# 响应式布局

bootstrap的栅格系统就是响应式布局

## 第一步：html页面的设置

1、设置<meta>标签

1)设置html页面的编码

**<meta charset="utf-8">**

2)声明文档兼容模式，表示使用IE浏览器的最新模式

**<meta http-equiv="X-UA-Compatible" content="IE=edge">**

3)设置视口的宽度(值为设备的理想宽度)，页面初始缩放值<理想宽度/可见宽度>

视口的作用：在移动浏览器中，当页面宽度超出设备，浏览器内部虚拟的一个页面容器，会将页面缩放到设备这么大来展示

content属性的参数如下：

width 设置layout viewport 的宽度，为一个正整数，或字符串"width-device"(表示采用设备的宽度)

initial-scale 设置页面的初始缩放值，为一个数字，可以带小数

minimum-scale 允许用户的最小缩放值，为一个数字，可以带小数

maximum-scale 允许用户的最大缩放值，为一个数字，可以带小数

height 设置layout viewport 的高度，这个属性对我们并不重要，很少使用

user-scalable 是否允许用户进行缩放，值为"no"或"yes", no 代表不允许，yes代表允许

如果设置"user-scalable=no",那么"minimum-scale"和"maximum-scale"无效

**<meta name="viewport" content="width=device-width, initial-scale=1">**

上述3个meta标签\*必须\*放在最前面，任何其他内容都\*必须\*跟随其后！

2、引入jquery和bootstrap的css与js

引入Bootstrap核心样式文件

<link href="../../css/bootstrap.min.css" rel="stylesheet">

引入js

前端开发建议：网站优化时，除了立即需要工作的js存放在head外，将大部分JS文件放在页面的末尾

<!-- 引入jQuery核心js文件，必须放置在bootStrap.js前面！ -->

<script src="../../js/jquery-1.11.0.min.js"></script>

<!-- 引入BootStrap核心js文件 -->

<script src="../../js/bootstrap.min.js"></script>

## 第二步 bootstrap布局容器

Bootstrap 需要为页面内容和栅格系统包裹一个 .container 容器。和.container-fluid

我们提供了两个作此用处的类。注意，由于 padding 等属性的原因，这两种 容器类不能互相嵌套。

使用步骤

html

<div class="container">

这里是内容

</div>

css样式：为了让其看到效果，为此class添加样式：边框颜色

<style>

.container{ 可直接用它的容器样式添加边框

border: 1px solid red;

}

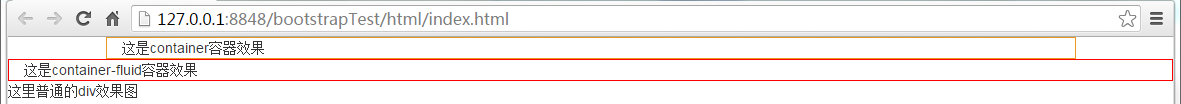
. container-fluid{ 可直接用它的容器样式添加边框

border: 1px solid red;

}

</style>

两个容器效果主要区别



## 第三步 栅格系统

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。

栅格系统用于通过一系列的行（row）与列（column）的组合来创建页面布局，你的内容就可以放入这些创建好的布局中。

### 工作原理

1. 行（row）：必须包含在 .container （固定宽度）或 .container-fluid （100% 宽度）中，以便为其赋予合适的排列（aligment）和内补（padding）。
2. 通过“行（row）”在水平方向创建一组“列（column）”
3. 你的内容应当放置于“列（column）”内，并且，只有“列（column）”可以作为行（row）”的直接子元素
4. 通过为“列（column）”设置 padding 属性，从而创建列与列之间的间隔（gutter）。通过为 .row 元素设置负值 margin 从而抵消掉为 .container 元素设置的 padding，也就间接为“行（row）”所包含的“列（column）”抵消掉了padding。
5. 栅格系统中的列是通过指定1到12的值来表示其跨越的范围。例如，三个等宽的列可以使用三个 .col-xs-4 来创建

参数



### 使用步骤

xml定义

设定4种不同屏幕的div显示布局

<body>

<!--第一步 需要布局容器支持 -->

<div class="container">

<!--第二步 需要行row -->

<div class="row">

<!--第三步 col-lg最大屏幕，col-md中屏幕，col-sm小屏幕 col-xs最小屏幕 -->

<div class="">.col-md-1</div>

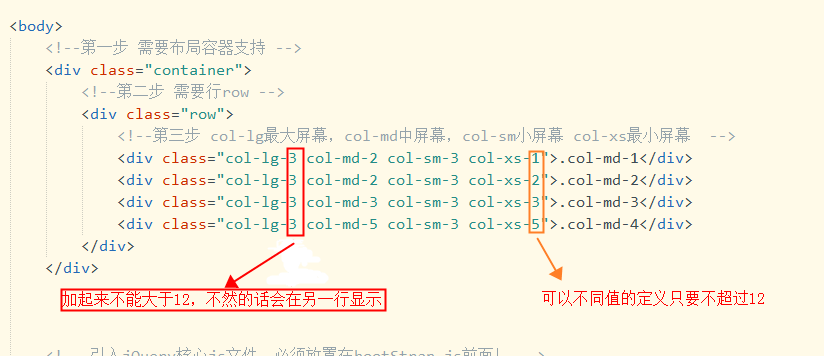
<div class="col-lg-3 col-md-2 col-sm-3 col-xs-2">.col-md-2</div>

<div class="col-lg-3 col-md-3 col-sm-3 col-xs-3">.col-md-3</div>

<div class="col-lg-3 col-md-5 col-sm-3 col-xs-5">.col-md-4</div>

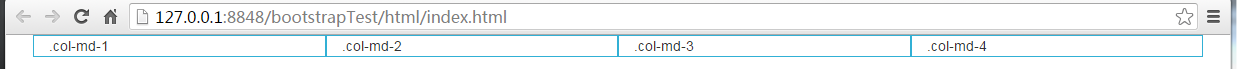
</div>

</div>

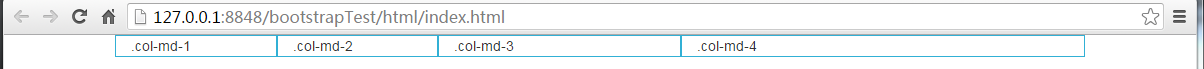
</body>

效果图

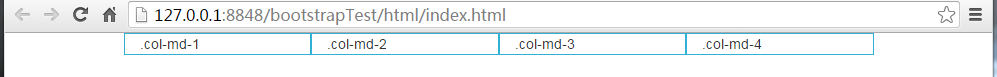
屏幕最大也就是col-lg-3显示



屏幕中也就是col-md显示



屏幕小也就是col-sm显示



屏幕最小也就是col-xs显示

