来源：https://www.cnblogs.com/jerehedu/p/7358329.html

三个基本概念：

①弹性布局：元素排列不再遵循常规文档流，采用弹性布局属性进行排列

②容器： 需要添加弹性布局的父元素

③项目： 弹性布局容器中的每一个子元素，称为项目

两个基本方向：

①主轴：通过属性规定水平/垂直方向为主轴，元素排列的方向

②交叉轴：与主轴垂直的另一个方向

# 一.弹性布局的使用

①为某一元素添加地display：flex/inline-flex属性即可使容器内容采用弹性布局

display：flex 容器添加该弹性布局后，显示为块级元素

display：inline-flex 容器添加该弹性布局后，显示为行级元素

②注意：

1.容器添加弹性布局后，仅仅是容器内的内容采用弹性布局，而容器本身在页面的定位方式依然遵循常规文档流

2.当一个父元素被设置弹性布局后，其子元素的float、clear和vertical-align属性将失效，但position属性仍然有效

# 二.可以为容器(父元素)添加的6大属性：

## 1.flex-direction：决定主轴的方向（即项目的排列方向）

row(默认值) 主轴为水平方向，起点在左端

row-reverse 主轴在水平方向，起点在右端

column 主轴为垂直方向，起点在上沿

column-reverse 主轴为垂直方向，起点在下沿

## 2.flex-wrap：如果一条轴线排不下，如何换行

nowrap(默认) 不换行。当容器宽度不够时，每个项目会被挤压

wrap 换行，并且第一行在容器最上方

wrap-reverse 换行，并且第一行在容器最下方

## 3.flex-flow：row nowrap (默认)

上面两个的缩写

## 4.justify-content：定义项目在主轴上的对齐方式

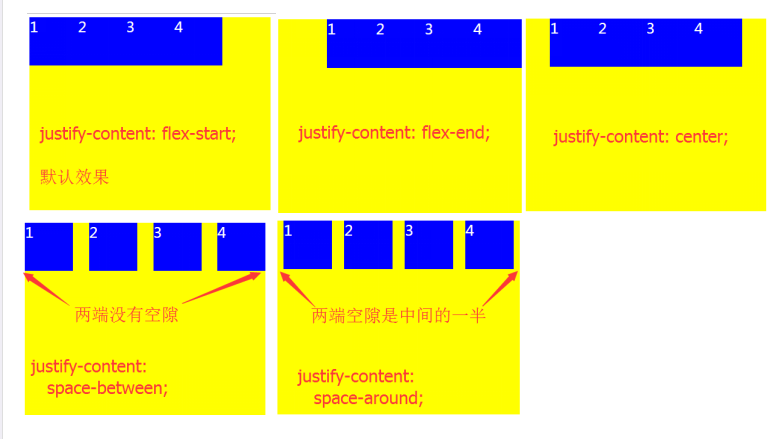
flex-start(默认值) 项目位于主轴起点

flex-end 项目位于主轴终点

center 项目居中

space-between 两端对齐，项目之间的间隔相等，开头和结尾的项目也父容器没有间隙

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



## 5.align-items：定义项目在交叉轴上如何对齐

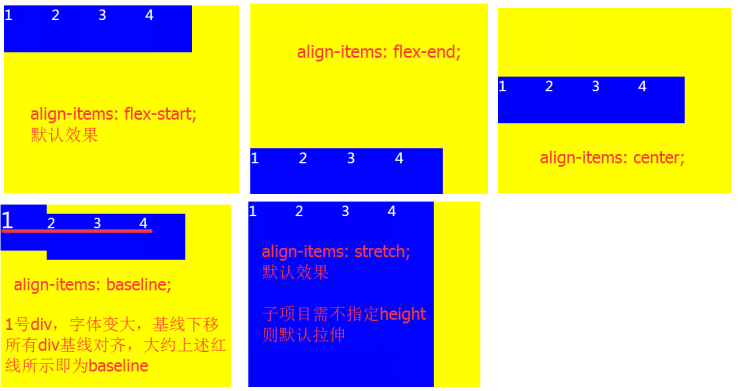
flex-start(默认值) 交叉轴起点对齐

flex-end 交叉轴终点对齐

center 交叉轴的中点对齐

baseline 以项目的第一行文字的基线对齐(该行文字的行高、大小将会影响基线，进而影响项目的对齐)

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



## 6.align-content：定义多根轴线的对齐方式

注意：①如果项目只用一根轴线，该属性不起作用

②当项目为多行时，可使用align-content取代align-items

flex-start：与交叉轴的起点对齐

flex-end：与交叉轴的终点对齐。

center：与交叉轴的中点对齐。

space-between：与交叉轴两端对齐，轴线之间的间隔平均分布。

space-around：每根轴线两侧的间隔都相等。所以，轴线之间的间隔比轴线与边框的间隔大一倍。

stretch（默认值）：轴线占满整个交叉轴。

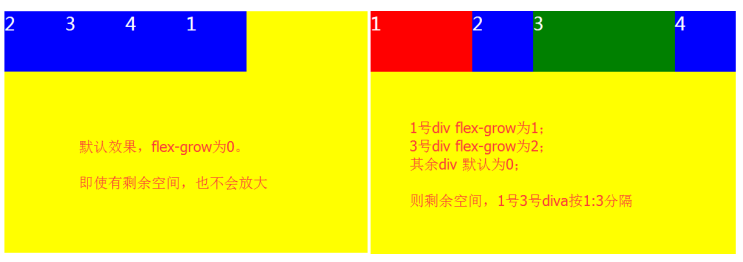
# 三.可以为子元素添加的6大属性

## 1.order：定义项目的排列顺序

数值越小，越靠近起点，默认为0

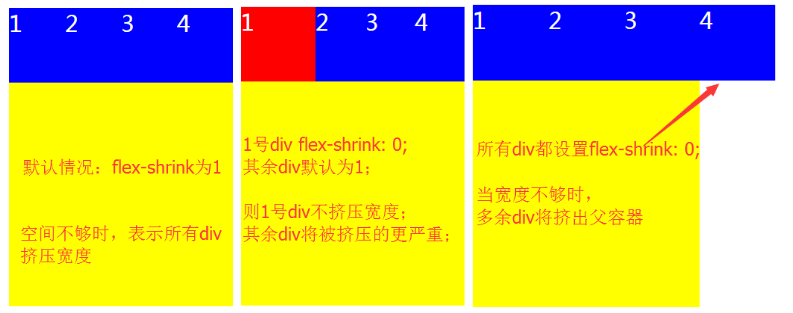
## 2.flex-grow：定义项目的放大比例

默认为0，即如果存在剩余空间，也不放大



## 3.flex-shrink：定义项目的缩小比例

默认为1，即空间不足时，该项目将缩小。



## 4.flex-basis：定义项目占据的主轴空间

注意：如果主轴为水平，则设置这个属性，相当于设置项目的宽度。原width将失效

## 5.flex：flex-grow，flex-shrink，flex-basis的缩写

默认值：0 1 auto

两个快捷设置： auto表示1 1 auto none表示0 0 auto

## 6.align-self：定义单个项目在交叉轴上的排列方式

可覆盖掉容器上的align-items属性

属性值：与align-items相同，默认为auto，表示继承父容器的align-items属性值