

1: 静态布局

2: 流式布局

二、流式布局(Liquid Layout)

1. 布局概念

流式布局也叫百分比布局

这边引入一下**自适应布局**:

分别为不同的屏幕设置布局格式, 当屏幕大小改变时, 会出现不同的布局, 意思就是在这个屏幕下这个元素块在这个地方, 但是在那个屏幕下, 这个元素块又会出现那个地方。只是布局改变, 元素不变。可以看成是不同屏幕下由多个静态布局组成的。

而流式布局的特点是随着屏幕的改变, 页面的布局没有发生大的变化, 可以进行适配调整, 这个正好与自适应布局互补。

流式布局常用的设计模板:

左侧固定+右侧自适应

左右固定宽度+中间自适应(参考[京东手机版](#))

页面元素的宽度按照屏幕进行适配调整, 主要的问题是如果屏幕尺度跨度太大, 那么在相对其原始设计而言过小或过大的屏幕上不能正常显示。你看到的页面, 元素的大小会变化而位置不会变化——这就导致如果屏幕太大或者太小都会导致元素无法正常显示。

2. 优点

元素的宽高用百分比做单位, 元素宽高按屏幕分辨率调整, 布局不发生变化

3. 缺点

屏幕尺度跨度过大的情况下, 页面不能正常显示。

三、弹性布局(Flex Layout)

1. 布局概念

弹性布局是CSS3引入的强大的布局方式, 用来替代以前Web开发人员使的一些复杂易错的 hacks方法(如float实现流式布局)。

2. 优点

简单、方便、快速

3. 缺点

CSS3新特性,浏览器兼容性非常头疼。而且手机浏览器对flex的支持也不是很理想。

4. 实现方法

flex-flow: ||

flex-direction和flex-wrap的简写,在两者选其一。

flex-direction: row (初始值) | row-reverse | column | column-reverse

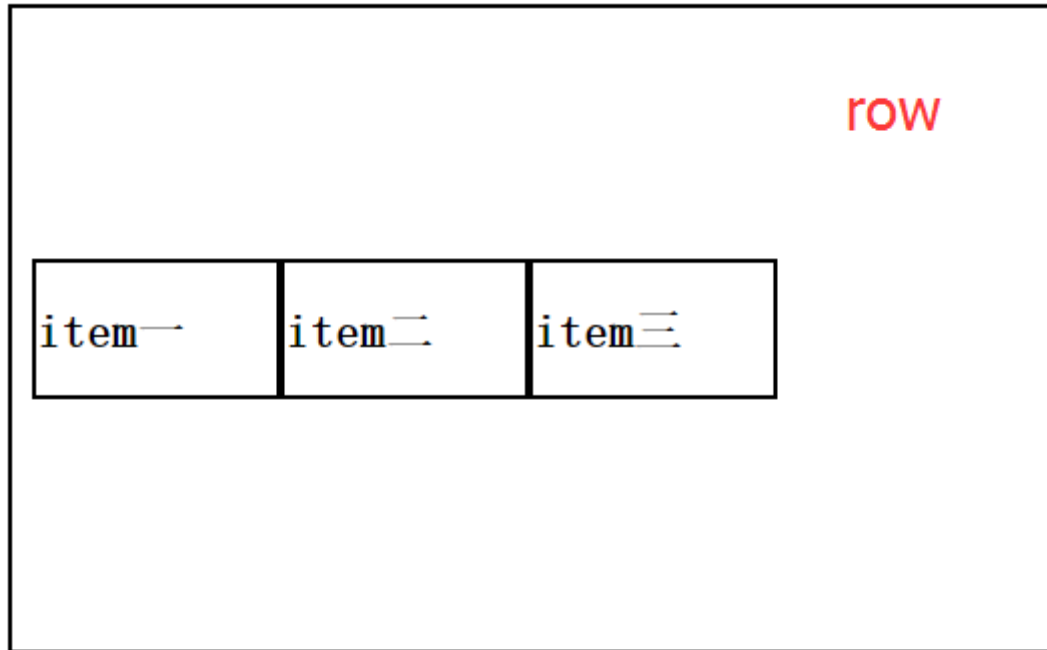
flex-direction定义了弹性项目在弹性容器中的放置方向，默认是row，即行内方向（一般而言是由左往右，但注意这个和书写模式有关）。

row: 主轴为水平方向，项目沿主轴从左至右排列

column: 主轴为竖直方向，项目沿主轴从上至下排列

row-reverse: 主轴水平，项目从右至左排列，与row反向

column-reverse: 主轴竖直，项目从下至上排列，与column反向



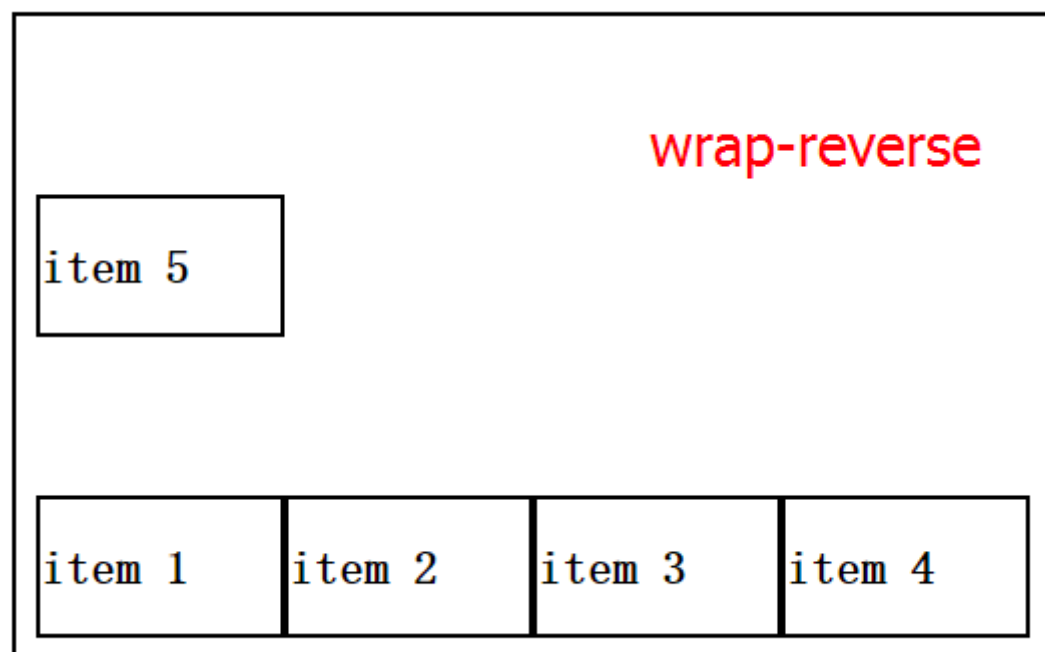
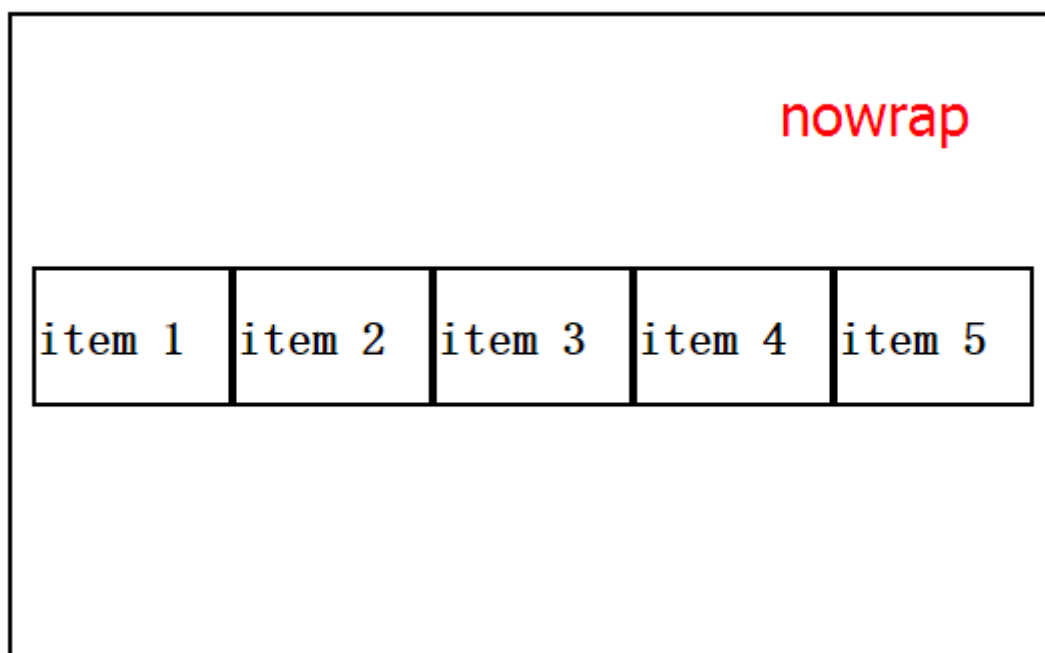
flex-wrap: nowrap (初始值) | wrap | wrap-reverse

flex-wrap定义是否需要拆行以使得弹性项目能被容器包含。*-reverse代表相反的方向。

nowrap: 自动缩小项目，不换行

wrap: 换行，且第一行在上方

wrap-reverse: 换行，第一行在下面



两者结合起来即flex-flow属性就确定了弹性容器在main axis和cross axis两个方向上的显示方式

自适应布局（Adaptive）的特点是分别为不同的屏幕分辨率定义布局。布局切换时页面元素发生改变，但在每个布局中，页面元素不随窗口大小的调整发生变化。你可以把自适应布局看作是静态布局的一个系列。就是说你看到的页面，里面元素的位置会变化而大小不会变化。

四、响应式布局(Responsive layout)

采用自适应布局和流式布局的综合方式，为不同屏幕分辨率范围创建**流式布局**
现在优秀的页面都追求一套代码可以实现三端的浏览；

从概念可以看出来,自适应布局的诞生是为了实现不同屏幕分辨率的终端上浏览网页的不同展示方式。

通过响应式设计能使网站在手机和平板电脑上有更好的浏览阅读体验。屏幕尺寸不一样展示给用户的网页内容也不一样.

利用**媒体查询**可以检测到屏幕的尺寸(主要检测宽度),并设置不同的CSS样式,就可以实现响应式的布局。

大名鼎鼎的bootstrap就是响应式布局的专家。

官方放出狠话:

Bootstrap 提供了一套响应式、移动设备优先的流式栅格系统,随着屏幕或视口 (viewport) 尺寸的增加,系统会自动分为最多12列。它包含了易于使用的预定义类,还有强大的mixin 用于生成更具语义的布局。