  
El valor predeterminado es **0 (cero)**  
y puede cambiarse a un valor **positivo o negativo**

## Propiedades CSS para los CONTENIDOS:

order

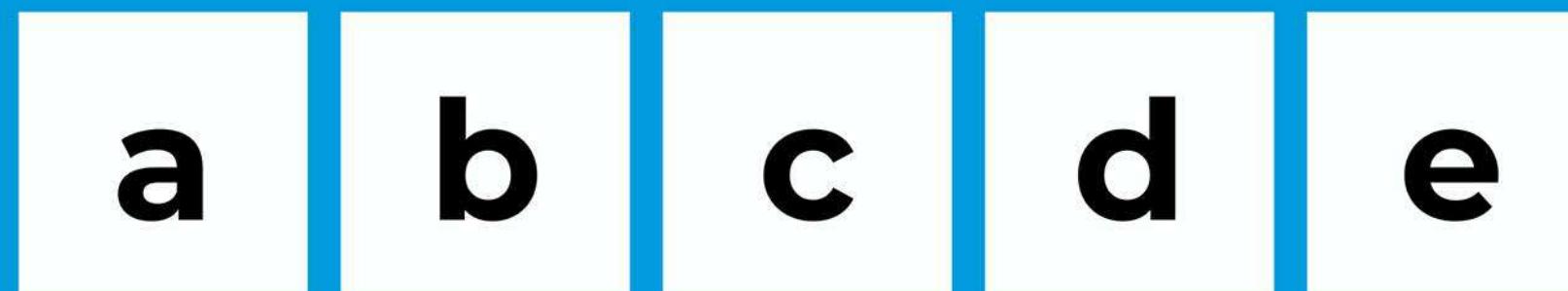


```
.b { order: 1; }
```

```
.e { order: 2; }
```

## Propiedades CSS para los CONTENIDOS:

order



0 0 0 0 0



0 0 0 1 2



0 2 3 29 853



-1 0 2 29 853

## Propiedades CSS para los CONTENIDOS:

**flex-grow**



Indica el factor de crecimiento de los ítems, en caso de que no tengan un ancho específico. Este número de cada ítem mantiene relación con los números definidos para los demás ítems. El valor predeterminado es 0 (cero)

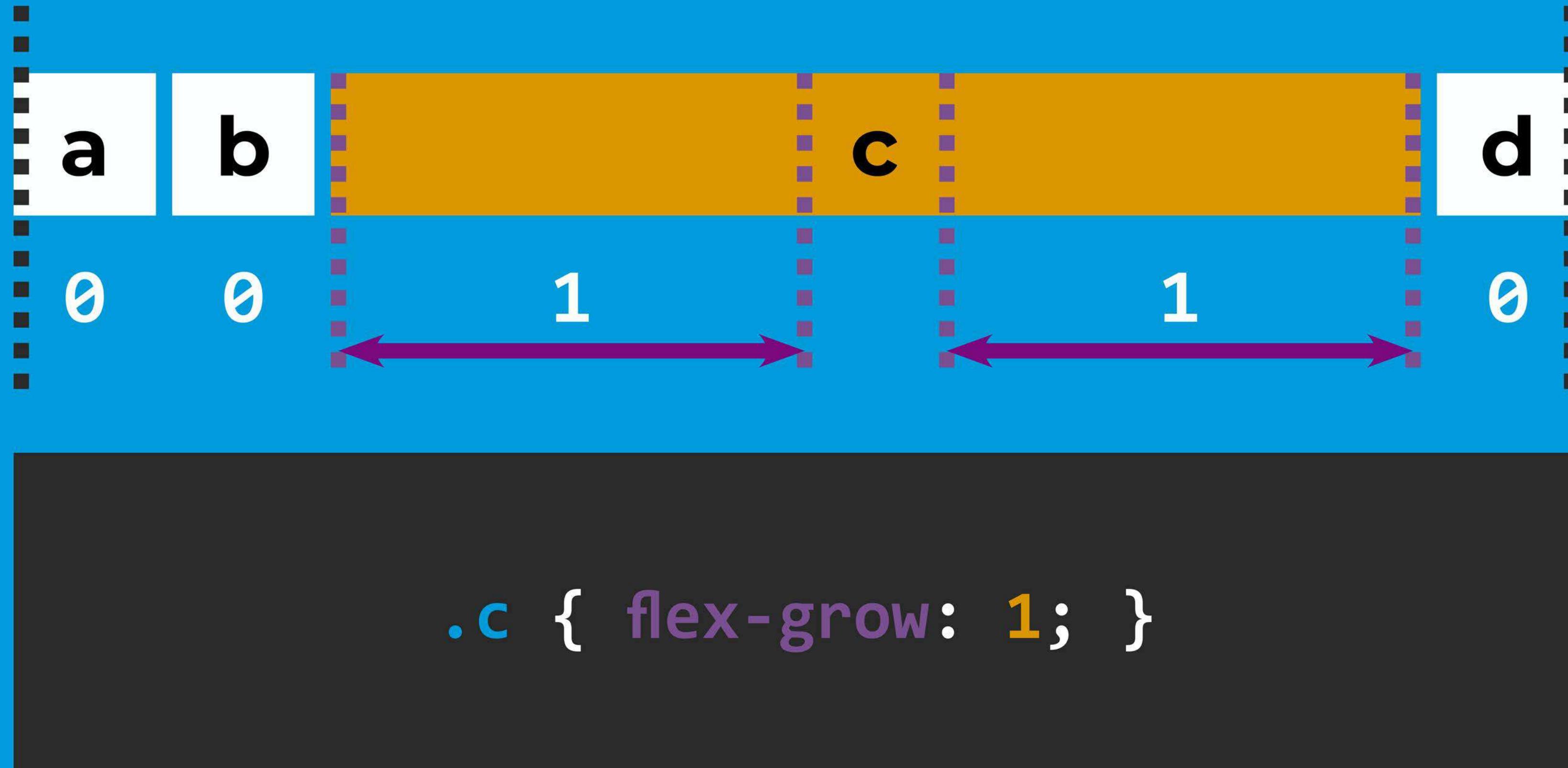
# Propiedades CSS para los CONTENIDOS:

**flex-grow**



# Propiedades CSS para los CONTENIDOS:

**flex-grow**



# Propiedades CSS para los CONTENIDOS:

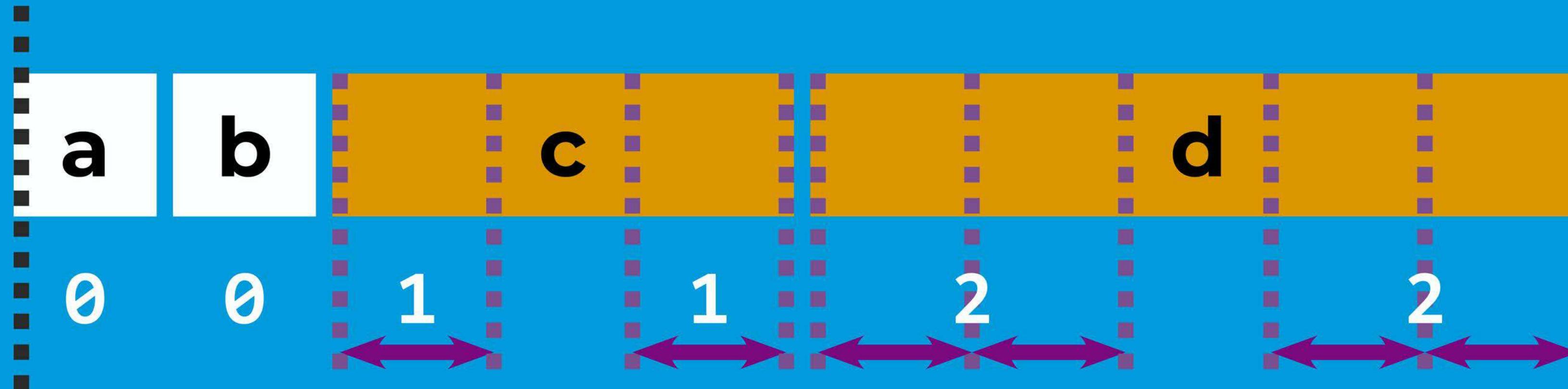
**flex-grow**



```
.c { flex-grow: 1; }  
.d { flex-grow: 2; }
```

# Propiedades CSS para los CONTENIDOS:

**flex-grow**



```
.c { flex-grow: 1; }  
.d { flex-grow: 2; }
```

Propiedades CSS para los CONTENIDOS:

**flex-shrink**

Ouesto a **flex-grow**, indica el factor de “achicamiento” de los ítems  
El valor predeterminado es **1**

## Propiedades CSS para los CONTENIDOS:

**flex-basis**



Define el tamaño predeterminado (de base) según el eje principal que tienen los ítems antes de aplicarle la distribución de espacio. El valor predeterminado es **content** (mide lo que el contenido del ítem)

## Propiedades CSS para los CONTENIDOS:

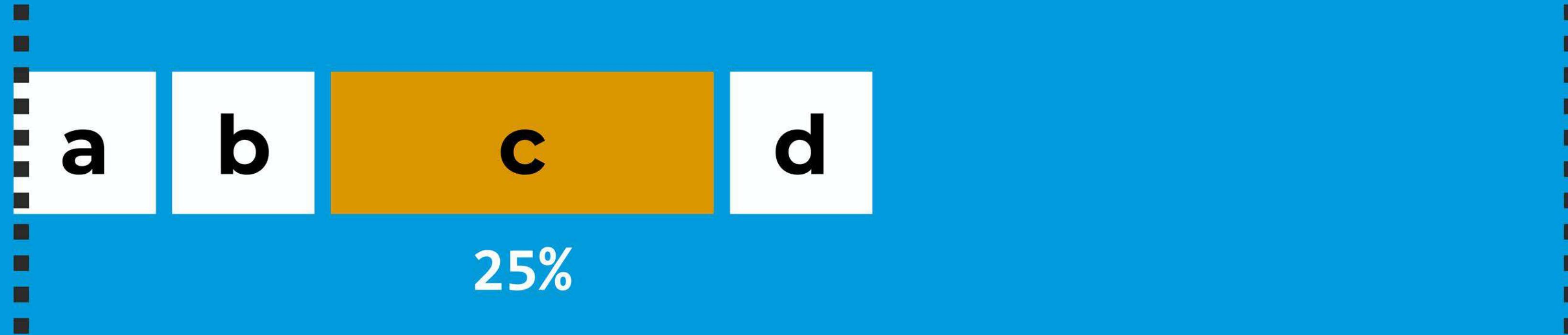
**flex-basis**



```
.c { flex-basis: content; }
```

## Propiedades CSS para los CONTENIDOS:

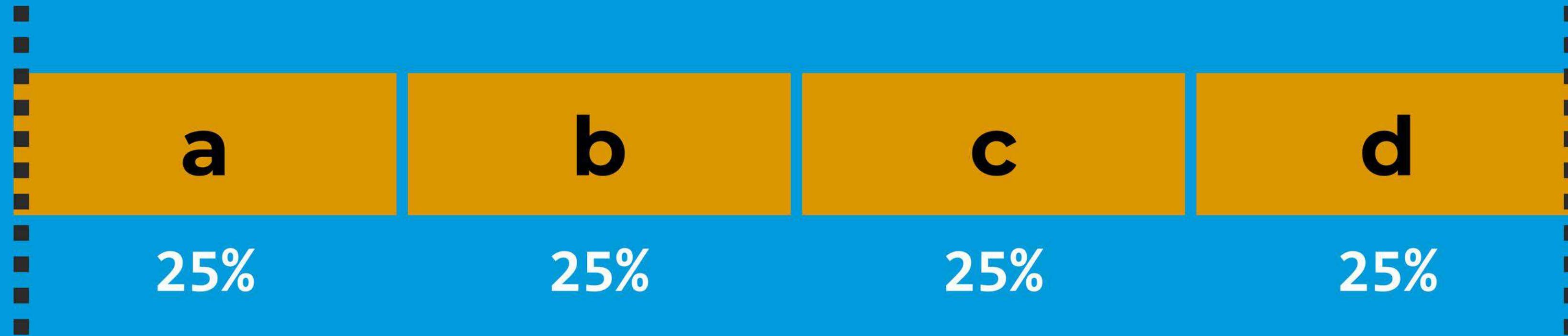
**flex-basis**



```
.c { flex-basis: 25%; }
```

## Propiedades CSS para los CONTENIDOS:

**flex-basis**



```
.a, .b, .c, .d { flex-basis: 25%; }
```

## Propiedades CSS para los CONTENIDOS:

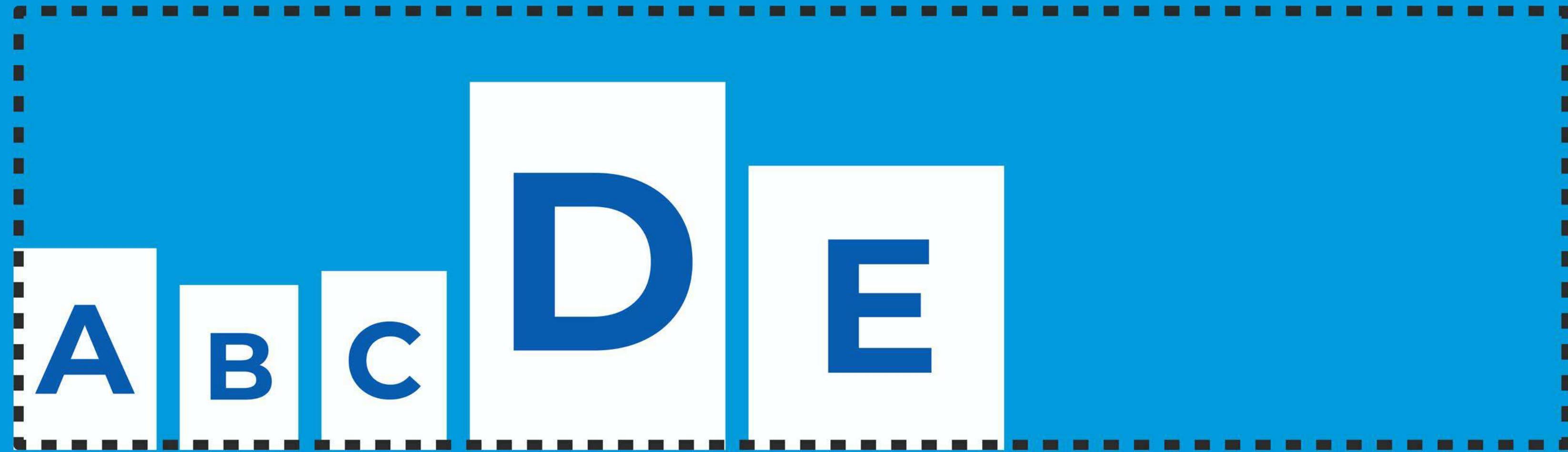
**align-self**



Actúa exactamente igual que `align-items`,  
pero lo hace específicamente para **cada ítem**  
y no de manera global para todo el contenido

Propiedades CSS para los CONTENIDOS:

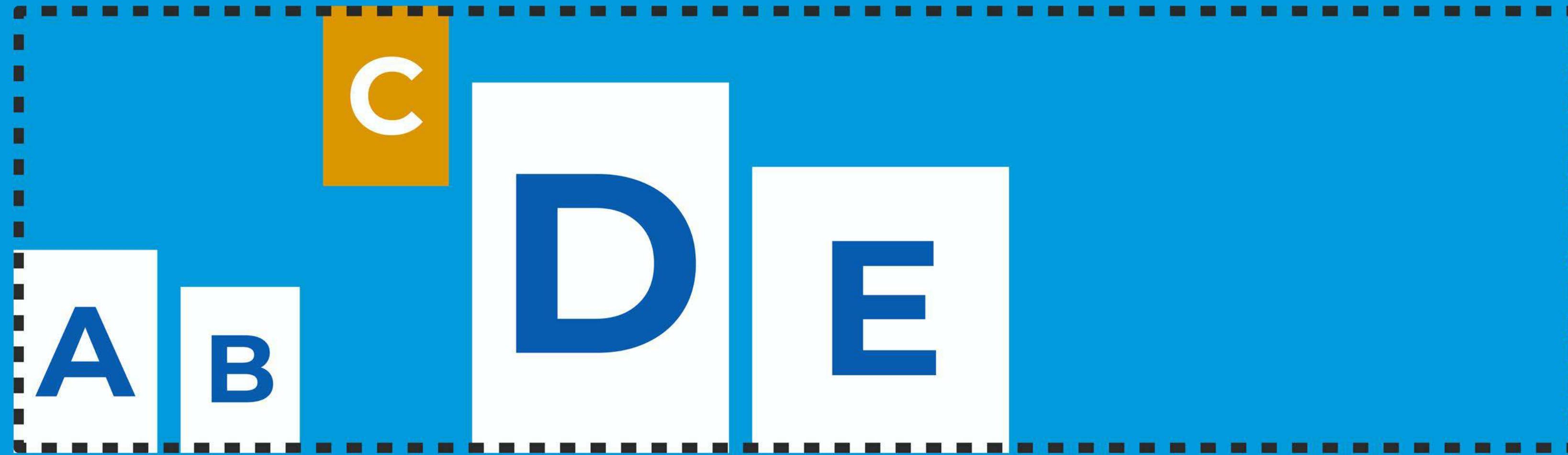
align-self



```
.contenedor { align-items: flex-end; }
```

Propiedades CSS para los CONTENIDOS:

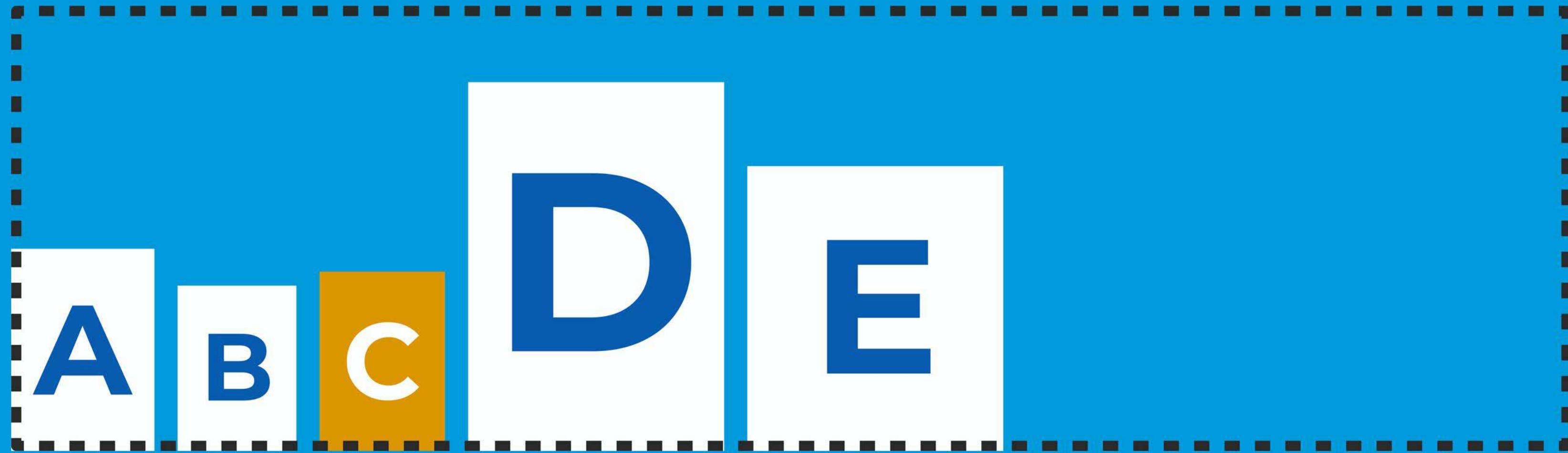
**align-self**



```
.contenedor { align-items: flex-end; }  
.c { align-self: flex-start; }
```

Propiedades CSS para los CONTENIDOS:

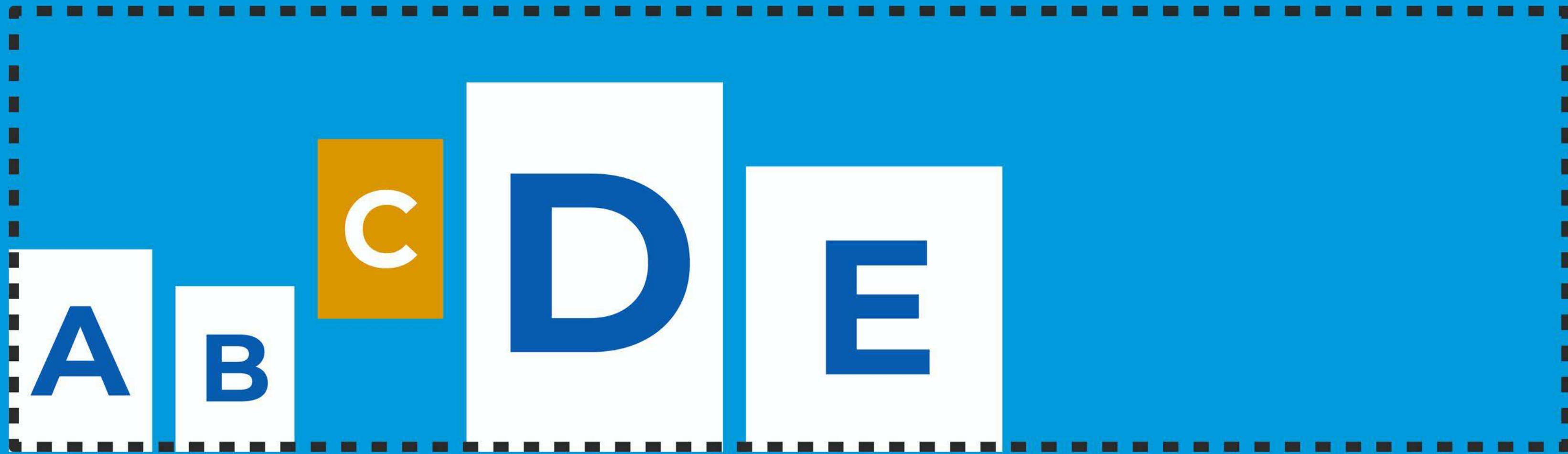
**align-self**



```
.contenedor { align-items: flex-end; }  
.c { align-self: flex-end; }
```

Propiedades CSS para los CONTENIDOS:

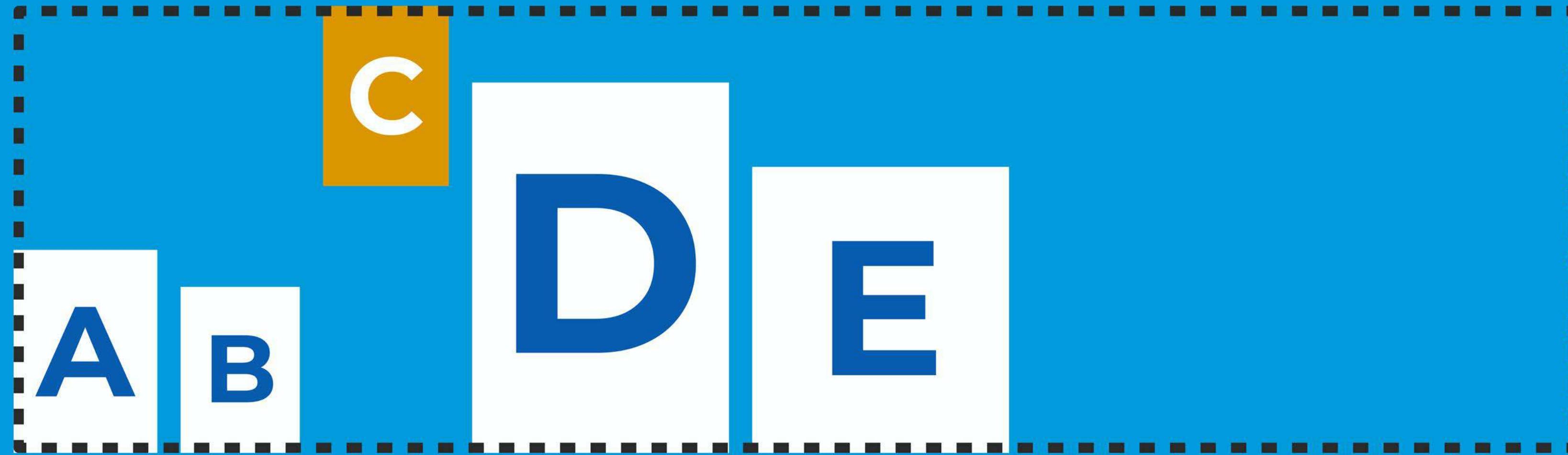
align-self



```
.contenedor { align-items: flex-end; }  
.c { align-self: center; }
```

## Propiedades CSS para los CONTENIDOS:

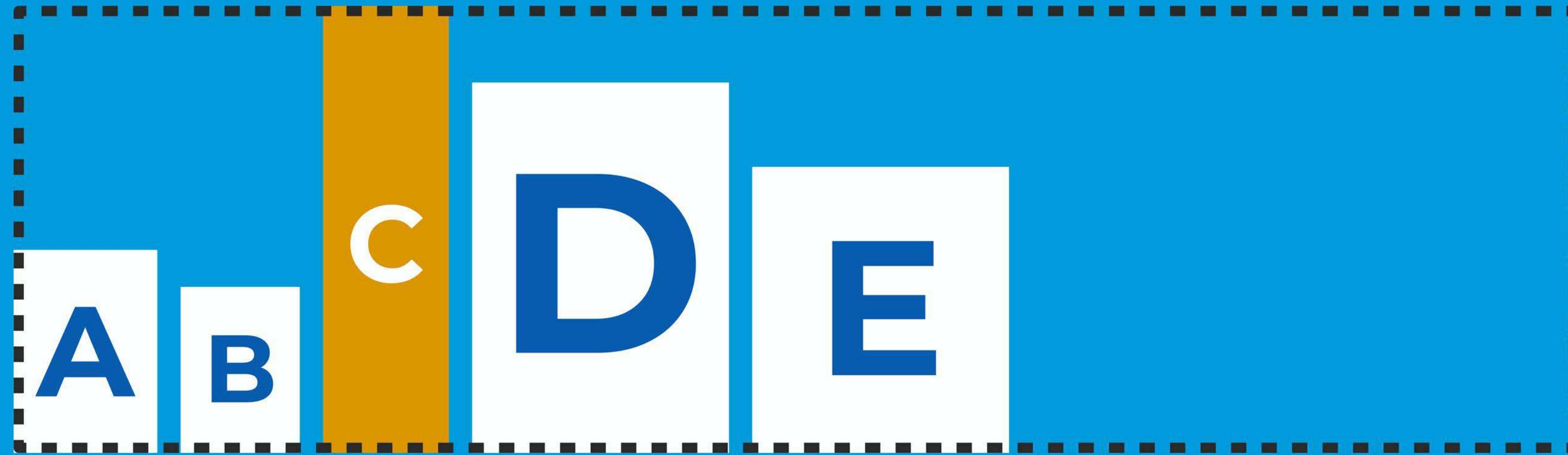
**align-self**



```
.contenedor { align-items: flex-end; }  
.c { align-self: baseline; }
```

Propiedades CSS para los CONTENIDOS:

align-self



```
.contenedor { align-items: flex-end; }  
.c { align-self: stretch; }
```