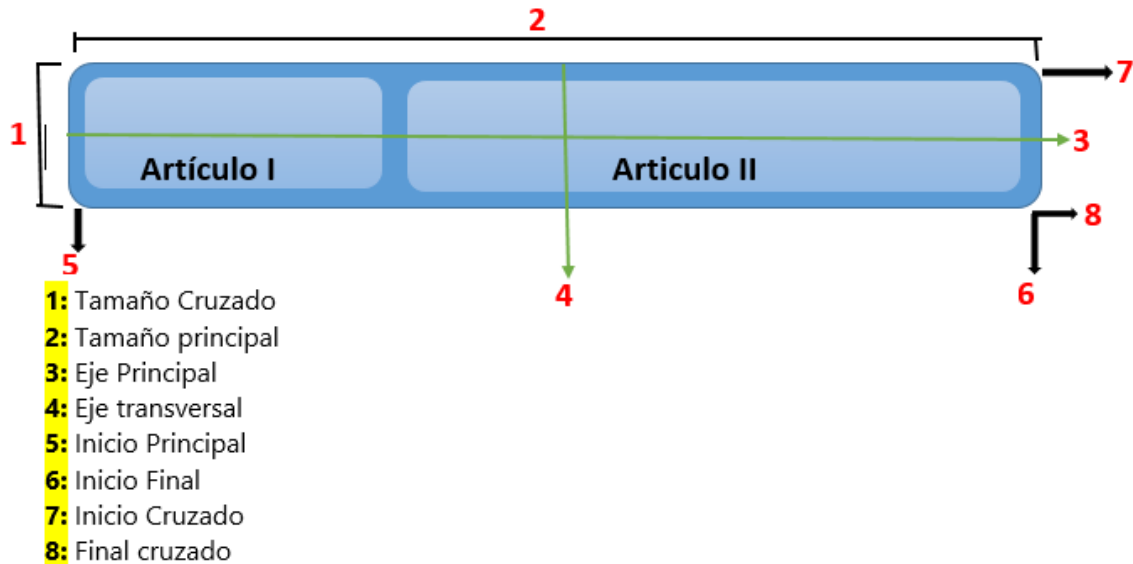


DISTRIBUCION EN FLEX BOX



- **eje principal:** el eje principal de un contenedor flexible es el eje principal a lo largo del cual se disponen los elementos flexibles. Cuidado, no es necesariamente horizontal; Depende de la **flex-direction** “propiedad (ver más abajo)”.
- **inicio principal | main-end** : los elementos flexibles se colocan dentro del contenedor comenzando desde el inicio principal y yendo al extremo principal.
- **tamaño principal:** el ancho o la altura de un elemento flexible, lo que esté en la dimensión principal, es el tamaño principal del elemento. La propiedad de tamaño principal del elemento flexible es la propiedad 'ancho' o 'alto', cualquiera que esté en la dimensión principal.
- **eje transversal:** el eje perpendicular al eje principal se denomina eje transversal. Su dirección depende de la dirección del eje principal.
- **inicio cruzado | extremo cruzado:** las líneas flexibles se llenan con artículos y se colocan en el contenedor comenzando en el lado de inicio cruzado del contenedor flexible y yendo hacia el lado del extremo cruzado.
- **tamaño cruzado:** el ancho o la altura de un elemento flexible, lo que esté en la dimensión cruzada, es el tamaño cruzado del elemento. La propiedad de tamaño cruzado es cualquiera de 'ancho' o 'alto' que esté en la dimensión cruzada.

PROPIEDADES

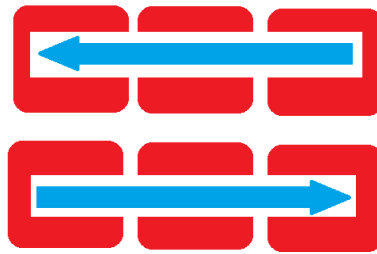
Contenedor flexible

DISPLAY

Esto define un contenedor flexible; en línea o bloque dependiendo del valor dado. Permite un contexto flexible para todos sus hijos directos. Tenga en cuenta que las columnas CSS no tienen efecto en un contenedor flexible.

```
.container {  
  display: flex; /* or inline-flex */  
}
```

FLEX-DIRECTION



Esto establece el eje principal, definiendo así la dirección en que se colocan los artículos flexibles en el contenedor flexible. Flexbox es (aparte del envoltorio opcional) un concepto de diseño de una sola dirección. Piense en los elementos flexibles como dispuestos principalmente en filas horizontales o columnas verticales.

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

FLEX-WRAP

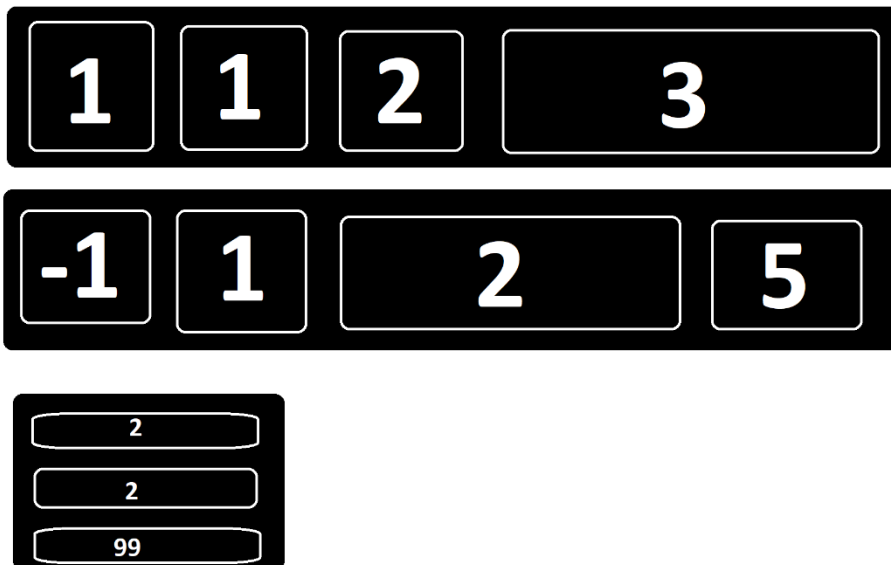
De forma predeterminada, todos los elementos flexibles intentarán encajar en una línea. Puede cambiar eso y permitir que los elementos se ajusten según sea necesario con esta propiedad.

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

- **nowrap:** (predeterminado): todos los elementos flexibles estarán en una línea
- **wrap:** los elementos flexibles se ajustarán a varias líneas, de arriba a abajo.
- **wrap-reverse:** los elementos flexibles se ajustarán a varias líneas de abajo hacia arriba.

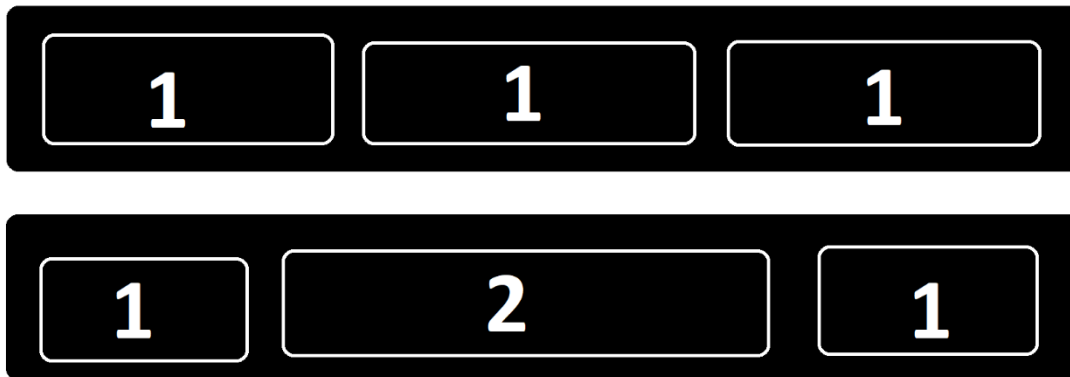
Artículos flexible

De forma predeterminada, los elementos flexibles se presentan en el orden de origen. Sin embargo, la propiedad controla el orden en que aparecen en el contenedor flexible. Los artículos con el mismo orden de origen vuelven.



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

FLEX-GROW



Esto define la capacidad de crecimiento de un elemento flexible si es necesario. Acepta un valor sin unidades que sirve como proporción. Determina qué cantidad de espacio disponible dentro del contenedor flexible debe ocupar el artículo.

Si todos los elementos se han establecido en 1, el espacio restante en el contenedor se distribuirá por igual a todos los niños. Si uno de los hijos tiene un valor de 2, ese hijo ocuparía el doble de espacio que cualquiera de los otros (o lo intentará, al menos).

```
.item {  
  flex-grow: 4; /* default 0 */  
}
```

Los números negativos no son válidos.

FLEX-SHRINK

Esto define la capacidad de que un elemento flexible se encoja si es necesario.

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

FLEX-BASIS

Esto define el tamaño predeterminado de un elemento antes de que se distribuya el espacio restante. Puede ser una longitud o una palabra clave. La palabra clave significa "mirar mi propiedad de ancho o alto" (lo cual fue hecho temporalmente por la **main-size** palabra clave hasta que quedó en desuso). La palabra clave significa "dimensionarlo en función del contenido del elemento". Esta palabra clave aún no es compatible, **content** por lo que es difícil probarla y saber qué hacen sus hermanos. **max-content****min-content****fit-content**

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

Si se establece en 0, el espacio adicional alrededor del contenido no se tiene en cuenta. Si se establece en auto, el espacio adicional se distribuye en función de su **flex-grow**

FLEX

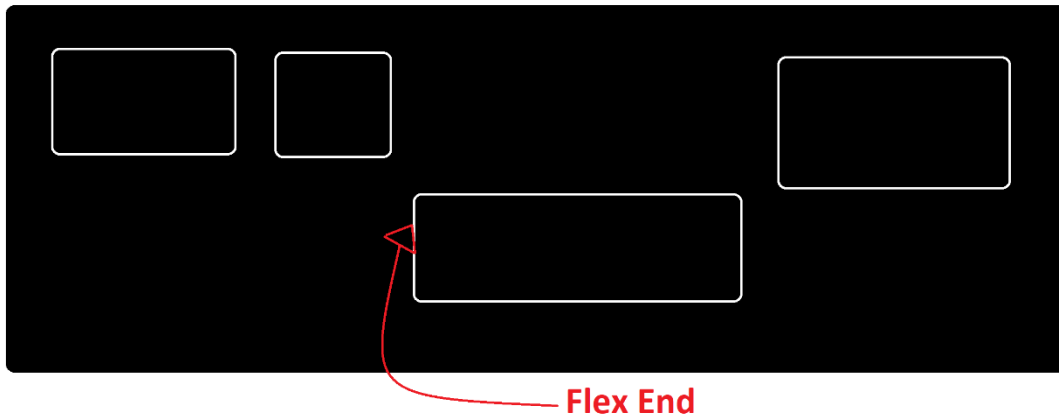
Esta es la abreviatura de **flex-grow**, **flex-shrink** y **flex-basis** combinado. Los parámetros segundo y tercero (**flex-shrink** y **flex-basis**) son opcionales. El valor predeterminado es 0-1, pero si lo configura con un solo valor numérico, como **flex: 5;**, eso cambia **flex-basis** a 0%, por lo que es como configurar **flex-grow: 5; flex-shrink: 1; flex-basis: 0%;**.

```
.item {  
  flex: none | [ <'flex-grow'> <'flex-shrink'>?  
}
```

ALIGN-SELF

Esto permite anular la alineación predeterminada (o la especificada por **align-items**) para elementos flexibles individuales.

flex start



Flex End

Tenga en cuenta que **float**, **clear** y **vertical-align** no tienen efecto en un elemento flexible.

```
.item {  
  align-self: auto | flex-start | flex-end |  
}
```