

[笔记][Bootstrap666][01-Bootstrap栅格系统]

前端

[笔记][Bootstrap666][01-Bootstrap栅格系统]

- 01. 课程对应代码资料下载
- 02. Bootstrap开篇
- 03. Bootstrap3模板
- 04. Bootstrap4模板
- 05. Bootstrap容器
- 06. Bootstrap容器实现原理
- 07. Bootstrap栅格系统基本使用
- 08. Bootstrap栅格系统实现原理
- 09. Bootstrap栅格系统尺寸设置
- 10. Bootstrap栅格系统对齐方式
- 11. Bootstrap栅格系统偏移和排序

01. 课程对应代码资料下载

略

02. Bootstrap开篇

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>01-Bootstrap开篇</title>
```

```
</head>
<body>
<script>
  /*
    1.什么是Bootstrap?
    Bootstrap 是twitter公司推出的，专门用于开发响应式布局、移动设备优先的 WEB 框架。
    Bootstrap当前最新的版本的Bootstrap4，但当下企业使用最多的是Bootstrap3

    2.Bootstrap3和4的区别
    2.1CSS预处理器不同，Bootstrap3采用Less，Bootstrap4采用SASS
    2.2格栅种类不同，Bootstrap3提供4种格栅，Bootstrap4提供5种格栅
    2.3使用单位不同，Bootstrap3使用px作为单位，Bootstrap4使用rem作为单位
    2.4内部布局方式不同，Bootstrap3使用float布局，Bootstrap4使用flexbox布局
    2.5 ... ...

    3.Bootstrap兼容性
    Bootstrap 的目标是在最新的桌面和移动浏览器上有最佳的表现，也就是说，在较老旧的浏览器上可能会导致某些组件表现出的样式有些不同，但是功能是完整的。
    IE8以上都能完美使用，IE8以下需要通过一些额外的配置来保证其完整性
    */
</script>
</body>
</html>
```

03. Bootstrap3模板

- 实现响应式网站都要通过媒体查询，而媒体查询是 **CSS3** 中推出的，所以不支持低版本的浏览器。
- **Respond.js** 不能在文件中打开，在 **IE8** 及以下是无效的。

- 建议编写任何网页都加上 `<meta http-equiv="X-UA-Compatible" content="IE=edge">`

```
<!DOCTYPE html>
<html lang="zh-CN">
<head>
  <meta charset="utf-8">
  <!-- 可以让部分国产浏览器默认采用高速模式渲染页面 -->
  <meta name="renderer" content="webkit">
  <!-- 为了让 IE 浏览器运行最新的渲染模式下 -->
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <!-- 为了保证在移动端能够正常的显示 -->
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>自己网页的标题</title>
  <!-- 导入Bootstrap的CSS文件 -->
  <link rel="stylesheet" href="css/bootstrap.css">

  <!-- 导入respond.js文件的目的，是为了能够在IE8以及IE8以下的浏览器中使用媒体查询 -->
  <!-- 导入html5shiv.js文件的目的，是为了能够在IE8以及IE8以下的浏览器中使用H5新增的标签 -->
  <!--
  [if xxx] ![endif] 这个是IE中的条件注释，只有在IE浏览器下才会执行
  以下代码的含义：如果当前是IE9以下的浏览器，那么就导入以下的两个JS文件
  -->
  <!--[if lt IE 9]>
    <script src="js/html5shiv.js"></script>
    <script src="js/respond.js"></script>
  <![endif]-->
</head>
<body>
  <!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery, 所以必须放在前边) -->
  <script src="js/jquery-1.12.4.js"></script>
  <!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只加载单个插件。 -->
  <script src="js/bootstrap.js"></script>
```

```
</body>
</html>
```

04. Bootstrap4模板

```
<!doctype html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <!--为了保证在移动端能够正常的显示-->
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- 导入Bootstrap的CSS文件 -->
    <link rel="stylesheet" href="css/bootstrap.css">
    <title>自己网页的标题</title>
</head>
<body>

    <!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery, 所以必须放在前边) -->
    <script src="js/jquery-3.1.1.js"></script>
    <!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的, 所以需要导入-->
    <script src="js/popper.js"></script>
    <!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只加载单个插件。 -->
    <script src="js/bootstrap.js"></script>
</body>
</html>
```

05. Bootstrap容器

```

<!doctype html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <!--为了保证在移动端能够正常的显示-->
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <!-- 导入Bootstrap的CSS文件 -->
    <link rel="stylesheet" href="css/bootstrap.css">
    <title>04-Bootstrap容器</title>
    <style>
        *{
            margin: 0;
            padding: 0;
        }
        div{
            height: 100px;
            background: red;
        }
    </style>
</head>
<body>
<div class="container"></div>
<!--

```

1. Bootstrap容器

1.1在Bootstrap中容器是响应式布局的基础，Bootstrap推荐将所有内容都定义在容器之中

1.2并且容器是启用Bootstrap栅格系统必不可少的前置条件

2. Bootstrap中提供了两个容器：container/container-fluid

2.1container容器(固定容器)：

只要给标签添加了container类名，这个标签就变成了Bootstrap的container容器

只要这个标签变成了Bootstrap的container容器，在不同视口大小下就会有不同的固定宽度

2.2container-fluid(自适应宽度容器)

只要给标签添加了container-fluid类名，这个标签就变成了Bootstrap的container-fluid容器

只要这个标签变成了Bootstrap的container-fluid容器，那么容器的宽度永远都是100%自适应

3. Bootstrap对视口划分

Bootstrap4将视口划分为了5种

超小屏幕(xs): <576px

小屏幕(sm): >=576px

中等屏幕(md): >=768

大屏幕(lg): >=992px

超大屏幕(xl): >= 1200px

-->

<!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery, 所以必须放在前边) -->

<script src="js/jquery-3.1.1.js"></script>

<!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的, 所以需要导入-->

<script src="js/popper.js"></script>

<!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只加载单个插件。 -->

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

</body>

</html>

06. Bootstrap容器实现原理

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>05-Bootstrap容器实现</title>
  <style>
    *{
      margin: 0;
      padding: 0;
    }
    .container {
      width: 100%;
```

```
        padding-right: 15px;
        padding-left: 15px;
        margin-right: auto;
        margin-left: auto;
    }

    @media (min-width: 576px) {
        .container {
            max-width: 540px;
        }
    }

    @media (min-width: 768px) {
        .container {
            max-width: 720px;
        }
    }

    @media (min-width: 992px) {
        .container {
            max-width: 960px;
        }
    }

    @media (min-width: 1200px) {
        .container {
            max-width: 1140px;
        }
    }

    div{
        height: 100px;
        background: red;
    }

</style>
</head>
<body>
<div class="container"></div>
</body>
</html>
```

07. Bootstrap栅格系统基本使用

```
<!doctype html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <!--为了保证在移动端能够正常的显示-->
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <!-- 导入Bootstrap的CSS文件 -->
  <link rel="stylesheet" href="css/bootstrap.css">
  <title>06-Bootstrap栅格系统</title>
  <style>
    *{
      margin: 0;
      padding: 0;
    }
    body>div{
      height: 100px;
      background: red;
    }
    .container>.row{
      height: 100%;
      background: blue;
    }
    .row>div{
      background: skyblue;
    }
    .row>div:nth-child(2){
      background: deeppink;
    }
    .row>div:nth-child(3){
      background: orangered;
    }
  </style>
</head>
<body>
<div class="container">
  <span class="row">
```



```

        <div class="col-6">我是第1列</div>
        <div class="col-4">我是第2列</div>
        <div class="col-6">我是第3列</div>
    </span>
</div>
<!--
    1.Bootstrap栅格系统
    Bootstrap的栅格系统使用一系列的"行"和"列"来实现复杂的响应式布局

    默认情况下栅格系统会将一行的内容等分为12份，
    然后我们可以通过绑定类名的方式指定这一行中的每一列占用多少份

    2.Bootstrap栅格系统格式
    <容器>
        <行>
            <列>我们的内容</列>
            <列>我们的内容</列>
            ... ..
        </行>
    </容器>

    3.Bootstrap栅格系统特点
    3.1默认情况下行的宽度和所在容器一样
    3.2默认情况下所有列的宽度是等分所在行的宽度
    3.3可以通过col-num方式指定当前列占用12分之几
    3.4如果一行中内容的宽度超过了12分，那么将自动换行
-->
<!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery,
所以必须放在前边) -->
<script src="js/jquery-3.1.1.js"></script>
<!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的，
所以需要导入-->
<script src="js/popper.js"></script>
<!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只
加载单个插件。 -->
<script src="js/bootstrap.js"></script>
</body>
</html>

```

- 容器里面的一行有默认的宽度，和容器宽度一样
- 添加了 `.col` 之后，列的宽度就不是由内容撑起了，而是等分一行的宽度

- 如果所有列加起来的宽度超过 12，同一行的内容会自动换行

08. Bootstrap栅格系统实现原理

```
<!doctype html>
<html>
<head>
  <meta charset="utf-8">
  <title>07-Bootstrap栅格系统实现</title>
  <style>
    *{
      margin: 0;
      padding: 0;
    }
    .container {
      width: 100%;
      padding-right: 15px;
      padding-left: 15px;
      margin-right: auto;
      margin-left: auto;
    }

    @media (min-width: 576px) {
      .container {
        max-width: 540px;
      }
    }

    @media (min-width: 768px) {
      .container {
        max-width: 720px;
      }
    }

    @media (min-width: 992px) {
      .container {
        max-width: 960px;
      }
    }
  </style>
</head>
</html>
```

```

    }

    @media (min-width: 1200px) {
        .container {
            max-width: 1140px;
        }
    }
    div{
        height: 100px;
        background: red;
    }
    .row{
        display: flex;
        flex-wrap: wrap;
        margin-left: -15px;
        margin-right: -15px;
    }
    .container>.row{
        height: 100%;
        background: blue;
    }

    .col{
        flex: 1;
    }
    .row>div{
        background: skyblue;
    }
    .row>div:nth-child(2){
        background: deeppink;
    }
    .row>div:nth-child(3){
        background: orangered;
    }
    .col-6{
        flex-basis: 50%;
    }
</style>
</head>
<body>
<div class="container">

```

```

    <span class="row">
      <div class="col-6">我是第1列</div>
      <div class="col">我是第2列</div>
      <div class="col">我是第3列</div>
    </span>
  </div>
</body>
</html>

```

- `flex: 0 0 50%;` 是连写格式，放大的比例，缩小的比例，当前伸缩项宽度
- 之所以列要放到行里面，是因为行是伸缩容器，列是伸缩项
- 3 版本把列变成浮动的，然后通过设置宽度实现。

09. Bootstrap栅格系统尺寸设置

```

<!doctype html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <!-- 为了保证在移动端能够正常的显示 -->
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <!-- 导入Bootstrap的CSS文件 -->
  <link rel="stylesheet" href="css/bootstrap.css">
  <title>08-Bootstrap栅格系统</title>
  <style>
    *{
      margin: 0;
      padding: 0;
    }
    body>div{
      height: 100px;
      background: red;
    }
    .container>.row{

```

```

        height: 100%;
        background: blue;
    }
    .row>div{
        background: skyblue;
    }
    .row>div:nth-child(2){
        background: deeppink;
    }
    .row>div:nth-child(3){
        background: orangered;
    }
</style>
</head>
<body>
<div class="container">
    <span class="row">
        <!--<div class="col-6">我是第1列</div>
        <div class="col-4">我是第2列</div>
        <div class="col-2">我是第3列</div>-->
        <!--<div class="col-xl-6">我是第1列</div>
        <div class="col-xl-4">我是第2列</div>
        <div class="col-xl-2">我是第3列</div>-->
        <div class="col-lg-2 col-xl-6">我是第1列</div>
        <div class="col-lg-4 col-xl-4">我是第2列</div>
        <div class="col-lg-6 col-xl-2">我是第3列</div>
    </span>
</div>
<!--

```

1. 栅格系统列的尺寸设置

1.1 Bootstrap的固定宽度容器提供了5种响应的尺寸，
同样Bootstrap也给列提供了5种响应的尺寸

col-*: 设置超小屏幕

col-sm-*: 设置小屏幕

col-md-*: 设置中等屏幕

col-lg-*: 设置大屏幕

col-xl-*: 设置超大屏幕

2. 栅格系统列的尺寸特点

1. 如果只设置了小屏幕的大小，那么大屏幕也会采用小屏幕设置的大小
2. 如果值设置了大屏幕的大小，那么小屏幕默认100%

```
3. 如果大小屏幕都设置了大小，那么在什么屏幕就显示什么尺寸
-->
<!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery,
所以必须放在前边) -->
<script src="js/jquery-3.1.1.js"></script>
<!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的,
所以需要导入-->
<script src="js/popper.js"></script>
<!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只
加载单个插件。 -->
<script src="js/bootstrap.js"></script>
</body>
</html>
```

10. Bootstrap栅格系统对齐方式

```
<!doctype html>
<html lang="en">
<head>
  <!-- Required meta tags -->
  <meta charset="utf-8">
  <!--为了保证在移动端能够正常的显示-->
  <meta name="viewport" content="width=device-width, in
itial-scale=1, shrink-to-fit=no">
  <!-- 导入Bootstrap的CSS文件 -->
  <link rel="stylesheet" href="css/bootstrap.css">
  <title>09-Bootstrap栅格系统</title>
  <style>
    *{
      margin: 0;
      padding: 0;
    }
    body>div{
      height: 100px;
      background: red;
    }
    .container>.row{
```

```

        height: 100%;
        background: blue;
    }
    .row>div{
        background: skyblue;
    }
    .row>div:nth-child(2){
        background: deeppink;
    }
    .row>div:nth-child(3){
        background: orangered;
    }
</style>
</head>
<body>
<!--<div class="container px-0">
    <span class="row no-gutters">
        <div class="col-6">我是第1列</div>
        <div class="col-2">我是第2列</div>
        <div class="col-2">我是第3列</div>
    </span>
</div>-->

<div class="container">
    <!--<span class="row justify-content-between">-->
    <span class="row align-items-center">
        <div class="col-6">我是第1列</div>
        <div class="col-2">我是第2列</div>
        <div class="col-2">我是第3列</div>
    </span>
</div>
<!--

```

1. 栅格系统-沟槽

Bootstrap默认的栅格和列间有间隙沟槽，一般是左右-15px的margin或padding处理，您可以使用.no-gutters类来消除它，这将影响到.row行、列平行间隙及所有子列

2. 栅格系统列-对齐方式

Bootstrap4的格栅系统是使用伸缩布局实现的，所以也可以通过类名快速的设置伸缩项的在主轴和侧轴对齐方式

```
-->
```

```

<!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery,
所以必须放在前边) -->
<script src="js/jquery-3.1.1.js"></script>
<!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的,
所以需要导入-->
<script src="js/popper.js"></script>
<!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只
加载单个插件。 -->
<script src="js/bootstrap.js"></script>
</body>
</html>

```

- `no-gutters` 应用于行上面，可以取消伸缩项的 `15px` 的内边距。
- `px-0` 应用于容器上面，可以取消容器的 `15px` 的内边距。

主轴对齐方式可以通过类名设置：

- `justify-content-start`（默认）
- `justify-content-end`
- `justify-content-center`
- `justify-content-between`
- `justify-content-around`

侧轴对齐方式可以通过类名设置：

- `align-items-start`
- `align-items-end`
- `align-items-center`
- `align-items-baseline`
- `align-items-stretch`（默认）

11. Bootstrap栅格系统偏移和排序

```

<!doctype html>
<html lang="en">
<head>
    <!-- Required meta tags -->

```



```
<meta charset="utf-8">
<!--为了保证在移动端能够正常的显示-->
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<!-- 导入Bootstrap的CSS文件 -->
<link rel="stylesheet" href="css/bootstrap.css">
<title>10-Bootstrap栅格系统</title>
<style>
    *{
        margin: 0;
        padding: 0;
    }
    body>div{
        height: 100px;
        background: red;
    }
    .container>.row{
        height: 100%;
        background: blue;
    }
    .row>div{
        background: skyblue;
    }
    .row>div:nth-child(2){
        background: deeppink;
    }
    .row>div:nth-child(3){
        background: orangered;
    }
</style>
</head>
<body>
<!--
<div class="container">
    <span class="row justify-content-center">
        <div class="col-2">1</div>
        <div class="col-2">2</div>
        <div class="col-2">3</div>
    </span>
</div>
```

```

<div class="container" style="margin-top: 20px">
  <span class="row">
    <div class="col-2 offset-3">1</div>
    <div class="col-2">2</div>
    <div class="col-2">3</div>
  </span>
</div>
-->
<!--
<div class="container">
  <span class="row justify-content-between">
    <div class="col-2">1</div>
    <div class="col-2">2</div>
  </span>
</div>
<div class="container" style="margin-top: 20px">
  <span class="row">
    <div class="col-2">1</div>
    <div class="col-2 offset-8">2</div>
  </span>
</div>
-->
<div class="container">
  <span class="row">
    <div class="col-2 order-3">1</div>
    <div class="col-2 order-2">2</div>
    <div class="col-2 order-1">3</div>
  </span>
</div>

```

<!--

1. 栅格系统-列偏移

offset-*: 但前列偏移多少份位置

注意点: 写在那一列就是那一列偏移

2. 栅格系统列-列排序

order-*: 从小到大顺序排序, 小的在前面, 大的在后面

注意点: 没有排序的列位置不会发生变化, 只有排序的列才参与位置变

化

-->

```
<!-- jQuery (Bootstrap 的所有 JavaScript 插件都依赖 jQuery,
所以必须放在前边) -->
<script src="js/jquery-3.1.1.js"></script>
<!--在Bootstrap4中很多的提示/弹窗都是通过popper.min.js实现的,
所以需要导入-->
<script src="js/popper.js"></script>
<!-- 加载 Bootstrap 的所有 JavaScript 插件。你也可以根据需要只
加载单个插件。 -->
<script src="js/bootstrap.js"></script>
</body>
</html>
```

完成于 2019.9.12