

前端布局神器display:flex

CSS Flexbox

2009年，W3C提出了一种新的方案--Flex布局，可以简便、完整、响应式地实现各种页面布局。目前已得到所有现在浏览器的支持。

Browser Support



Chrome
21+



Opera
12.1+



Firefox
22+



Safari
6.1+



IE
10+

flex浏览器支持

一、Flex布局是什么？

Flex是Flexible Box的缩写，翻译成中文就是“弹性盒子”，用来为盒装模型提供最大的灵活性。任何一个容器都可以指定为Flex布局。

```
.box{  
  display: -webkit-flex; /*在webkit内核的浏览器上使用要加前缀*/  
  display: flex; //将对象作为弹性伸缩盒显示  
}
```

当然，行内元素也可以使用Flex布局。

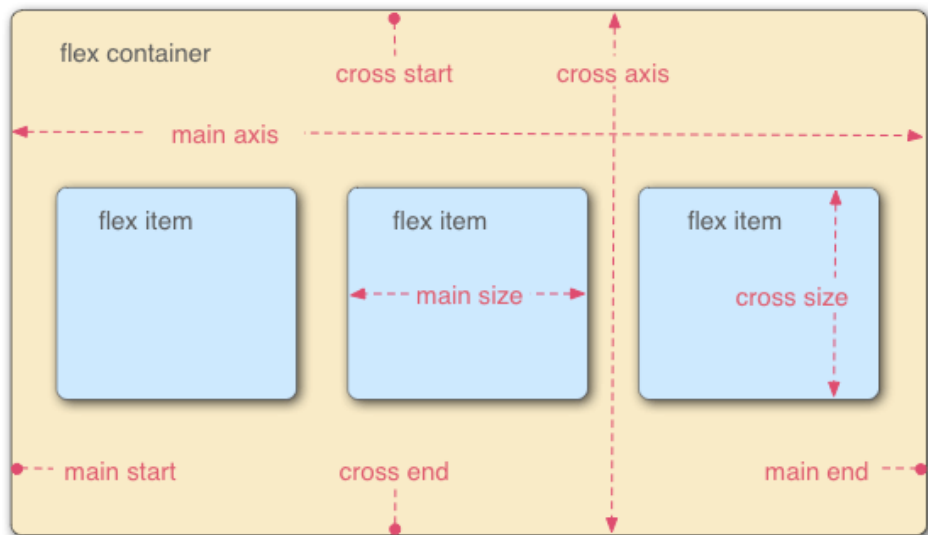
```
.box {  
  display: inline-flex; //将对象作为内联块级弹性伸缩盒显示  
}
```

兼容性写法

```
.box {  
  display: flex || inline-flex;  
}
```

二、基本概念

采用Flex布局的元素，被称为Flex容器(flex container)，简称“容器”。其所有子元素自动成为容器成员，成为Flex项目(Flex item)，简称“项目”。



结构示意图

容器默认存在两根主轴：水平方向主轴(main axis)和垂直方向交叉轴(cross axis)，默认项目按主轴排列。

- main start/main end：主轴开始位置/结束位置；
- cross start/cross end：交叉轴开始位置/结束位置；
- main size/cross size：单个项目占据主轴/交叉轴的空间；

三、容器属性

设置在容器上的属性有6种。

- flex-direction
- flex-wrap
- flex-flow
- justify-content
- align-item
- align-content

flex-direction属性：决定主轴的方向（即项目的排列方向）

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

- row（默认）：主轴水平方向，起点在左端；
- row-reverse：主轴水平方向，起点在右端；
- column：主轴垂直方向，起点在上边沿；

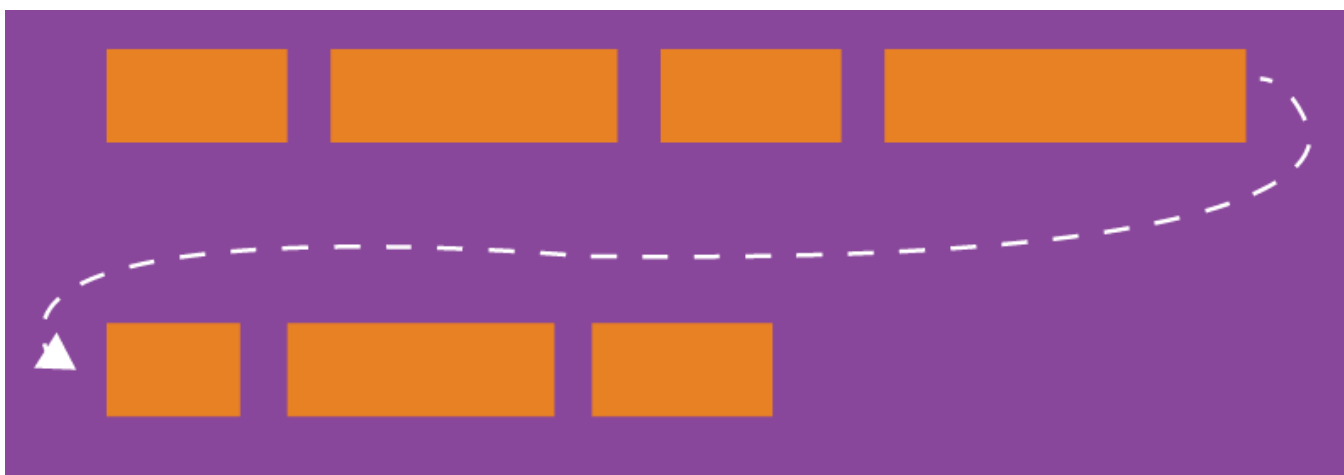
- column-reserve: 主轴垂直方向，起点在下边沿。



主轴的4个方向

flex-wrap属性：定义换行情况

默认情况下，项目都排列在一条轴线上，但有可能一条轴线排不下。



一条轴线排不下

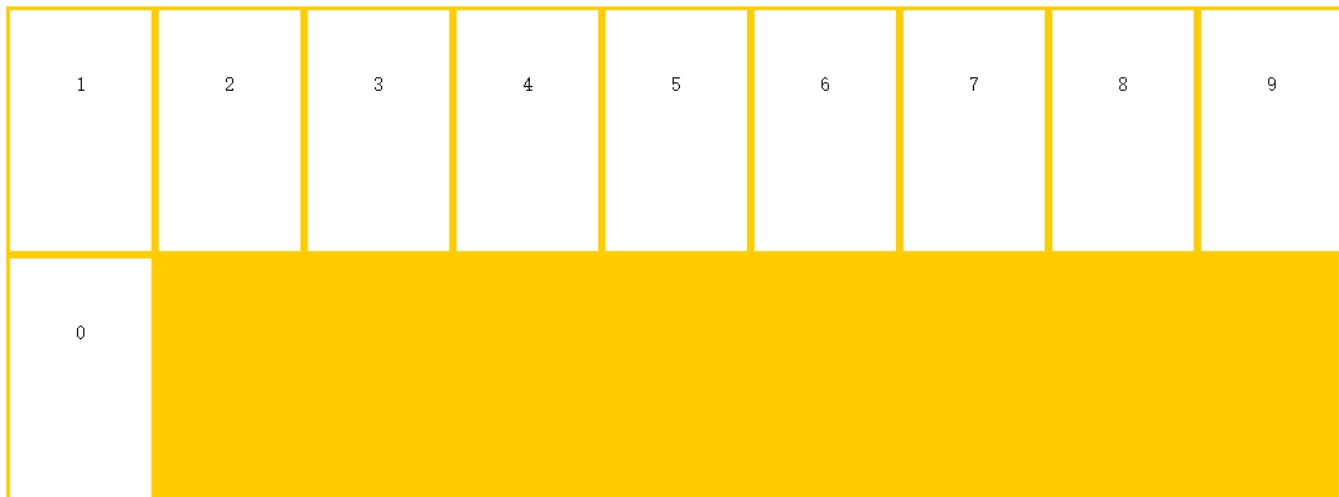
```
.box{  
  flex-wrap: nowrap | wrap | wrap-reverse;  
}
```

- nowrap (默认)：不换行；

1	2	3	4	5	6	7	8	9	0

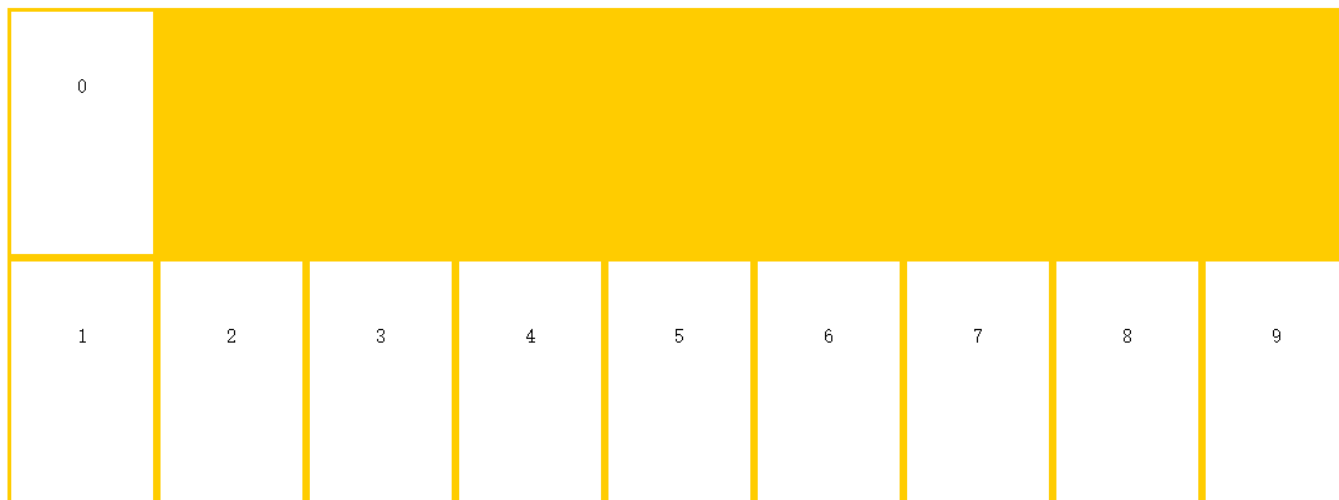
不换行nowrap

- wrap: 换行，第一行在上方；



换行，第一行在上

- wrap-reverse: 换行，第一行在下方。



换行，第一行在下

flex-flow属性：flex-direction和flex-wrap的简写，默认row nowrap

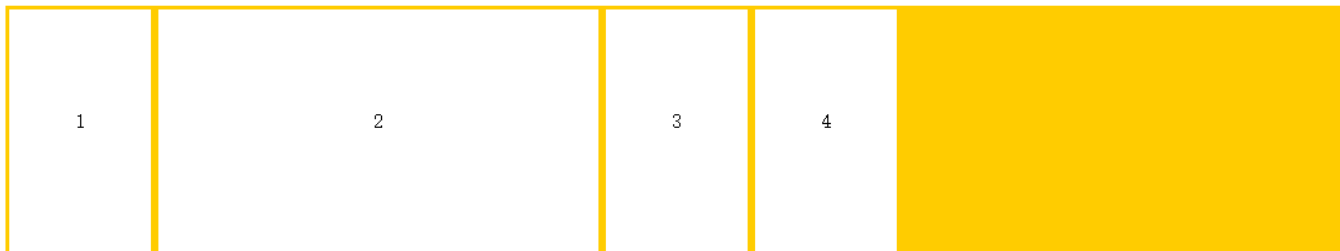
```
.box{  
  flex-flow: <flex-direction> || <flex-wrap>;  
}
```

justify-content属性：定义项目在主轴上的对齐方式。

对齐方式与轴的方向有关，本文中假设主轴从左到右。

```
.box {  
  justify-content: start | end | flex-start | flex-end | center | left | right | space-between | space-around  
| space-evenly | stretch | safe | unsafe | baseline | first baseline | last baseline;  
}
```

- flex-start (默认值)：左对齐；



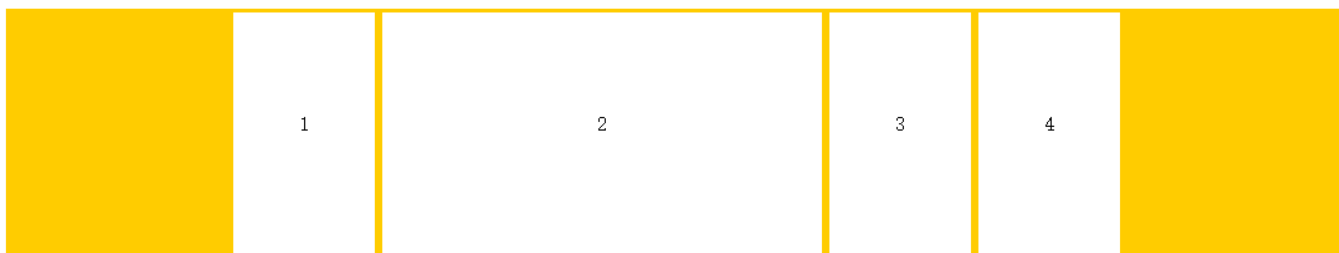
左对齐

- flex-end：右对齐；



右对齐

- center：居中；



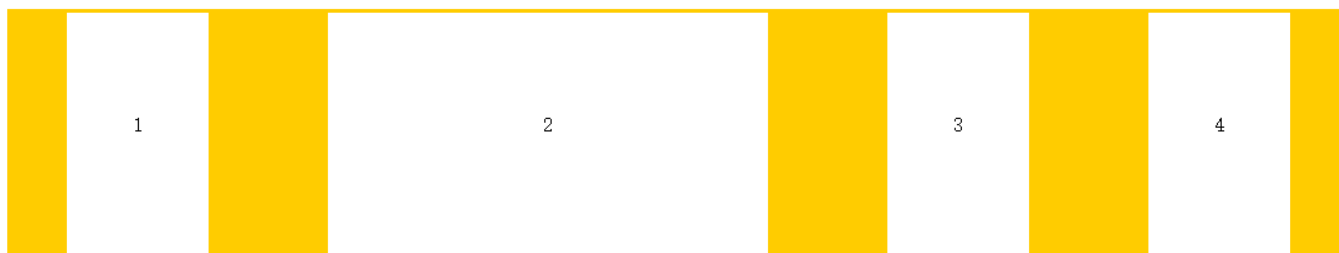
居中对齐

- space-between：两端对齐，项目之间间隔相等；



两端对齐

- space-around：每个项目两侧的间隔相等，即项目之间的间隔比项目与边框的间隔大一倍。



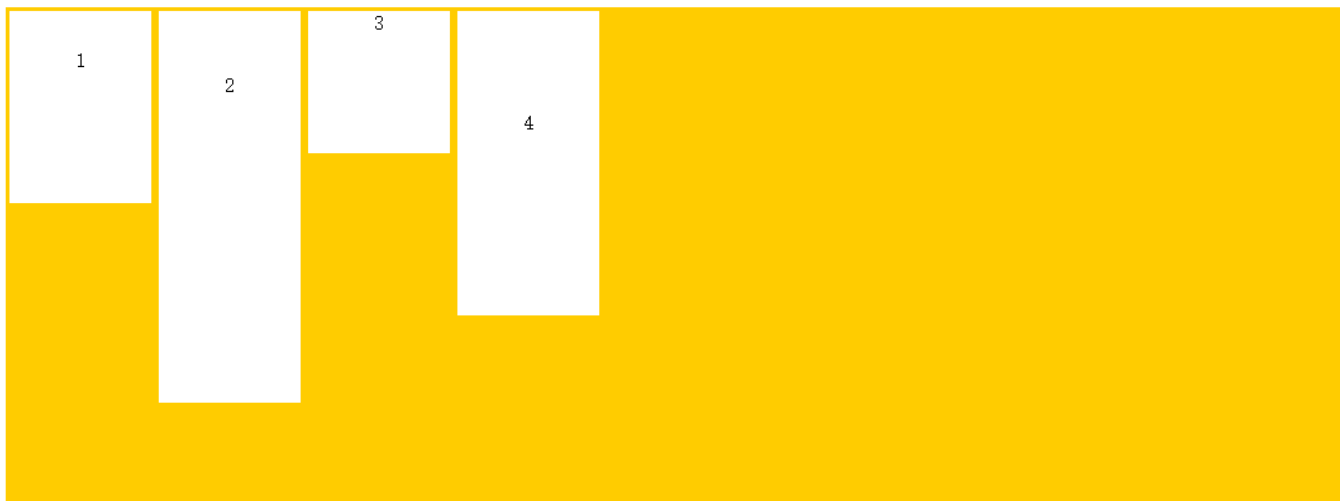
两侧间隔相等

align-items属性：定义在交叉轴上的对齐方式

对齐方式与交叉轴的方向有关，假设交叉轴从下到上。

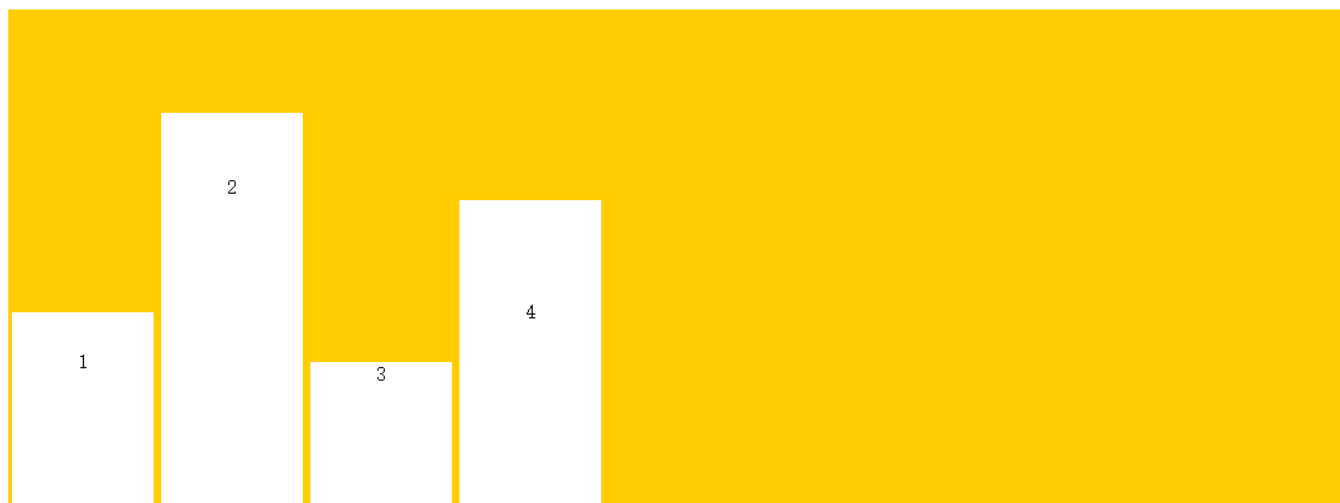
```
.box{  
  align-items: flex-start | flex-end | center | baseline | stretch;  
}
```

- flex-start: 起点对齐;



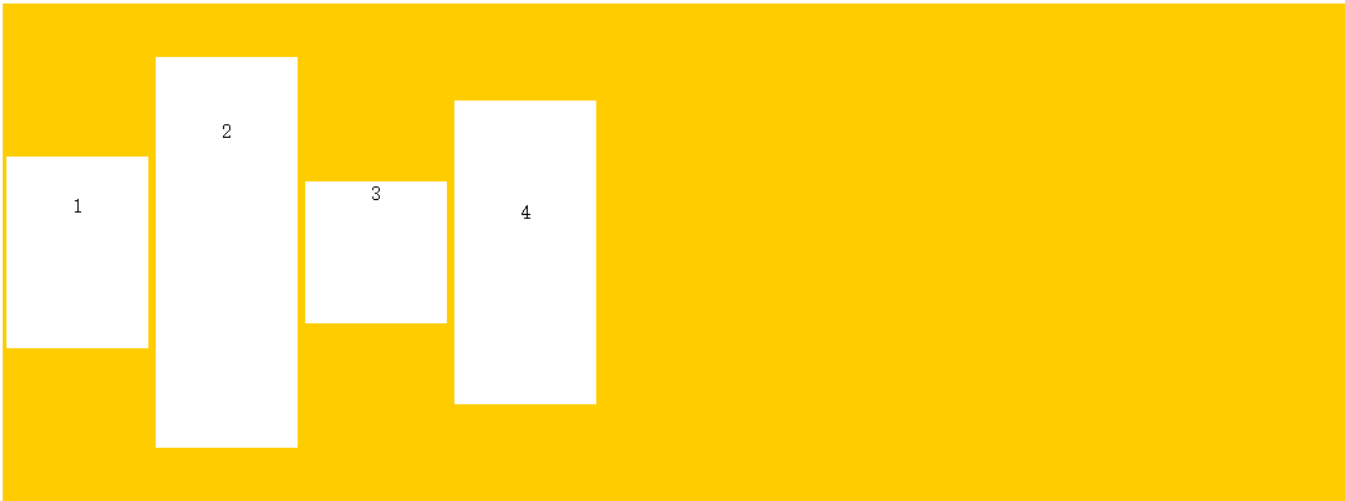
起点对齐

- flex-end: 终点对齐;



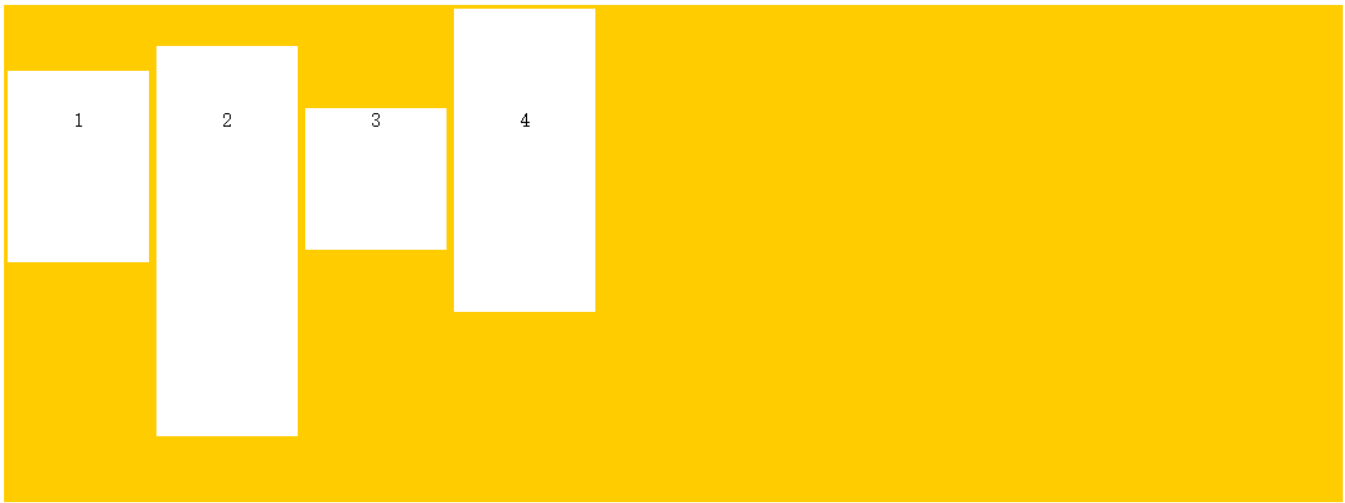
终点对齐

- center: 中点对齐;



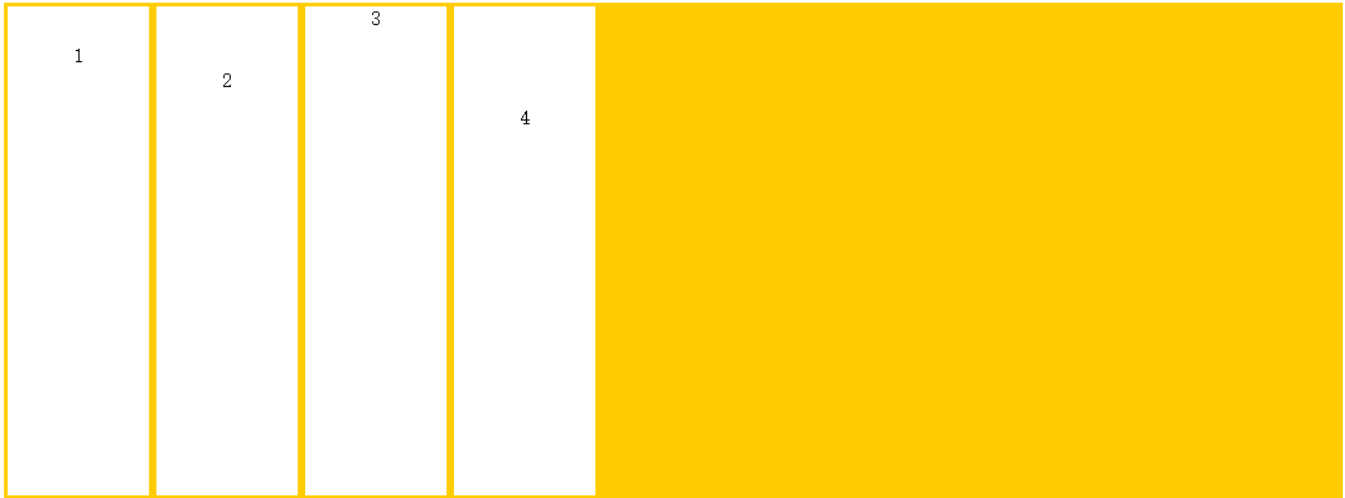
中点对齐

- baseline: 项目的第一行文字的基线对齐;



基线对齐

- stretch (默认值) : 如果项目未设置高度或设为auto, 将占满整个容器的高度。

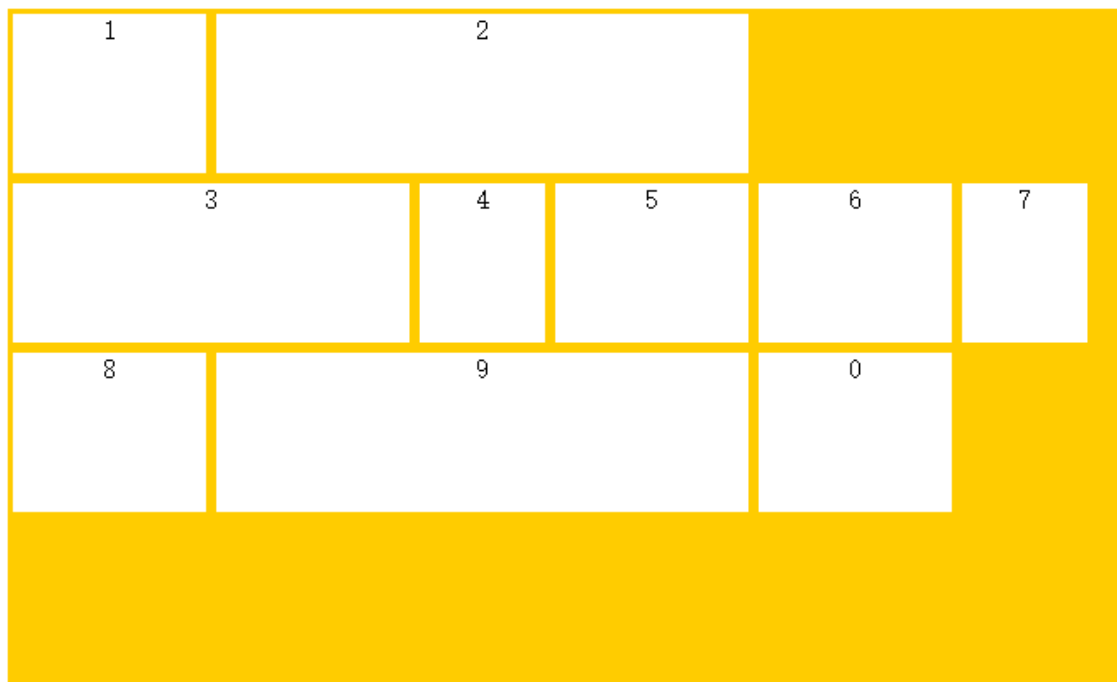


align-content属性: 定义多根轴线的对齐方式

如果项目只有一根轴线, 该属性不起作用。
所以, 容器必须设置flex-wrap: ...;

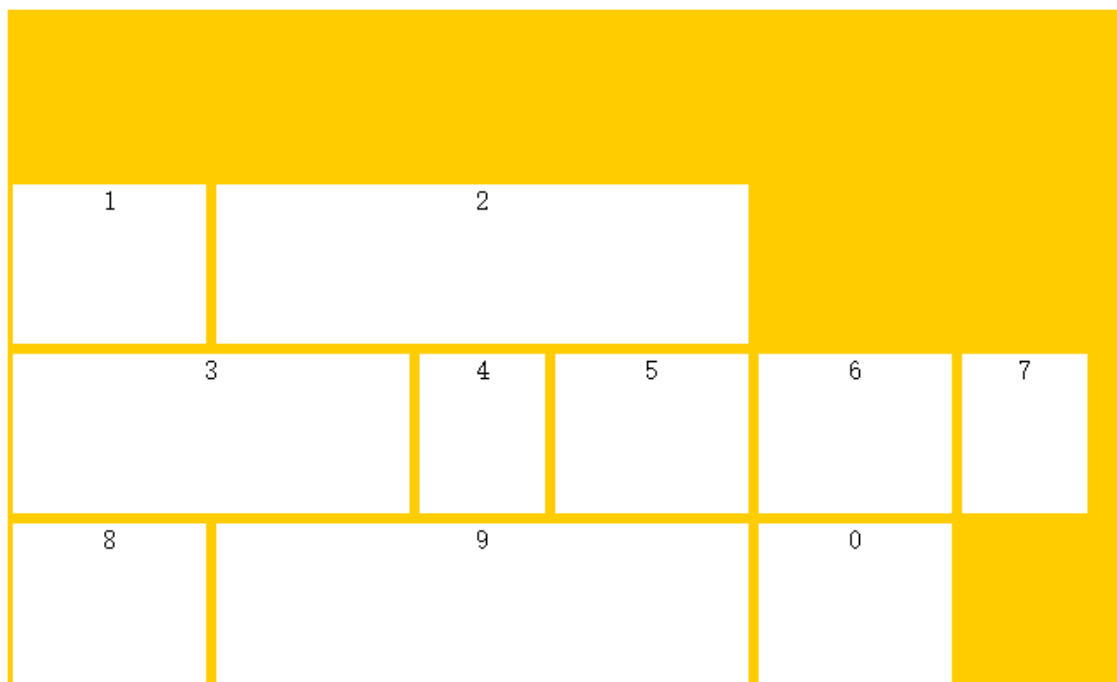
```
.box{  
  align-content: flex-start | flex-end | center | space-between | space-around | stretch;  
}
```

- flex-start: 与交叉轴的起点对齐;



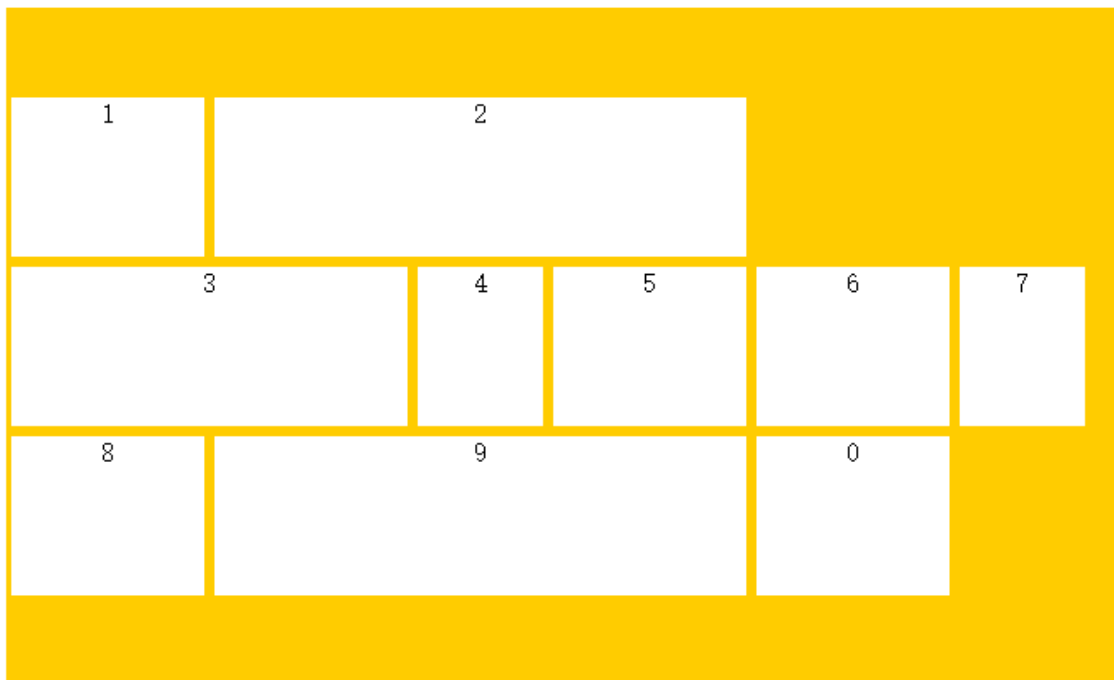
起点对齐

- flex-end: 与交叉轴的终点对齐;



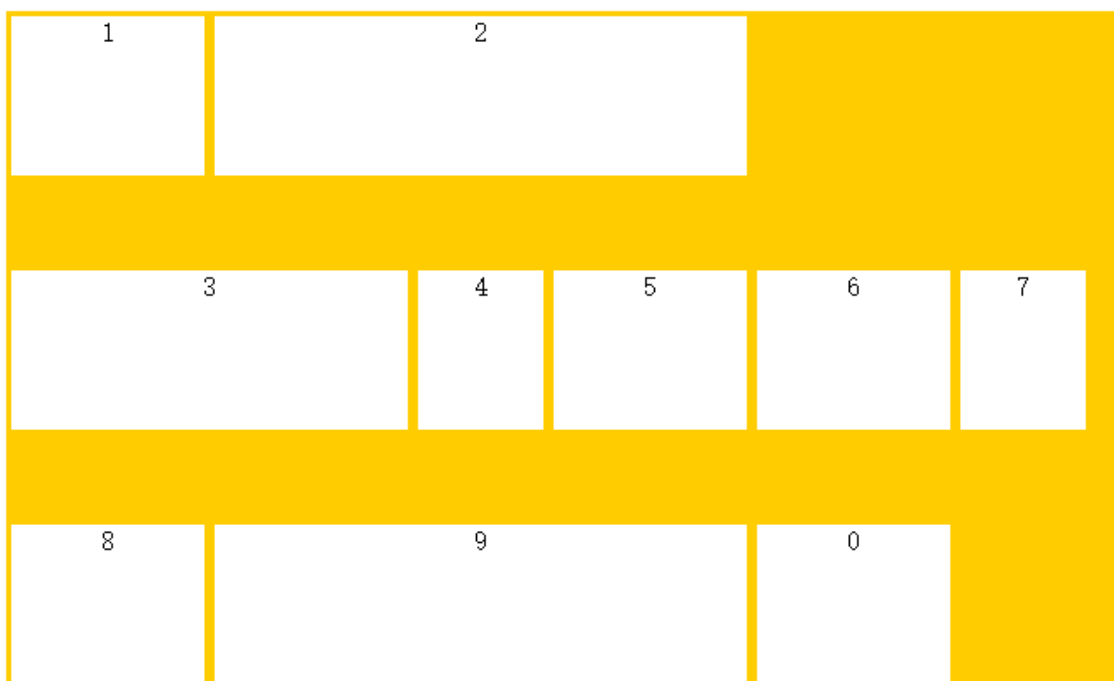
终点对齐

- center: 与交叉轴的中点对齐;



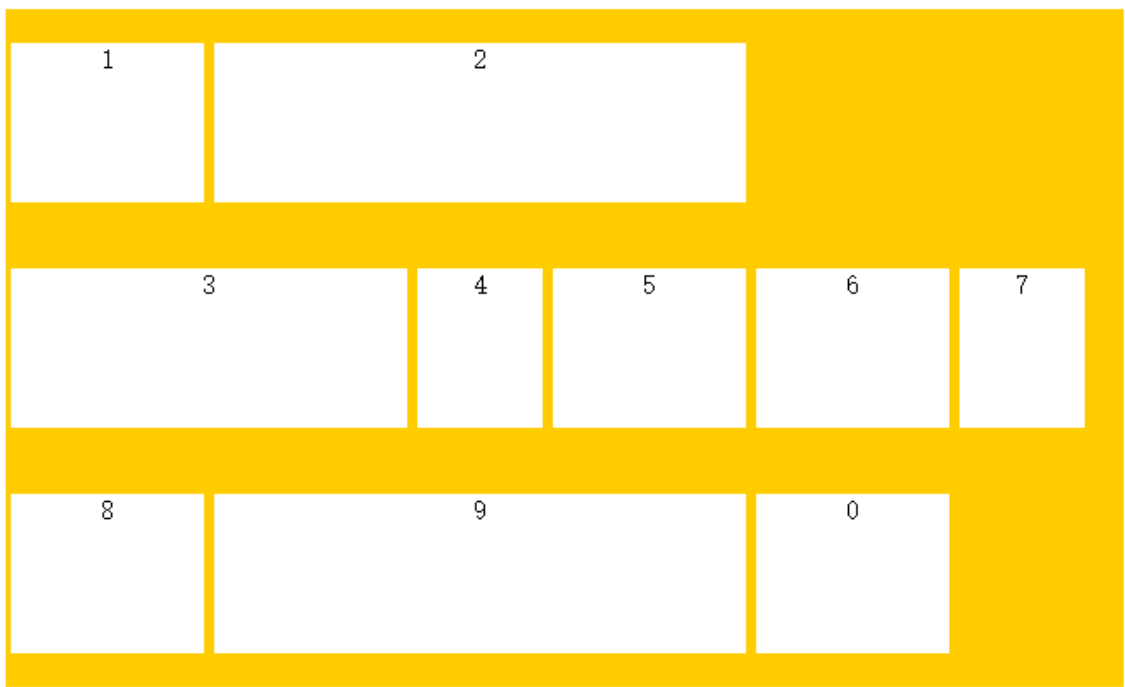
中点对齐

- space-between: 与交叉轴的两端对齐，轴线之间的间隔平均分布；



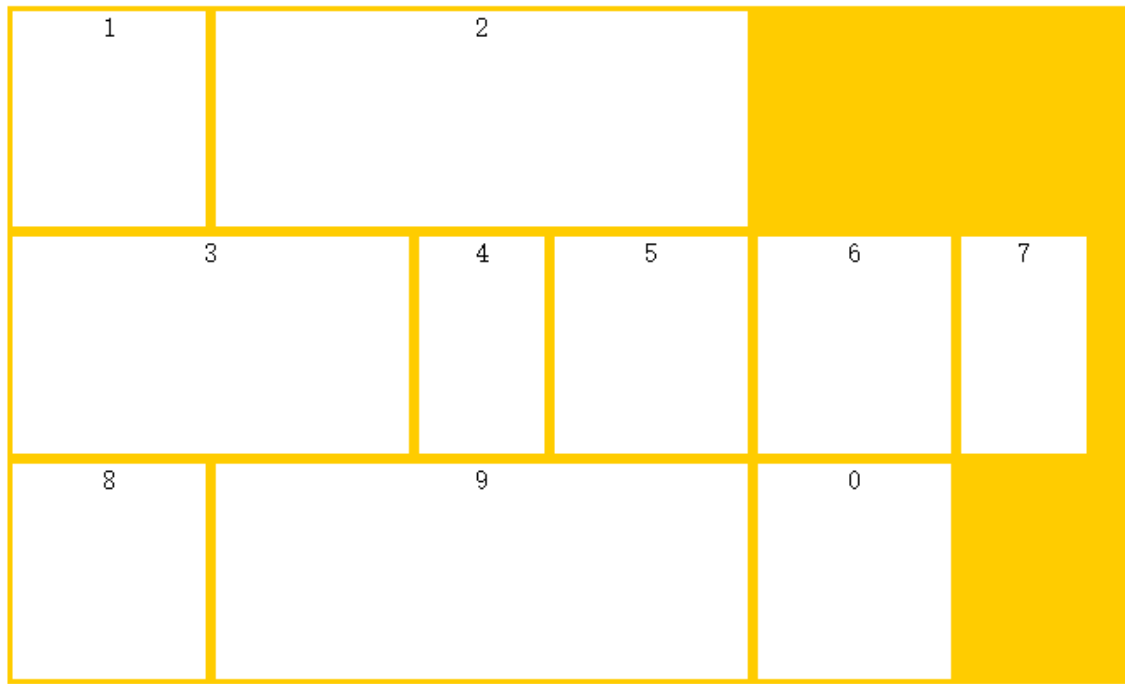
轴线之间等间距

- space-around: 每根轴线两侧的间隔相等，即轴线之间的间隔比轴线与边框的间隔大一倍；



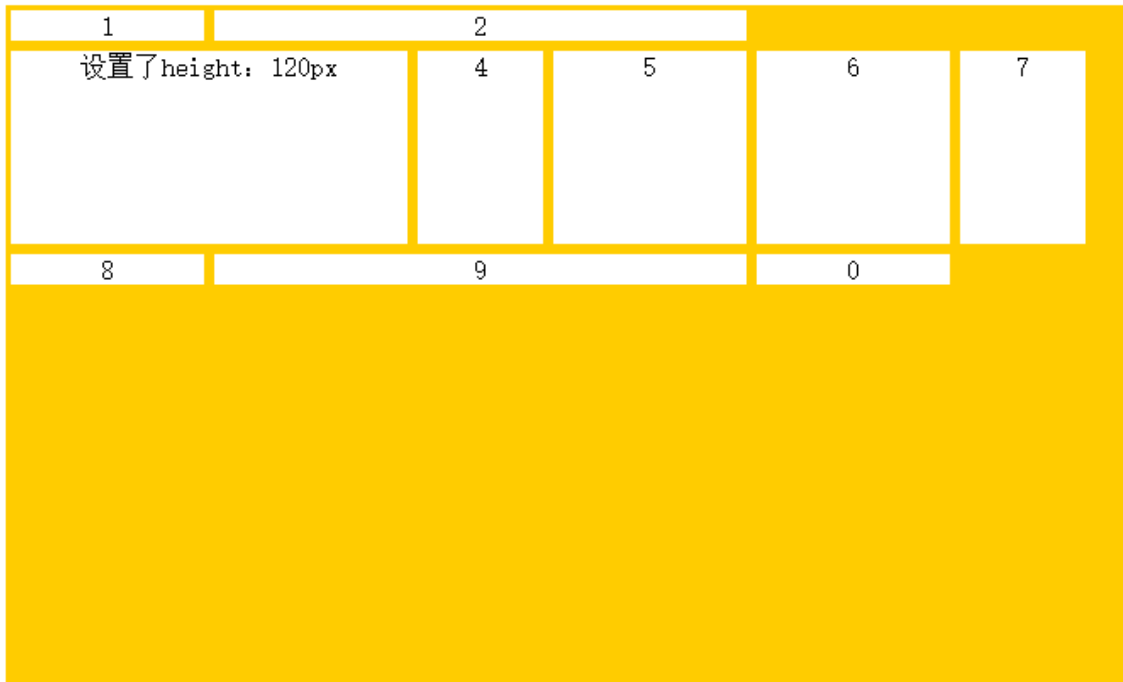
轴线两侧等间距

- stretch（默认值）：轴线占满整个交叉轴。



项目未设置高度时

有意思的是，当你不给项目设置高度但是给容器设置align-content不为stretch时，同一轴线上的项目的高度将等于项目中高度最高的项目。



四、项目的属性

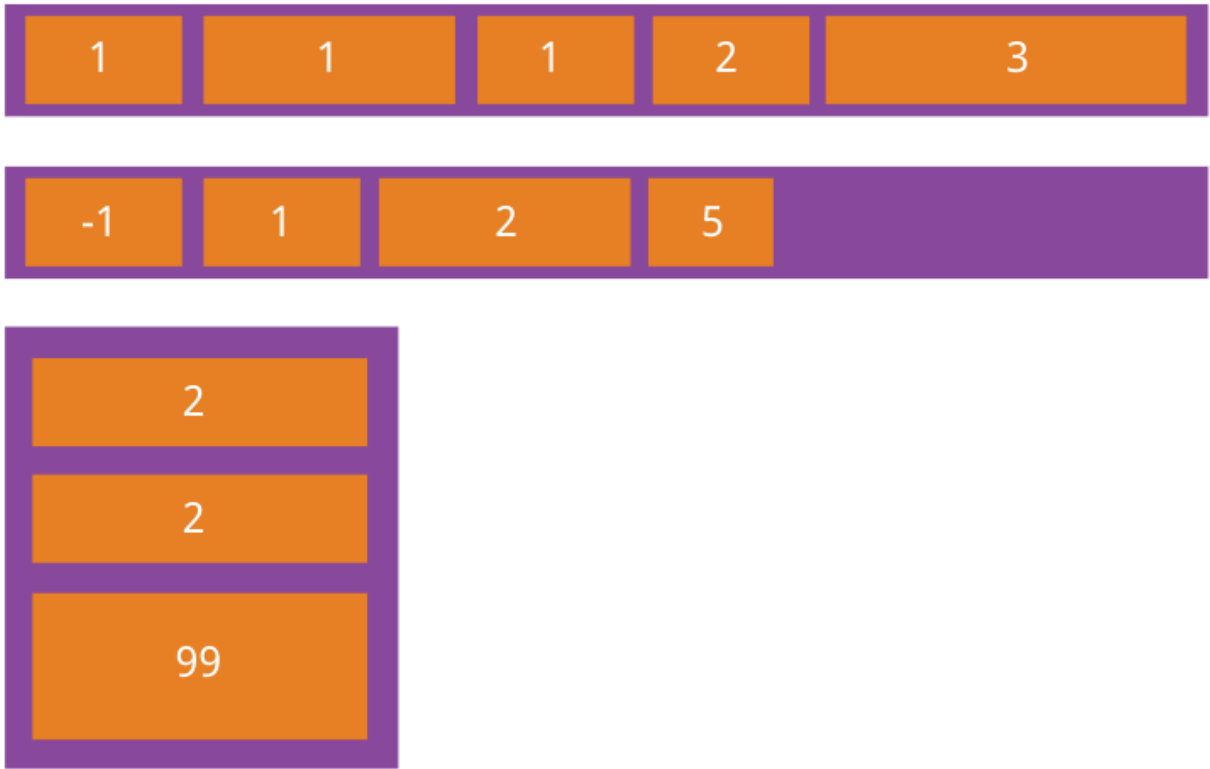
设置在项目上的属性也有6个。

- order
- flex-grow
- flex-shrink
- flex-basis
- flex
- align-self

order属性：定义项目的排列顺序。

数值越小，排列越靠前，默认为0，可以是负值。

```
.item {  
  order: <整数>;  
}
```



展示效果不明显，直接盗图

flex-grow属性：定义项目的放大比例

默认值为0，即如果空间有剩余，也不放大。
可以是小数，按比例占据剩余空间。



默认情况

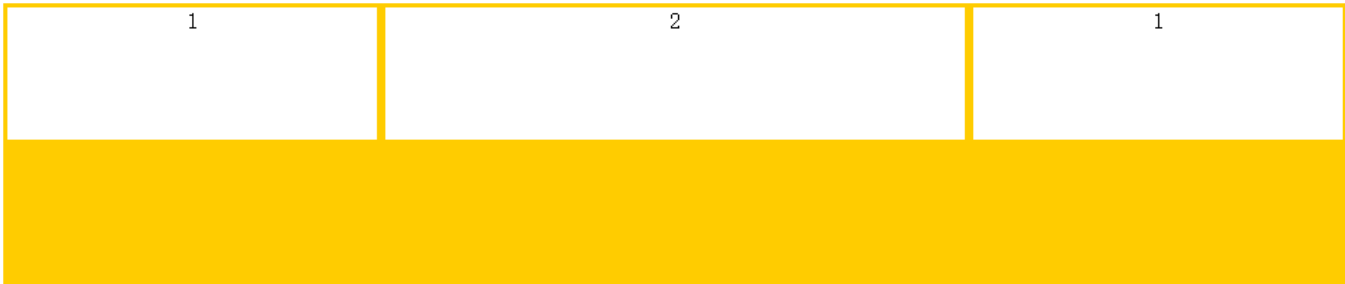
```
.item{
  flex-grow: <数字>;
}
```

若所有项目的flex-grow的数值都相同，则等分剩余空间



等分剩余空间

若果有一个项目flex-grow为2，其余都为1，则该项目占据剩余空间是其余的2倍



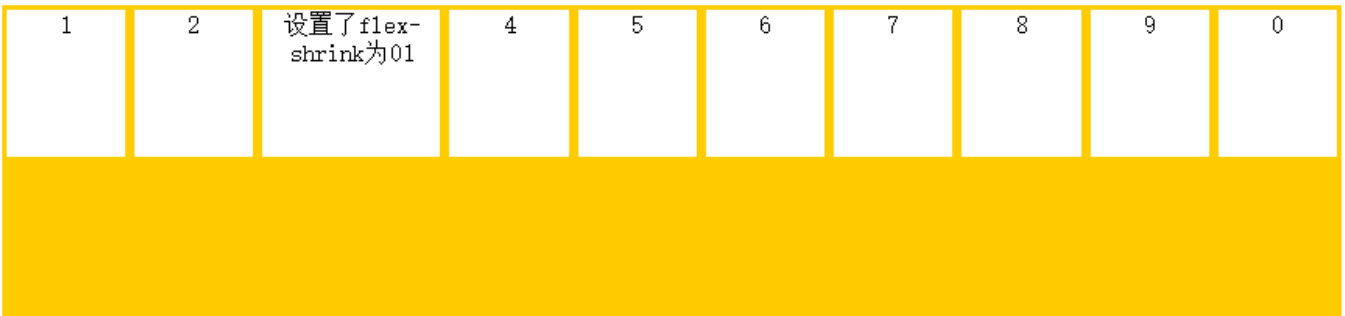
不等分占据

flex-shrink属性：定义项目的缩小比例

- 默认值都为1，即如果空间不足将等比例缩小。
- 如果有一个项目的值为0，其他项目为1，当空间不足时，该项目不缩小。
- 负值对该属性无效，容器不应该设置flex-wrap。

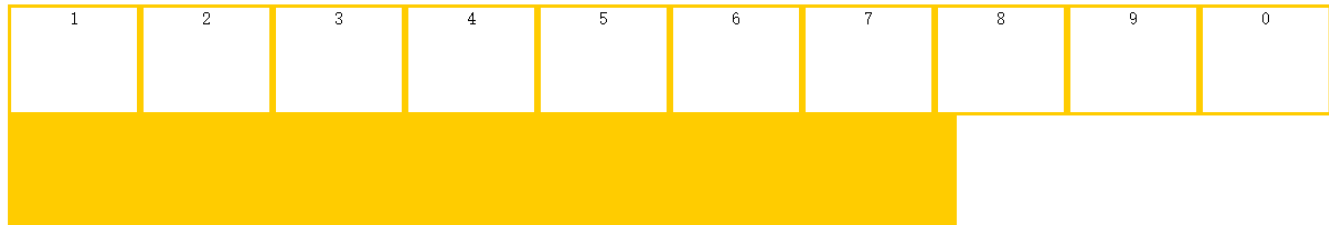
```
.item{
  flex-shrink: <非负整数>;
}
```

如果一个项目设置flex-shrink为0；而其他项目都为1，则空间不足时，该项目不缩小。



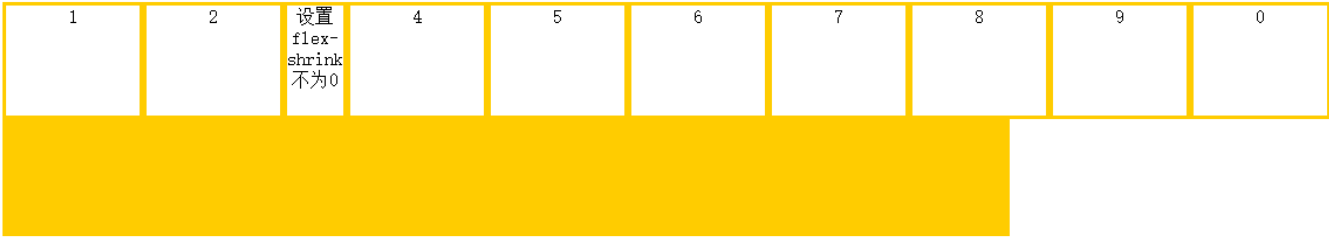
设置flex-shrink为0的项目不缩小

如果所有项目都为0，则当空间不足时，项目撑破容器而溢出。



不缩小

如果设置项目的flex-shrink不为0的非负数效果同设置为1。

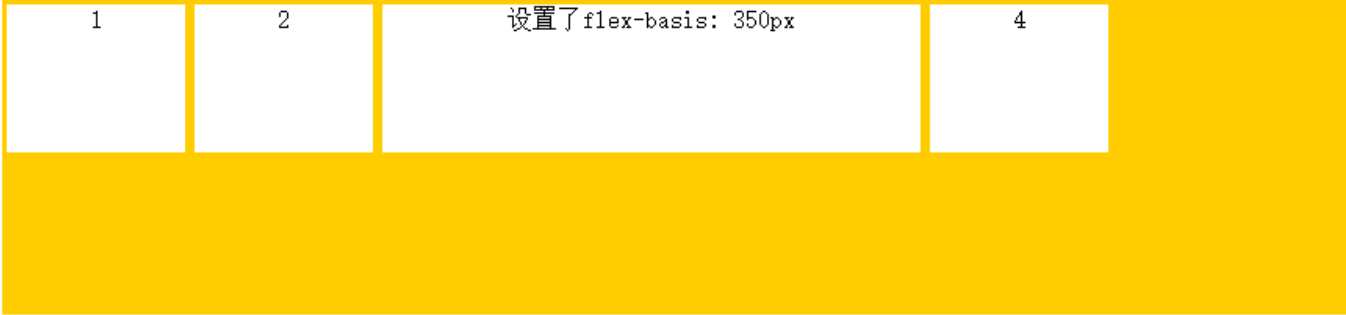


flex-basis属性：定义在分配多余空间之前，项目占据的主轴空间。

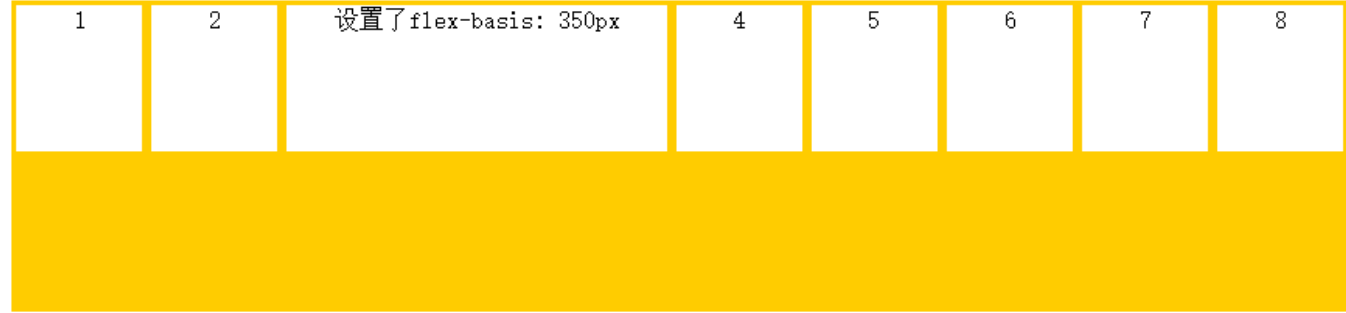
默认值为auto，浏览器根据此属性检查主轴是否有多余空间。

```
.item{
  flex-basis:  <auto或者px>;
}
```

注意设置的flex-basis是分配多余空间之前项目占据的主轴空间，如果空间不足则默认情况下该项目也会缩小。



设置flex-basis为350px，但空间充足



空间不足，项目缩小，小于设定值

flex属性是flex-grow，flex-shrink和flex-basis的简写

默认值为0 1 auto，第一个属性必须，后两个属性可选。

```
.item{
  flex: none | [<flex-grow><flex-shrink><flex-basis>];
}
```

- 可以用 `flex:auto;` 代替 `flex: 1 1 auto;` ;
- 可以用 `flex: none;` 代替 `flex: 0 0 auto;` ;

####align-self属性：允许单个项目与其他项目有不一样的对齐方式
>默认为auto，表示继承父元素的align-items属性，并可以覆盖align-items属性。

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