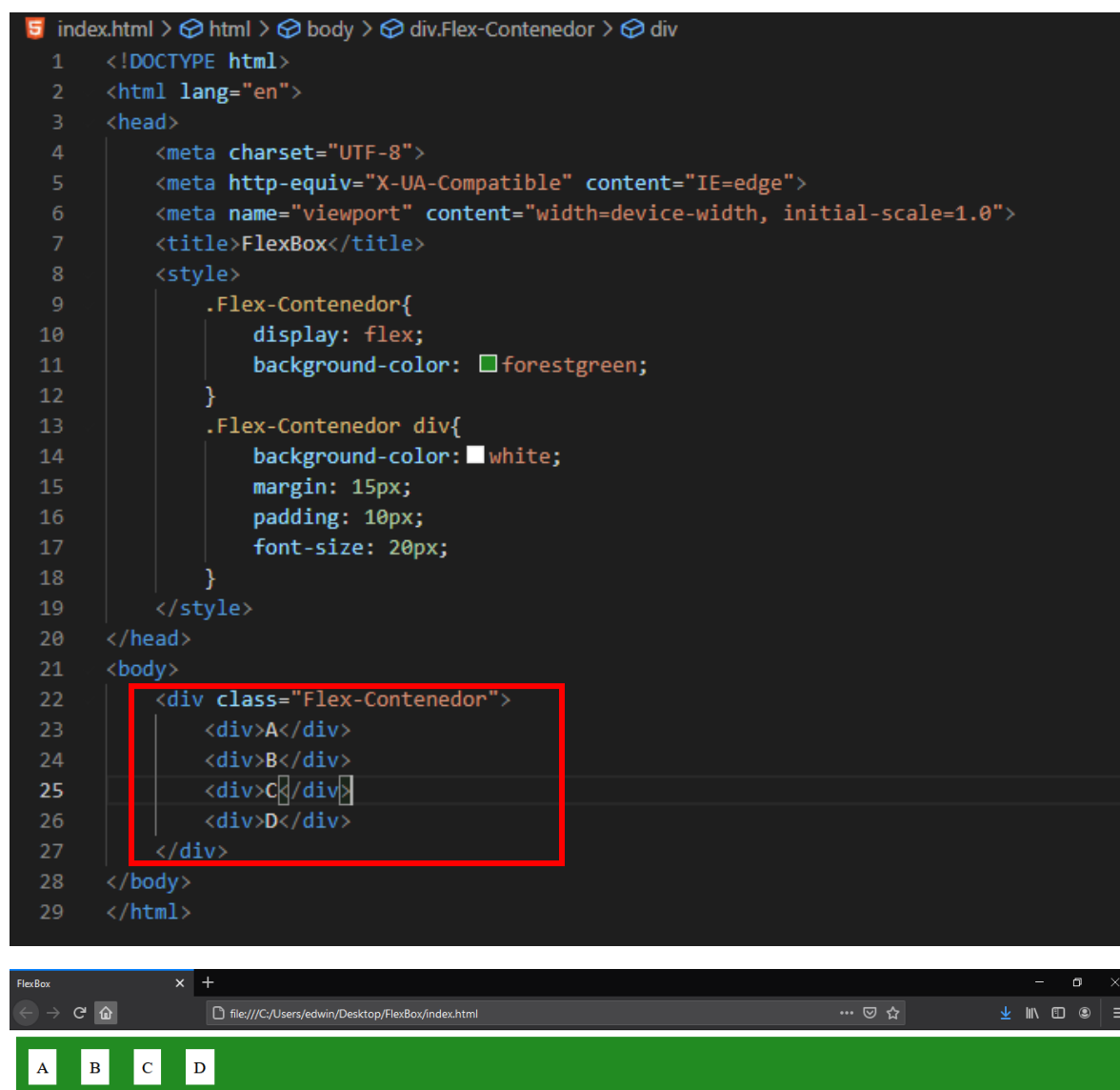


## CSS Flexbox

El Módulo de Caja Flexible, llamado flexbox, se diseñó como un modelo unidimensional de layout, y como un método que pueda ayudar a distribuir el espacio entre los ítems de una interfaz y mejorar las capacidades de alineación.

En la utilización de un Flexbox se debe definir inicialmente un contenedor flexible.

En La siguiente figura definimos un contenedor de 4 elementos



## Contenedor Flexbox

Para que el contenedor sea flexible se establece en el style la propiedad display flex.

```
.Flex-Contenedor{  
    display: flex;
```

## Propiedades del Flexbox

### 1. Dirección Flexible **flex-direction**

Flexbox posee cuatro valores posibles

- El **row** apila los elementos flex horizontalmente (de izquierda a derecha)

```
.Flex-Contenedor{  
    display: flex;  
    flex-direction: row;
```

Diagrama de flex-direction: row. Una barra horizontal verde contiene cuatro cuadros blancos con las letras A, B, C y D en orden de izquierda a derecha.

- El **row-reverse** apila los elementos flex horizontalmente (de derecha a izquierda):

```
.Flex-Contenedor{  
    display: flex;  
    flex-direction: row-reverse;
```

Diagrama de flex-direction: row-reverse. Una barra horizontal verde contiene cuatro cuadros blancos con las letras D, C, B y A en orden de izquierda a derecha.

- El **column** apila los elementos flex verticalmente (de arriba a abajo):

```
.Flex-Contenedor{  
    display: flex;  
    flex-direction: column;
```

Diagrama de flex-direction: column. Una barra vertical verde contiene cuatro cuadros blancos con las letras A, B, C y D en orden de arriba a abajo.

- El **column-reverse** apila los elementos flex verticalmente (de abajo hacia arriba):

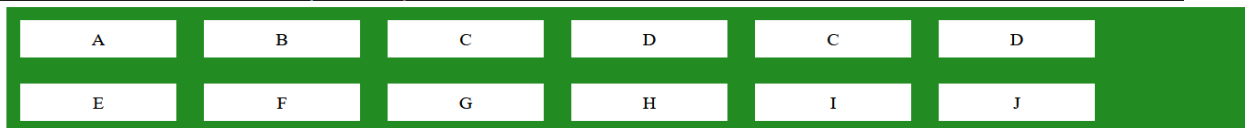


## 2. Envoltura Flexible **flex-wrap**

Esta propiedad tiene 3 posibles valores

- **wrap** nos permite que cuando los ítems son demasiados para desplegarlos en una línea, serán repartidos en la línea siguiente.

```
.Flex-Contenedor{  
  display: flex;  
  flex-wrap: wrap;
```



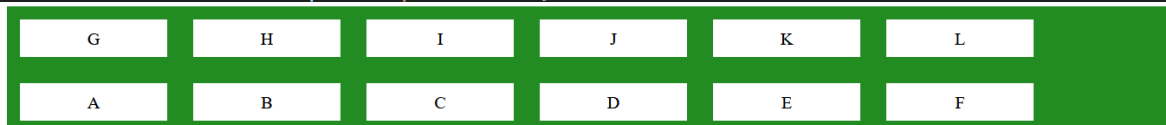
- **nowrap** indica que los elementos flexibles no se ajustarán (esto es predeterminado)

```
.Flex-Contenedor{  
  display: flex;  
  flex-wrap: nowrap;
```



- **wrap-reverse** indica que los elementos se ajustarán si es necesario, en orden inverso.

```
.Flex-Contenedor{  
  display: flex;  
  flex-wrap: wrap-reverse;
```

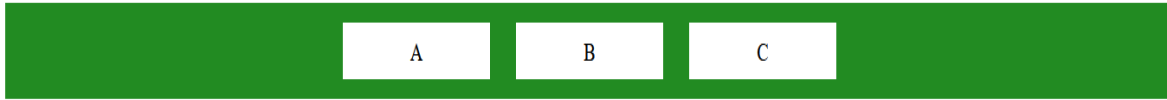


## 3. Justificar el contenido **justify-content**

Se usa para alinear los elementos y tiene 5 posibles valores.

- **center** alinea los elementos flexbox en el centro del contenedor

```
.Flex-Contenedor{
  display: flex;
  justify-content: center;
```



- **flex-start** alinea los elementos al comienzo del contenedor (predeterminado):

```
.Flex-Contenedor{
  display: flex;
  justify-content: flex-start;
```



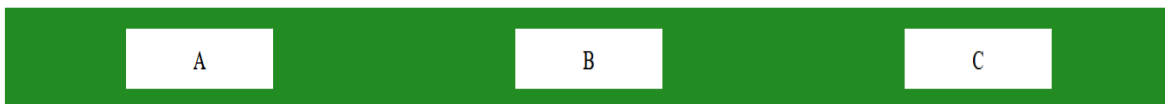
- **flex-end** alinea los elementos al final del contenedor:

```
.Flex-Contenedor{
  display: flex;
  justify-content: end;
```



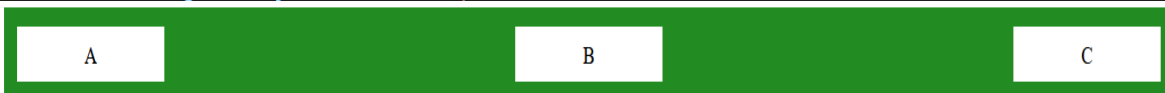
- **space-around** muestra los elementos con espacio antes, entre y después de las líneas:

```
.Flex-Contenedor{
  display: flex;
  justify-content: space-around;
```



- **space-between** muestra los elementos con espacio entre las líneas:

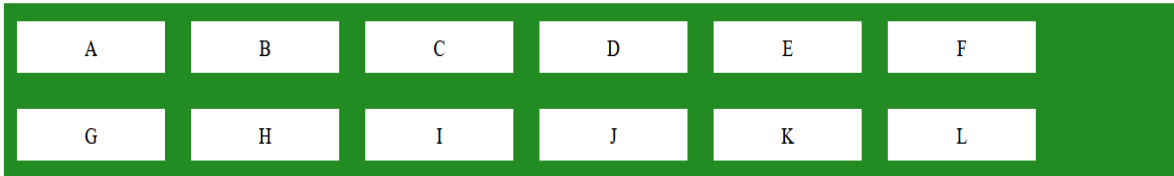
```
.Flex-Contenedor{
  display: flex;
  justify-content: space-between;
```



#### 4. flujo flexible **flex-flow**

- **flex-flow** es una propiedad abreviada para establecer las propiedades **flex-direction** y **flex-wrap**.

```
.Flex-Contenedor{  
    display: flex;  
    flex-flow: row wrap;
```



#### 5. Alinear elementos **align-items**

Se usa para alinear los elementos y tiene 5 posibles valores

- **center** alinea los elementos flexibles en el medio del contenedor:

```
.Flex-Contenedor{  
    display: flex;  
    height: 300;  
    align-items: center;
```



- **flex-start** alinea los elementos flexibles en la parte superior del contenedor:

```
.Flex-Contenedor{  
    display: flex;  
    height: 300px;  
    align-items: flex-start;
```



- **flex-end** alinea los elementos flexibles en la parte inferior del contenedor:

```
.Flex-Contenedor{  
    display: flex;
```

```
height: 300px;
align-items: flex-end;
```



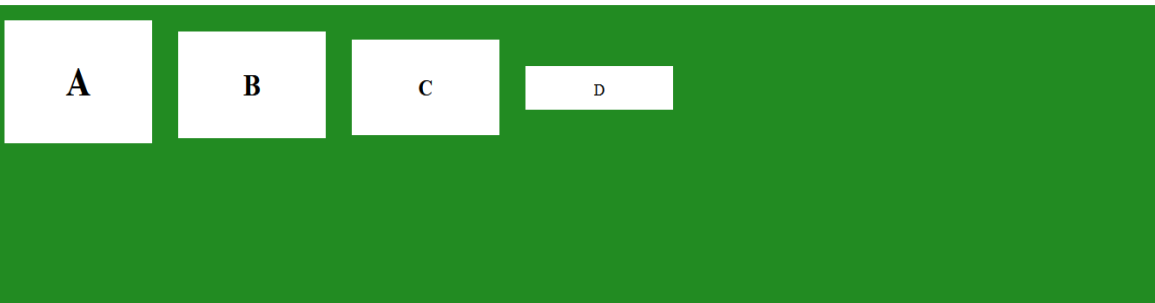
- **stretch** estira los elementos flexibles para llenar el contenedor (predeterminado):

```
.Flex-Contenedor{
  display: flex;
  height: 300px;
  align-items: stretch;
```



- **baseline** alinea los elementos flexibles, como sus líneas de base se alinean: para el ejemplo se utilizaron diferentes tamaños de letra.

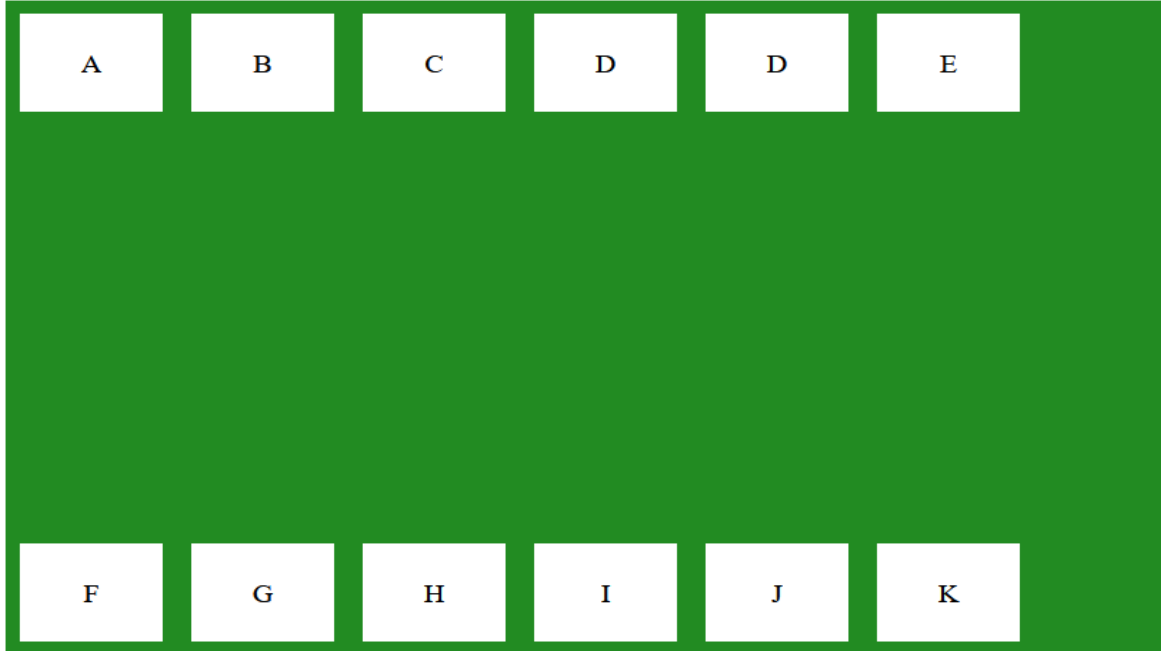
```
.Flex-Contenedor{
  display: flex;
  height: 300px;
  align-items: baseline;
```



6. Alinear Contenido **align-content**  
propiedad se usa para alinear las líneas flexibles y tiene 5 posibles valores.

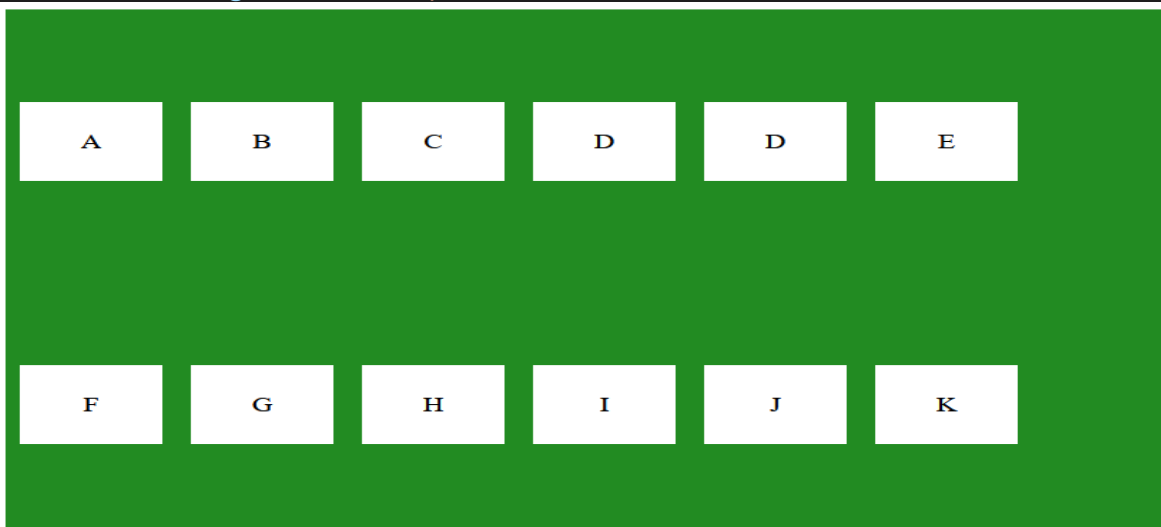
- **space-between** muestra las líneas flexibles con el mismo espacio entre ellas:

```
.Flex-Contenedor{  
    display: flex;  
    height: 800px;  
    flex-wrap: wrap;  
    align-content: space-between;
```

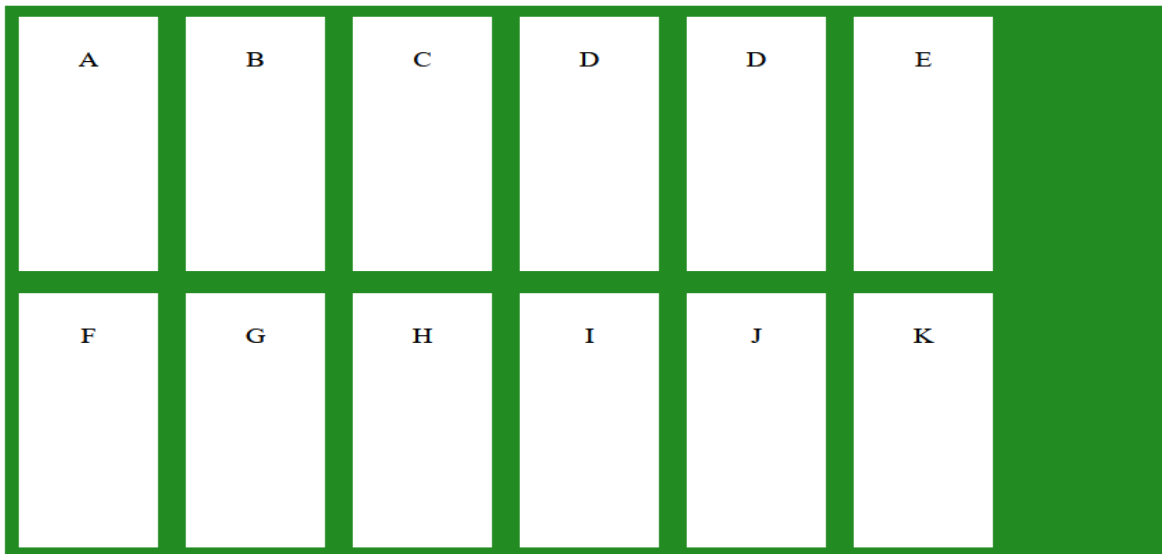


- **space-around** muestra las líneas flexibles con espacio antes, entre y después de ellas:

```
.Flex-Contenedor{  
    display: flex;  
    height: 500px;  
    flex-wrap: wrap;  
    align-content: space-around;
```

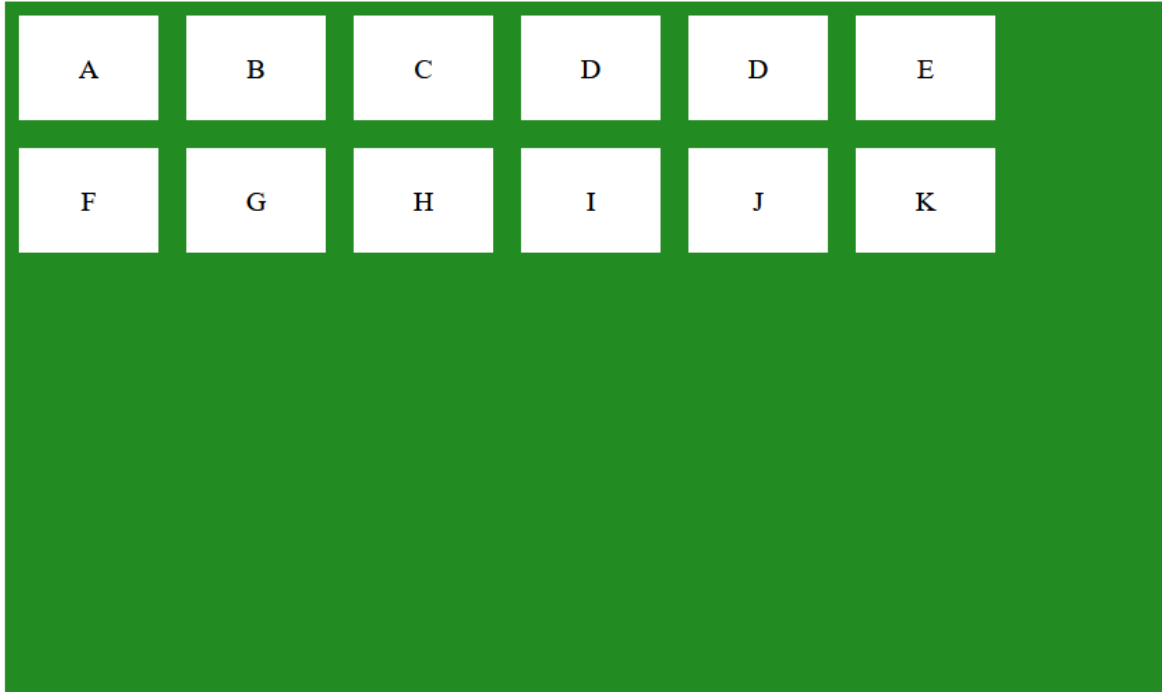


- **stretch** estira las líneas flexibles para ocupar el espacio restante (predeterminado):



- **flex-start** muestra las líneas flexibles al comienzo del contenedor:

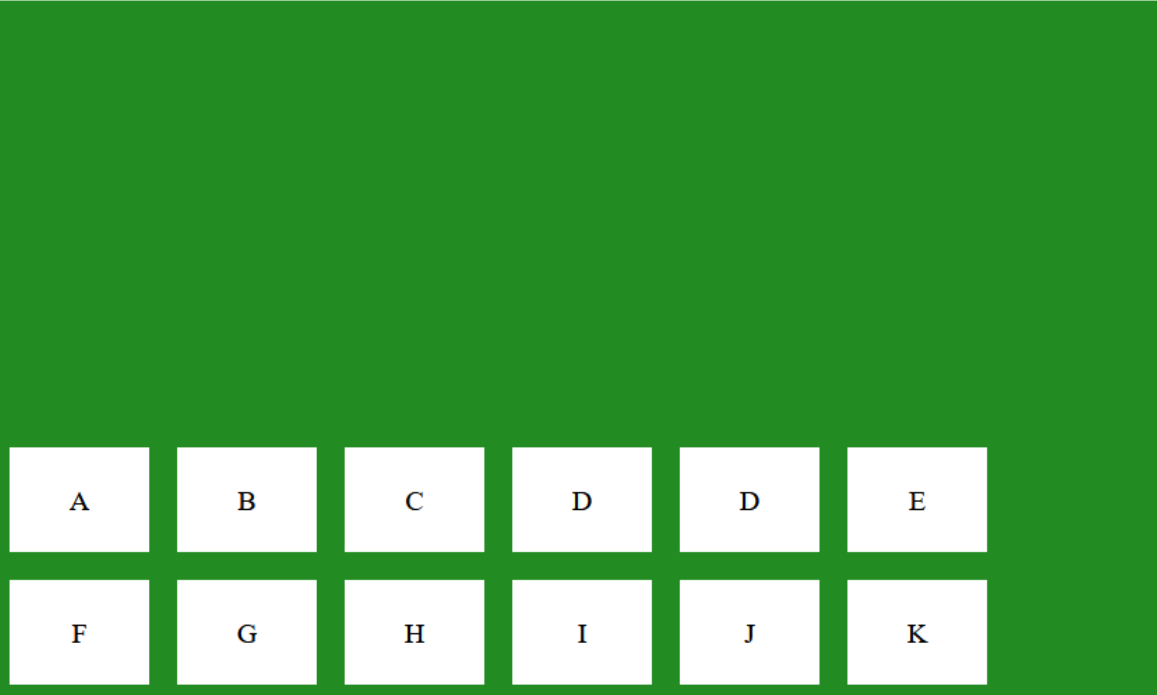
```
.Flex-Contenedor{  
  display: flex;  
  height: 500px;  
  flex-wrap: wrap;  
  align-content: flex-start;
```





- **flex-end** muestra las líneas flexibles al final del contenedor:

```
.Flex-Contenedor{  
  display: flex;  
  height: 500px;  
  flex-wrap: wrap;  
  align-content: flex-end;
```

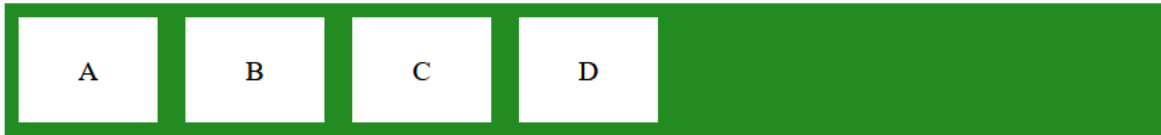


### Elementos de CSS FlexBox

Los elementos secundarios directos de un contenedor flexible se convierten automáticamente en elementos flexibles (flex).

El elemento siguiente muestra cuatro elementos flexibles azules dentro de un contenedor flexible verde.

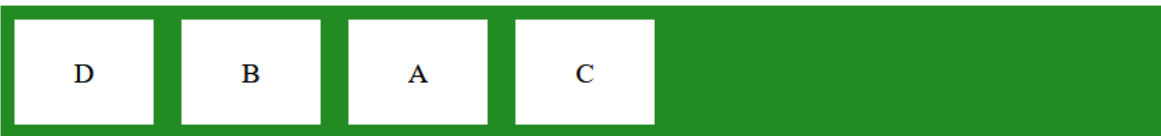
```
<div class="Flex-Contenedor">  
  <div>A</div>  
  <div>B</div>  
  <div>C</div>  
  <div>D</div>  
  
</div>
```



Las propiedades del elemento flexible son:

- order
  - flex-grow
  - flex-shrink
  - flex-basis
  - flex
  - align-self
- **order** que especifica el orden de los elementos flexibles. El primer elemento flexible en el código no tiene que aparecer como el primer elemento en el diseño. El valor del pedido debe ser un número,

```
<div class="Flex-Contenedor">
  <div style="order: 3" >A</div>
  <div style="order: 2" >B</div>
  <div style="order: 4" >C</div>
  <div style="order: 1" >D</div>
</div>
```



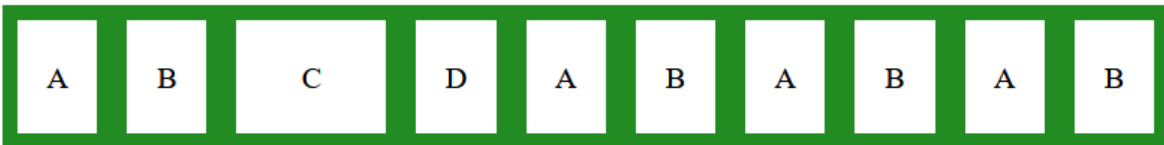
- **flex-grow** especifica cuánto crecerá un elemento flexible en relación con el resto de los elementos flexibles. El valor debe ser un número, el valor predeterminado es 0.

```
<div class="Flex-Contenedor">
  <div style="flex-grow: 1" >A</div>
  <div style="flex-grow: 1" >B</div>
  <div style="flex-grow: 5" >C</div>
  <div style="flex-grow: 8" >D</div>
</div>
```



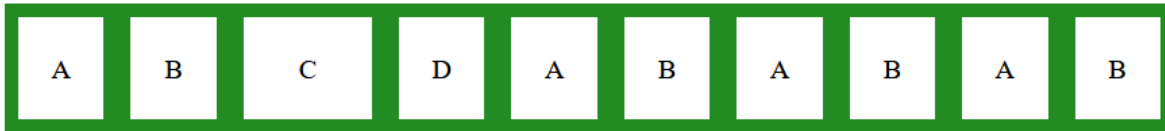
- **flex-shrink** especifica cuánto se encogerá un elemento flexible en relación con el resto de los elementos flexibles. El valor debe ser un número, el valor predeterminado es 1.

```
<div class="Flex-Contenedor">
  <div>A</div>
  <div>B</div>
  <div style="flex-shrink: 0;">C</div>
  <div>D</div>
  <div>A</div>
  <div>B</div>
  <div>A</div>
  <div>B</div>
  <div>A</div>
  <div>B</div>
</div>
```



**flex-basis** especifica la longitud inicial de un elemento flexible.

```
<div class="Flex-Contenedor">
  <div>A</div>
  <div>B</div>
  <div style="flex-basis: 150px;">C</div>
  <div>D</div>
  <div>A</div>
  <div>B</div>
  <div>A</div>
  <div>B</div>
  <div>A</div>
  <div>B</div>
</div>
```



- **flex** es una propiedad abreviada para los **flex-grow**, **flex-shrink** y **flex-basis** propiedades.

```
<div class="Flex-Contenedor">
  <div>A</div>
  <div>B</div>
  <div style="flex: 0 0 200px">C</div>
  <div>D</div>
</div>
```



- **align-self** especifica la alineación del elemento seleccionado dentro del contenedor flexible.  
**align-self** anula la alineación predeterminada establecida por **align-items** del contenedor.

```
<div class="Flex-Contenedor">
  <div>A</div>
  <div>B</div>
  <div style="align-self: center;">C</div>
  <div>D</div>
</div>
```

