

弹性布局属性小结

日期 2017-7-20

张笑

采用Flex布局的元素称为Flex容器，简称容器。它的所有子元素自动称为容器成员，称为Flex项目，简称项目。

弹性布局主要思想就是让容器有能力让其子项目能够改变其宽度，高度，弹性布局方向不可预知，可以自定义。

弹性布局最适合应用程序的组件和小规模的布局，网格布局适合那些更大规模的布局。

1. 常用容器属性

flexDirection、justifyContent、alignItems、flexWrap

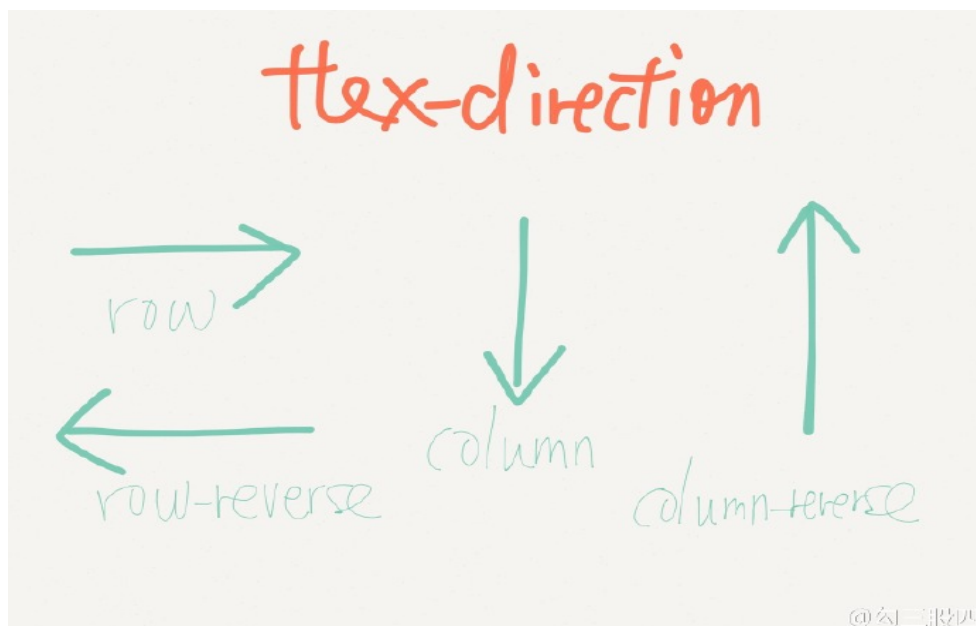
2. 常用元素属性

alignSelf、flex、position

3. 单独介绍

flexDirection: 该属性通过定义flex容器的主轴方向来决定flex子项在flex容器的位置，这将决定flex需要如何进行排列。

- row 主轴方向为水平，起点在左端。
- row- reverse 主轴方向为水平，起点在右端
- column(默认) 主轴方向为垂直，起点在上端
- column-reverse 主轴方向为垂直，起点在下端



justifyContent:该属性确定了组件在主轴方向上的对齐方式。

- **flex-start(默认)** 组件沿着主轴方向的起始位置靠齐。

如：假如主轴方向是 **row** 的话就是左对齐，是 **row-reverse** 的话就是右对齐，是 **column** 的话就是上对齐，是 **column-reverse** 的话就是下对齐。

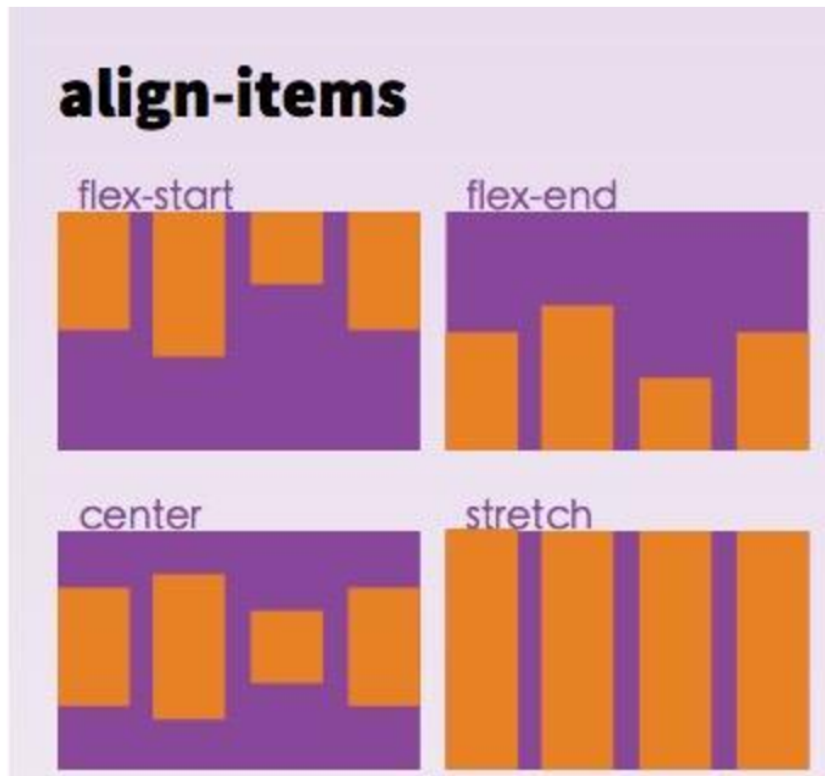
- **flex-end** 组件沿着主轴方向的结束位置靠齐，和 **flex-start** 相反。
- **space-between** 组件在主轴方向上两端对齐，其中的间隔相等。
- **space-around** 组件会平均分配在主轴方向上，两端保留一定的位置空间。

以主轴为横轴为例（**row**）



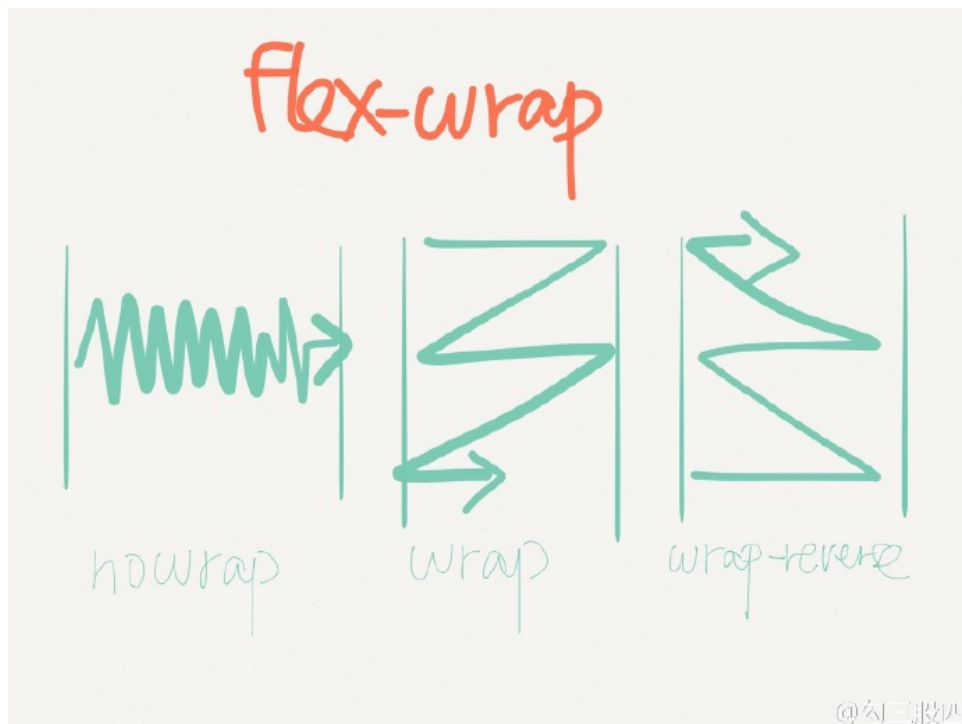
alignItems:组件在侧轴方向上的对齐方式。

- **flex-start** 组件沿着侧轴上的起点对齐
- **flex-end** 组件沿着侧轴上的终点对齐
- **center** 组件在侧轴方向上居中对齐
- **stretch(默认)** 组件在侧轴方向上占满



flexWrap: 是否换行

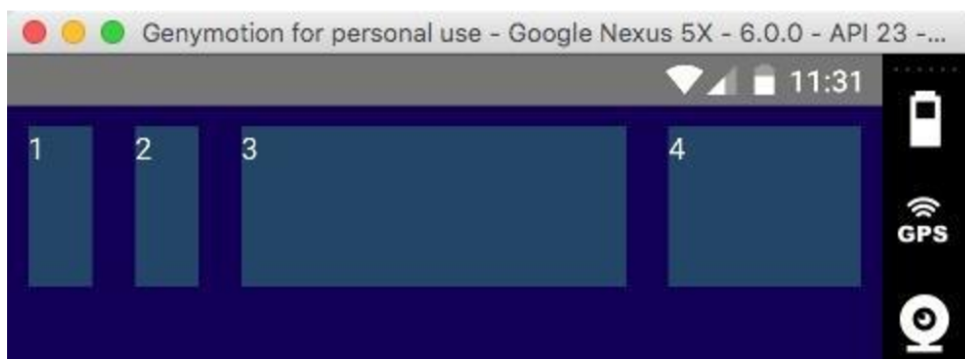
- 默认情况下，项目都排列在一条线上，放不下的部分则不放置，flexWrap 就是定义是否换行的。
- nowrap(默认) 不换行
- wrap 换行，第一行在上方
- wrap-reverse 换行，第一行在下方



flex:有三个参数，默认参数为 0 1 auto。 第一个参数为 flex-grow,第二，第三个为:flex-shrink 和 flex-basis。

例子：假如 1 和 2 为固定宽度为 30，

$3Width = (sumWidth - 30 * 2) * 3flexNum / (3flexNum + 4flexNum)$



alignSelf:定义单个组件自己的对齐方式，默认为 auto。:

```
// flexDirection: row
```

```
alignSelf
```

```
enum(
```

```
  'auto',
```

```
  'flex-start',
```

```
  'flex-end',
```

```
  'center',
```

```
  'stretch'
```

```
)
```

flex-start



alignSelf: flex-end

文档参考地址链接: <https://zhuanlan.zhihu.com/p/21322277?refer=fyales>