# day30-前端【Bootstrap】

# 学习目标

01\_能够创建bootstrap的模板

02\_能够使用boostrap的两种布局容器

03\_能够理解bootstrap的响应式布局的特点

04\_能够查询文档创建bootstrap的按钮、表格、表单等常用组件

05\_能够理解bootstrap的栅格系统

06\_能够查询文档使用bootstrap的导航条

07\_能够查询文档使用bootstrap的轮播图

08\_能够利用Bootstrap完成黑马旅游的首页

# 项目清单

|  |
| --- |
|  |

# 回顾和提问

01\_/^\d+$/是什么意思？

|  |
| --- |
| 是一个JS中的正则表达式  匹配1个或以上的数字 |

02\_document.getElementById(“time”)是什么意思？

$(“#time”)

|  |
| --- |
| 定位到页面中，id属性值为time的标签/元素节点 |

# 问题引入

现在我们制作前端网页，用的是HTML、CSS、JavaScript这三个技术。

要制作出好看的网页，一是对HTML、CSS、JavaScript技术要强，

二是非常消耗时间，效率低，对于基础不好的我们来说，制作一个好看

的网页还是比较困难的。那么有没有一种现成可用的技术，直接拿来用，

且这个技术制作出来的网页比较美观，效率高呢，最后还能适应不同

设备的要求，显示出网页效果呢？

答案是：Bootstrap技术。

|  |
| --- |
|  |

# 第01章 Bootstrap的概述

## 1.1 Boostrap的简介

什么是框架：在现有的HTML、CSS、JavaScript语言基础上，由第三方厂商开发的一些功能组件，可以提高开发效率，降低开发难度。我们就是学会使用这些框架来开发程序即可。

什么是Bootstrap：它是一个简洁、直观、强悍的前端开发框架，让web开发更迅速、简单。

中文官网：www.bootcss.com。

和用Bootstrap写的网站页面。

|  |
| --- |
|  |

|  |
| --- |
|  |

## 1.2 Bootstrap的作用

1\_用于前端开发的框架。

2\_基于HTML5、CSS3和JavaScript技术。

3\_制作网站的页面。

## 1.3 Bootstrap的优势

1\_移动设备优先：自 Bootstrap 3 起，框架包含了贯穿于整个库的移动设备优先的样式。

2\_浏览器支持：Bootstrap支持所有的主流浏览器。如：Internet Explorer、 Firefox、 Opera、 Google Chrome、Safari。

3\_容易上手：只要您具备 HTML 和 CSS 的基础知识，您就可以开始学习 Bootstrap。

4\_响应式设计：Bootstrap 的响应式 CSS 能够自适应于台式机、平板电脑和手机。

## 1.4响应式设计

### 1.4.1传统的网页设计

淘宝网站在台式机上的效果：

|  |
| --- |
|  |

淘宝网站在手机上的效果：

|  |
| --- |
|  |

### 1.4.2 Bootstrap的响应式网页设计

苹果网站在台式机上的效果：

|  |
| --- |
|  |

苹果网站在手机上的效果：

|  |
| --- |
|  |

概念：一个网站能够兼容多个终端，而不是为每个终端做一个特定的版本，这个概念是为解决移动互联网浏览而诞生的。响应式布局可以为不同终端的用户提供更加舒适的界面和更好的用户体验。响应式布局基于HTML5和CSS3才可以实现。

响应式布局：一个网页可以自动适配设备的宽度，在手机，平板电脑，台式电脑上看到的是同一个页面。

注意：

传统方式，要定制多个网页，针对不同设备。

响应式方式，只有一个网页，针对不同设备。

# 第02章 Bootstrap的使用

## 2.1准备使用Bootstrap

### 2.1.1 Bootstrap包含的内容

1\_设置全局CSS样式：基本的HTML元素均可以通过class设置样式并得到增强效果。

2\_组件：无数可复用的组件，包括字体图标、下拉菜单、导航、警告框、弹出框等更多功能。

3\_JavaScript 插件：jQuery 插件为 Bootstrap的组件赋予了“活的生命”。

### 2.1.2 Bootstrap下载

<http://www.bootcss.com>，下载用于生产环境的Bootstrap即可。

|  |
| --- |
|  |

### 2.1.3 Bootstrap的目录结构

|  |
| --- |
|  |

|  |
| --- |
| bootstrap/  ├── css/ 全局的CSS样式，把所有的标签重新定义了样式。  │ ├── bootstrap.css 样式文件，标准版  │ ├── **bootstrap.min.css 压缩版，我们开发的时候使用压缩版**  │ ├── bootstrap-theme.css   │ ├── bootstrap-theme.min.css   ├── js/ JavaScript的插件目录。  │ ├── bootstrap.js JS文件，标准版  │ └── **bootstrap.min.js 压缩版，我们开发的时候使用压缩版**  └── fonts/ 字体图标，用于在网页上使用各种BootStrap提供的小图标，如果不用，这个文件夹可以不复制。  ├── glyphicons-halflings-regular.eot  ├── glyphicons-halflings-regular.svg  ├── glyphicons-halflings-regular.ttf  ├── glyphicons-halflings-regular.woff  └── glyphicons-halflings-regular.woff2 |

### 2.1.4压缩版与标准版的区别

压缩版与标准版的功能是一样的，压缩版文件更小，更省资源，加载速度更快。但是没有换行，没有注释，变量名都很短。

### 2.1.5目录下的字体图标

|  |
| --- |
|  |

## 2.2创建Bootstrap模板

效果：

|  |
| --- |
|  |

### 2.2.1创建Bootstrap模板文件的步骤

1\_复制bootstrap-3.3.7-dist.zip解压出来的三个文件夹css、js、fonts到项目中。

2\_复制jQuery框架jquery-3.2.1.min.js文件到js目录下。

|  |
| --- |
|  |

3\_创建HTML文件，复制“基本模板”代码到我们创建的HTML文件中。

位置：day30-前端【bootstrap】\资料\Bootstrap离线文档\v3.bootcss.com\index.htm中。

|  |
| --- |
|  |

### 2.2.2模板的详细解释

01\_index.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> *<!--指定为中文-中国-->* <**head**>  *<!-- 指定网页的字符集为utf-8-->* <**meta charset="utf-8"**>  *<!--使用微软最新的浏览器Edge的内核来解析当前的HTML文件，最新的浏览器都支持 -->* <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  *<!--  响应式布局参数：  viewport： 视口的参数  width=device-width：默认的网页宽度是设备的宽度  initial-scale=1: 初始的缩放比1：1  -->* <**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！ -->* <**title**>我的第一个Bootstrap程序</**title**>  *<!-- 1. 导入bootstrap的全局css样式 -->* <**link href="css/bootstrap.min.css" rel="stylesheet"**>  *<!-- 2. 导入 jQuery 框架 -->* <**script src="js/jquery-3.2.1.min.js"**></**script**>  *<!-- 3. 导入bootstrap框架 -->* <**script src="js/bootstrap.min.js"**></**script**> </**head**>  <**body**>  <**h1**>你好，世界！</**h1**> </**body**> </**html**> |

### 2.2.3什么是jQuery框架

后期有jQuery课程祥讲，今天理解一下。

|  |
| --- |
|  |

# 第03章 栅格系统【重点】

## 3.1组成

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。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。

下面就介绍一下 Bootstrap 栅格系统的特点：

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）和内间距（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\_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。

5\_如果一“行（row）”中包含了的“列（column）”大于 12，多余的“列（column）”所在的元素将被作为一个整体另起一行排列。

## 3.2网页布局的两种容器

|  |  |
| --- | --- |
| **布局的两种容器的类样式名** | **说明** |
| **container** | **表示固定宽度，在不同的设备上有不同的固定宽度**  **手机上100px；平板上400px；电脑上600px；** |
| **container-fluid** | **在所有的设备上都是以100%占全屏**  **手机，平板，电脑，铺满整个屏幕；** |

### 3.2.1案例需求

针对台式机电脑/电脑，用div标签，演示container和container-fluid的区别。

### 3.2.2案例效果

|  |
| --- |
|  |

### 3.2.3案例代码

02\_container-fluid.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>网页布局的两种容器</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **div**{  **height**: 200**px**;  **border-style**: **solid**;  }  </**style**> </**head**>  <**body**>  <**div class="container"**>  不同设备固定宽高  </**div**>  <**div class="container-fluid"**>  所有设备自动填充宽高  </**div**> </**body**> </**html**> |

## 3.3了解一下媒介查询@media

通过不同的媒介/设备类型和条件定义样式表CSS规则。媒介查询让CSS可以更精确作用于不同的媒介/设备类型和同一媒介的不同条件。媒介查询的大部分媒介特性都接受min和max用于表达“大于或等于”和“小于或等于”。 打开文件：bootstrap.css，可以看到以下代码：

|  |
| --- |
| .container {  padding-right: 15px;  padding-left: 15px;  margin-right: auto;  margin-left: auto;  }  @media (min-width: 768px) {  .container {  width: 750px;  }  }  @media (min-width: 992px) {  .container {  width: 970px;  }  }  @media (min-width: 1200px) {  .container {  width: 1170px;  }  } |

所以当我们创建div的类样式名container，则会根据屏幕的大小发生变化。

<div class=”.container”></div>

注意：今天都是使用类样式，其中类样式都是Bootstrap内置的，我们无需创建，直接使用。

## 3.4基本写法

|  |  |  |
| --- | --- | --- |
| **栅格系统** | **描述** | **类似于表格** |
| **.container或 .container-fluid** | **表示栅格系统的两种容器，一般放在最外面** | **<table>** |
| **.row或不写** | **表示容器中的一行，一行最多有12列** | **<tr>** |
| **.col-xx-n**  **xx有四个取值**  **1) lg大型设备，如：电视机**  **2) md 中型设备，如：电脑**  **3) sm小型设备，如：平板**  **4) xs微型设备，如：手机**  **n 这一列占几格，一行最多12格** | **这一格在某种设备上占多少列**  **.col-lg-3 在大型设备上占3格**  **.col-md-6 在中型/电脑设备占6格**  **.col-sm-2 在小型/平板设备上占2格**  **.col-xs-8 在微型/手机设备上占8格** | **<td>** |

## 3.5栅格的参数

|  |
| --- |
|  |

# 第04章 栅格系统的案例【重点】

## 4.1案例1

### 4.1.1案例需求

栅格系统的基本结构。为了让div可见，设置div的边框和高度的样式。

一个单元格占1列。

### 4.1.2案例效果

|  |
| --- |
|  |

### 4.1.3案例代码

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>案例1</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **div**{  **height**: 200**px**;  **border-style**: **solid**;  **border-color**: **red**;  }  </**style**> </**head**> <**body**>  *<!--table-->* <**div class="container-fluid"**>  *<!--tr-->* <**div class="row"**>  *<!--td-->* <**div class="col-md-1"**>1列</**div**>  <**div class="col-md-1"**>2列</**div**>  <**div class="col-md-1"**>3列</**div**>  <**div class="col-md-1"**>4列</**div**>  <**div class="col-md-1"**>5列</**div**>  <**div class="col-md-1"**>6列</**div**>  <**div class="col-md-1"**>7列</**div**>  <**div class="col-md-1"**>8列</**div**>  <**div class="col-md-1"**>9列</**div**>  <**div class="col-md-1"**>10列</**div**>  <**div class="col-md-1"**>11列</**div**>  <**div class="col-md-1"**>12列</**div**>  </**div**>  </**div**> </**body**> </**html**> |

注意：省略<div class=”row”></div>也行，但为了清晰，建议加上class=”row”。

## 4.2案例2

### 4.2.1案例需求

栅格系统的基本结构。为了让div可见，设置div的边框和高度的样式。

一个单元格占3列。

一个单元格占6列。

一个单元格占12列。

### 4.2.2案例效果

|  |
| --- |
|  |

### 4.2.3案例代码

04\_grid.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>案例2</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **div**{  **height**: 100**px**;  **border-style**: **solid**;  **border-color**: **red**;  }  </**style**> </**head**> <**body**>  *<!--table-->* <**div class="container-fluid"**>  *<!--tr-->* <**div class="row"**>  *<!--td-->* <**div class="col-md-1"**>1个单元格占1列</**div**>  <**div class="col-md-1"**>2列</**div**>  <**div class="col-md-1"**>3列</**div**>  <**div class="col-md-1"**>4列</**div**>  <**div class="col-md-1"**>5列</**div**>  <**div class="col-md-1"**>6列</**div**>  <**div class="col-md-1"**>7列</**div**>  <**div class="col-md-1"**>8列</**div**>  <**div class="col-md-1"**>9列</**div**>  <**div class="col-md-1"**>10列</**div**>  <**div class="col-md-1"**>11列</**div**>  <**div class="col-md-1"**>12列</**div**>  </**div**>  *<!--tr-->* <**div class="row"**>  *<!--td-->* <**div class="col-md-3"**>1个单元格占3列</**div**>  <**div class="col-md-3"**>1个单元格占3列</**div**>  <**div class="col-md-3"**>1个单元格占3列</**div**>  <**div class="col-md-3"**>1个单元格占3列</**div**>  </**div**>  *<!--tr-->* <**div class="row"**>  *<!--td-->* <**div class="col-md-6"**>1个单元格占6列</**div**>  <**div class="col-md-6"**>1个单元格占6列</**div**>  </**div**>  *<!--tr-->* <**div class="row"**>  *<!--td-->* <**div class="col-md-12"**>1个单元格占12列</**div**>  </**div**>  </**div**> </**body**> </**html**> |

## 4.3案例3

### 4.3.1案例需求

不同屏幕的适配。每个div设置三个样式，col-md-3 col-sm-4 col-xs-6，格子的数量会随着屏幕尺寸的变化而不同，一行中多余的单元格会显示在下一行中。

### 4.3.2案例效果

手机中显示：

|  |
| --- |
|  |

平板中显示：

|  |
| --- |
|  |

电脑中显示：

|  |
| --- |
|  |

### 4.3.3案例代码

05\_grid.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>案例3</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **div**{  **height**: 200**px**;  **border-style**: **solid**;  **border-color**: **red**;  }  </**style**> </**head**> <**body**>  *<!--table-->* <**div class="container-fluid"**>  *<!--tr-->* <**div class="row"**>  *<!--td-->  <!--在电脑中，1个单元格占3列-->  <!--在平板中，1个单元格占4列-->  <!--在手机中，1个单元格占6列-->* <**div class="col-md-3 col-sm-4 col-xs-6"**>1</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6"**>2</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6"**>3</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6"**>4</**div**>  </**div**>  </**div**> </**body**> </**html**> |

## 4.4案例4

### 4.4.1案例需求

显示与隐藏单元格。不同屏幕尺寸的时候，让某些div块的显示，也可以让某些div隐藏。

### 4.4.2案例效果

电脑中显示：

|  |
| --- |
|  |

平板中显示：

|  |
| --- |
|  |

手机中显示：

|  |
| --- |
|  |

### 4.4.3案例代码

06\_grid.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>案例4</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **div**{  **height**: 200**px**;  **border-style**: **solid**;  **border-color**: **red**;  }  </**style**> </**head**> <**body**>  <**div class="container-fluid"**>  <**div class="row"**>  <**div class="col-md-3 col-sm-4 col-xs-6"**>1</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6"**>2</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6 hidden-xs"**>手机中隐藏</**div**>  <**div class="col-md-3 col-sm-4 col-xs-6 hidden-sm hidden-xs"**>平板中隐藏&手机中隐藏</**div**>  </**div**>  </**div**> </**body**> </**html**> |

## 4.5栅格系统中类的小结

|  |  |
| --- | --- |
| **类样式名** | **作用** |
| **.container** | **不同的设备上不同的固定大小** |
| **.container-fluid** | **所有的设备上100%** |
| **.row** | **表示一行，可以省略** |
| **.col-xs-n** | **在微型设备上占n格，如：手机** |
| **.col-sm-n** | **在小型设备上占n格，如：平板** |
| **.col-md-n** | **在中型设备上占n格，如：电脑** |
| **.col-lg-n** | **在大型设备上占n格，如：电视** |
| **.hidden-lg/md/sm/xs** | **在指定设备上隐藏** |
| **.visible-lg/md/sm/xs** | **只在指定的设备上显示，默认都是显示的** |

# 第05章 全局CSS样式

## 5.1按钮

### 5.1.1普通按钮

可作为按钮使用的标签或元素，为 <a>、<button> 或 <input> 元素添加按钮类。

注意：class=”btn btn-default”表示这个标签或元素即使用btn样式，又使用btn-default样式，

二个样式的前后顺序无关。

效果：

|  |
| --- |
|  |

示例：

07\_button.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>普通按钮</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**> </**head**> <**body**>  <**a href="" class="btn btn-default"**>a标签按钮1</**a**>  <**input type="button" value="input标签按钮2" class="btn btn-default"**/>  <**button class="btn btn-default"**>button标签按钮3</**button**> </**body**> </**html**> |

### 5.1.2预定义样式的按钮

有不同颜色的按钮。

其中btn btn-primary，btn btn-btn-success，btn btn-info这些样式都是Bootstrap框架内置的按钮样式。

效果：

|  |
| --- |
|  |

示例：

08\_button.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>预定义样式的按钮</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  </**head**> <**body**>  <**input type="button" value="input [class=btn btn-primary]" class="btn btn-primary"**/><**br**/><**br**/>  <**input type="button" value="input [class=btn btn-success]" class="btn btn-success"**/><**br**/><**br**/>  <**input type="button" value="input [class=btn btn-info]" class="btn btn-info"**/><**br**/><**br**/> </**body**> </**html**> |

## 5.2图片

### 5.2.1响应式图片

通过为图片添加.img-responsive类可以让图片支持响应式布局。其实质是为图片设置了 max-width: 100%;、 height: auto; 和 display: block; 属性，从而让图片在其父元素中更好的缩放。

如果需要让使用了.img-responsive 类的图片水平居中，请使用.center-block 类，不要用 .text-center。

### 5.2.2图片形状

|  |  |
| --- | --- |
| **类样式名** | **作用** |
| **.img-responsive** | **图片具有响应式** |
| **.center-block** | **图片水平居中** |
| **.img-rounded** | **图片背景是圆角矩形** |
| **.img-circle** | **图片背景是圆形** |
| **.img-thumbnail** | **图片背景是矩形** |

效果：

|  |
| --- |
|  |

示例：

09\_img.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>图片</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**>  <**style**>  **img**{  **background-color**: **yellow**;  }  </**style**> </**head**> <**body**>  <**img src="img/gou.png" class="img-responsive center-block img-rounded"**/><**br**/>  <**img src="img/gou.png" class="img-responsive center-block img-circle"**/><**br**/>  <**img src="img/gou.png" class="img-responsive center-block img-thumbnail"**/><**br**/> </**body**> </**html**> |

## 5.3表单

### 5.3.1与表单有关的类样式

收集客户端数据，并提交给服务器。

|  |  |
| --- | --- |
| **类样式名** | **作用** |
| **.form-control** | **让表单项元素宽为width:100%** |
| **.form-group** | **让 label 元素和表单项控件包裹在一起，达到排列效果** |

|  |  |
| --- | --- |
| **.form-inline** | **让表单项控件值水平排列，如：姓别** |

所有设置了.form-control 类的 <input>、<textarea> 和 <select> 元素都将被默认设置宽度属性为 width: 100%;。 将 label 元素和前面提到的控件包裹在 .form-group 中可以获得最好的排列，在Bootstrap看来，排列不要用table标签，而用div标签。

|  |
| --- |
| <form action=”” method=”>  <div class="form-group">  <label for="username">姓名：</label>  <input type="email" class="form-control" id="username" placeholder="请输入姓名"/>  </div>  </form> |

### 5.3.2案例：添加联系人的表单

#### 5.3.2.1案例需求

使用Bootstrap创建如图所示的添加联系人的表单，支持响应式布局。

#### 5.3.2.2案例效果

|  |
| --- |
|  |

#### 5.3.2.3案例代码

10\_contact.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>添加联系人的表单</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**> </**head**> <**body**>  <**h3**>添加联系人</**h3**>  <**form action="09\_img.html" method="post" style="width**:20%**"**>  <**div class="form-group"**>  <**label**>用户名：</**label**>  <**input type="text" name="username" class="form-control" placeholder="请输入用户名"**/>  </**div**>  <**div class="form-group"**>  <**label**>性别：</**label**>  <**input type="radio" name="sex" class="form-inline" value="男"**/><**label**>男</**label**>  <**input type="radio" name="sex" class="form-inline" value="女"**/><**label**>女</**label**>  </**div**>  <**div class="form-group"**>  <**label**>生日：</**label**>  <**input type="date" name="birthday" class="form-control"**/>  </**div**>  <**div class="form-group"**>  <**label**>学历：</**label**>  <**select name="edu" class="form-control"**>  <**option**>小学</**option**>  <**option**>中学</**option**>  <**option**>大学</**option**>  <**option**>社会</**option**>  </**select**>  </**div**>  <**div class="form-group"**>  <**input type="submit" value="提交" class="btn btn-primary"**/>  **&nbsp;&nbsp;&nbsp;&nbsp;** <**input type="submit" value="取消" class="btn btn-default"**/>  **&nbsp;&nbsp;&nbsp;&nbsp;** <**input type="submit" value="退出" class="btn btn-danger"**/>  </**div**>  </**form**> </**body**> </**html**> |

## 5.4表格

### 5.4.1与表格有关的类样式

|  |  |
| --- | --- |
| **类样式名** | **作用** |
| **.table** | **基本实例** |
| **.table-striped** | **条纹状表格** |
| **.table-bordered** | **带边框的表格** |
| **.table-hover** | **鼠标悬停** |

### 5.4.2表格中的状态类

即单独对某个td或tr设置不同的颜色，通过这些状态类可以为行或单元格设置颜色。

|  |  |
| --- | --- |
| **类样式名** | **作用** |
| **.active** | **单元格上时所设置的颜色** |
| **.success** | **标识成功或积极的动作** |
| **.info** | **标识普通的提示信息或动作** |
| **.warning** | **标识警告或需要用户注意** |
| **.danger** | **标识危险或潜在的带来负面影响的动作** |

### 5.4.3案例：联系人信息的表格

#### 5.4.3.1案例需求

使用Bootstrap制作如下所示的表格效果，支持响应式布局。

#### 5.4.3.2案例效果

|  |
| --- |
|  |

#### 5.4.3.3案例代码

11\_contact.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>联系人信息的表格</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**> </**head**> <**body**>  <**h3**>联系人信息</**h3**>  <**table class="table table-striped table-bordered table-hover"**>  <**tr class="success"**>  <**th**>编号</**th**>  <**th**>姓名</**th**>  <**th**>工资</**th**>  </**tr**>  <**tr class="warning"**>  <**td**>111</**td**>  <**td**>张三</**td**>  <**td**>8000</**td**>  </**tr**>  <**tr class="active"**>  <**td**>222</**td**>  <**td**>李四</**td**>  <**td**>9000</**td**>  </**tr**>  </**table**> </**body**> </**html**> |

# ========查询手册，复制代码，进行修改。

# 第06章 组件

|  |
| --- |
|  |

|  |
| --- |
|  |

## 6.1导航条组件

### 6.1.1作用

导航条是在您的应用或网站中作为导航页头的响应式基础组件。它们在移动设备上可以折叠（并且可开可关），且在视口（viewport）宽度增加时逐渐变为水平展开模式。

### 6.1.2组成

整个导航条就是一个nav标签。

|  |  |
| --- | --- |
| **导航条的类样式** | **作用** |
| **.navbar** | **默认的颜色为白色** |
| **.navbar-default** |
| **.navbar-header** | **指定商标和开关的样式，让它在手机上更好的显示** |
| **.collapse navbar-collapse** | **可以折叠的所有项** |
| **.dropdown** | **下拉菜单** |
| **.navbar-left** | **左对齐** |
| **.navbar-right** | **右对齐** |

|  |  |
| --- | --- |
| **开头相关的属性** | **作用** |
| **data-** | **表示这有一个事件激活，会产生相应的功能。不能删除** |
| **aria-** | **给残障人士使用，可以删除** |
| **sr-only** | **屏幕阅读的功能，给残障人士使用，可以删除** |

### 6.1.3案例：自制导航条组件

#### 6.1.3.1案例需求

参考文档，自制导航条。

#### 6.1.3.2案例效果

|  |
| --- |
|  |

#### 6.1.3.3案例代码

12\_nav.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**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！ -->* <**title**>自制导航条组件</**title**>  <**link href="css/bootstrap.min.css" type="text/css" rel="stylesheet"**>  <**script src="js/jquery-3.2.1.min.js" type="text/javascript"**></**script**>  <**script src="js/bootstrap.min.js" type="text/javascript"**></**script**> </**head**> <**body**>  <**nav class="navbar navbar-inverse"**>  <**div class="container-fluid"**>  *<!-- Brand and toggle get grouped for better mobile display -->* <**div class="navbar-header"**>  <**button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-expanded="false"**>  <**span class="sr-only"**>Toggle navigation</**span**>  <**span class="icon-bar"**></**span**>  <**span class="icon-bar"**></**span**>  <**span class="icon-bar"**></**span**>  </**button**>  <**a class="navbar-brand" href="#"**>首页</**a**>  </**div**>   *<!-- Collect the nav links, forms, and other content for toggling -->* <**div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1"**>  <**ul class="nav navbar-nav"**>  <**li class="active"**><**a href="#"**>电脑<**span class="sr-only"**>(current)</**span**></**a**></**li**>  <**li**><**a href="#"**>平板</**a**></**li**>  <**li class="dropdown"**>  <**a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true" aria-expanded="false"**>手机<**span class="caret"**></**span**></**a**>  <**ul class="dropdown-menu"**>  <**li**><**a href="#"**>华为</**a**></**li**>  <**li**><**a href="#"**>中兴</**a**></**li**>  <**li**><**a href="#"**>小米</**a**></**li**>  <**li role="separator" class="divider"**></**li**>  <**li**><**a href="#"**>苹果</**a**></**li**>  <**li role="separator" class="divider"**></**li**>  <**li**><**a href="#"**>其它</**a**></**li**>  </**ul**>  </**li**>  </**ul**>  <**form class="navbar-form navbar-right" action="13\_fy.html" method="get"**>  <**div class="form-group"**>  <**input type="text" class="form-control" placeholder="输入商品" name="name"**>  </**div**>  <**button type="submit" class="btn btn-default"**>查询</**button**>  </**form**>  </**div**>*<!-- /.navbar-collapse -->* </**div**>*<!-- /.container-fluid -->* </**nav**> </**body**> </**html**> |

## 6.2分页组件

### 6.2.1作用

为您的网站或应用提供带有展示页码的分页组件。

### 6.2.2组成

|  |  |
| --- | --- |
| **分页类样式** | **作用** |
| **.pagination** | **分页容器** |
| **&laquo;** | **向左的两个箭头** |
| **&raquo;** | **向右的两个箭头** |
| **.disabled** | **禁用状态** |
| **.active** | **激活状态** |

### 6.2.3案例：自制分页条组件

#### 6.2.3.1案例需求

参考文档，自制分页条。

#### 6.2.3.2案例效果

|  |
| --- |
|  |

|  |
| --- |
|  |

#### 6.2.3.3案例代码

13\_fy.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**meta name="viewport" content="width=device-width, initial-scale=1"**>  <**title**>自制分页组件</**title**>  <**link type="text/css" rel="stylesheet" href="css/bootstrap.min.css"**>  <**script src="js/jquery-3.2.1.min.js"**></**script**>  <**script src="js/bootstrap.min.js"**></**script**> </**head**> <**body**> <**nav**>  *<!--分页条-->* <**ul class="pagination"**>  *<!--下一页，已失效-->* <**li class="disabled"**>  <**a href="#"**>  <**span**>**&laquo;**</**span**>  </**a**>  </**li**>  <**li**><**a href="#"**>1</**a**></**li**>  <**li**><**a href="#"**>2</**a**></**li**>  *<!--第三页，已激活-->* <**li class="active"**><**a href="#"**>3</**a**></**li**>  <**li**><**a href="#"**>4</**a**></**li**>  <**li**><**a href="#"**>5</**a**></**li**>  *<!--下一页-->* <**li**>  <**a href="#"**>  <**span**>**&raquo;**</**span**>  </**a**>  </**li**>  </**ul**> </**nav**> </**body**> </**html**> |

# 第07章 JavaScript插件

|  |
| --- |
|  |

|  |
| --- |
|  |

注意：JavaScript插件必须要使用到jQuery框架。

## 7.1模态框

### 7.1.1什么是模态框

页面上出现一个对话框在其它的元素前面，如果这个对话框没有关闭，其它的元素也不能操作。如：window.alert(“警告框”)。

|  |
| --- |
|  |

### 7.1.2对话框的样式

|  |  |
| --- | --- |
| **样式名** | **描述** |
| **.modal-dialog** | 模态框的容器 |
| **.modal-content** | 模态框的内容 |
| **.modal-header** | 模态框的头部 |
| **.modal-body** | 模态框的主体 |
| **.modal-footer** | 模态框的脚部 |
| **.modal-title** | 头部中的标题 |

### 7.1.3对话框的属性

|  |  |
| --- | --- |
| **按钮上的属性** | **作用** |
| **data-toggle="modal"** | **打开模态框的功能** |
| **data-target="#id值" 或 =".类名"** | **指定需要打开的id或类名来打开模态框** |
| **data-dismiss="modal"** | **关闭模态框的功能** |

### 7.1.4案例：自制模态框组件

#### 7.1.4.1案例需求

参考文档，自制模态框。

#### 7.1.4.2案例效果

|  |
| --- |
|  |

#### 7.1.4.3案例代码

14\_dialog.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**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！ -->* <**title**>自制模态框组件</**title**>  <**link href="css/bootstrap.min.css" type="text/css" rel="stylesheet"**>  <**script src="js/jquery-3.2.1.min.js" type="text/javascript"**></**script**>  <**script src="js/bootstrap.min.js" type="text/javascript"**></**script**> </**head**> <**body**>  *<!-- Button trigger modal -->* <**button type="button" class="btn btn-primary btn-lg" data-toggle="modal" data-target="#myModal"**>  打开注册用户页面  </**button**>   *<!-- Modal -->* <**div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel"**>  <**div class="modal-dialog" role="document"**>  <**div class="modal-content"**>  <**div class="modal-header"**>  <**button type="button" class="close" data-dismiss="modal" aria-label="Close"**><**span aria-hidden="true"**>**&times;**</**span**></**button**>  <**h4 class="modal-title" id="myModalLabel"**>Modal title</**h4**>  </**div**>  <**div class="modal-body"**>  <**form action="javascript:void(0)" method="post" id="myform" onsubmit="return** *check*()**"**>  <**table class="main" border="0" cellspacing="0" cellpadding="0"**>  <**tr**>  <**td**><**img src="img/logo.jpg" alt="logo"** /><**img src="img/banner.jpg" alt="banner"** /></**td**>  </**tr**>  <**tr**>  <**td class="hr\_1"**>新用户注册</**td**>  </**tr**>  <**tr**>  <**td style="height**:10**px**;**"**></**td**>  </**tr**>  <**tr**>  <**td**>  <**table width="100%" border="0" cellspacing="0" cellpadding="0"**>  <**tr**>  <**td class="left"**>用户名：</**td**>  <**td class="center"**>  <**input id="user" name="user" type="text" class="in"** />  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>密码：</**td**>  <**td class="center"**>  <**input id="pwd" name="pwd" type="password" class="in"** />  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>确认密码：</**td**>  <**td class="center"**>  <**input id="repwd" name="repwd" type="password" class="in"**/>  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>电子邮箱：</**td**>  <**td class="center"**>  <**input id="email" name="email" type="text" class="in"**/>  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>手机号码：</**td**>  <**td class="center"**>  <**input id="mobile" name="mobile" type="text" class="in"**/>  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>生日：</**td**>  <**td class="center"**>  <**input id="birth" name="birth" type="date" class="in"**/>  </**td**>  </**tr**>  <**tr**>  <**td class="left"**>**&nbsp;**</**td**>  <**td class="center"**>  <**input name="" type="image" src="img/register.jpg"** />  </**td**>  </**tr**>  </**table**>  </**tr**>  </**table**>  </**form**>  </**div**>  <**div class="modal-footer"**>  <**button type="button" class="btn btn-default" data-dismiss="modal"**>关闭</**button**>  </**div**>  </**div**>  </**div**>  </**div**>      <**script**>  *//针对用户名验证的函数* **function** *checkUsername*(username){*//" "* **var** flag = **false**;  *//去掉二边的空格* username = username.trim();  *//如果长度不为0* **if**(username.length > 0){  *//如果符合要求* **if**(/^[a-z][a-zA-Z0-9]{3,15}$/.test(username)){  flag = **true**;  }**else**{  ***window***.alert(**"用户名不合法"**);  }  *//如果长度为0* }**else**{  ***window***.alert(**"用户名必填"**);  }  **return** flag;  }   *//针对密码验证的函数* **function** *checkPassword*(password){  **var** flag = **false**;  password = password.trim();  **if**(password.length > 0){  **if**(/^\d{6,20}$/.test(password)){  flag = **true**;  }**else**{  ***window***.alert(**"密码不合法"**);  }  }**else**{  ***window***.alert(**"密码必填"**);  }  **return** flag;  }    *//针对确认密码验证的函数* **function** *checkRePassword*(repassword,password){  **var** flag = **false**;  repassword = repassword.trim();  **if**(repassword.length > 0){  **if**(repassword == password){  flag = **true**;  }**else**{  ***window***.alert(**"二次密码输入不相同"**);  }  }**else**{  ***window***.alert(**"确认密码必填"**);  }  **return** flag;  }   *//针对邮箱验证的函数* **function** *checkEmail*(email){  **var** flag = **false**;  email = email.trim();  **if**(email.length > 0){  *//zhaojun@itcast.cn  //zhaojun@itcast.com.cn  //这里的\w等价于[a-zA-Z0-9\_]* **if**(/^\w+@\w+(.\w{2,3}){1,2}$/.test(email)){  flag = **true**;  }**else**{  ***window***.alert(**"邮箱格式不合法"**);  }  }**else**{  ***window***.alert(**"邮箱必填"**);  }  **return** flag;  }   *//针对手机号验证的函数* **function** *checkMobile*(mobile){  **var** flag = **false**;  mobile = mobile.trim();  **if**(mobile.length > 0){  **if**(/^1\d{10}$/.test(mobile)){  flag = **true**;  }**else**{  ***window***.alert(**"手机号格式不合法"**);  }  }**else**{  ***window***.alert(**"手机号必填"**);  }  **return** flag;  }   *//总的验证函数* **function** *check*(){  **var** flag = **false**;   *//获取用户名* **var** username = ***document***.getElementById(**"user"**).**value**;  *//获取密码* **var** password = ***document***.getElementById(**"pwd"**).**value**;  *//获取确认密码* **var** repassword = ***document***.getElementById(**"repwd"**).**value**;  *//获取电子邮箱* **var** email = ***document***.getElementById(**"email"**).**value**;  *//获取手机号* **var** mobile = ***document***.getElementById(**"mobile"**).**value**;    *//验证用户名* **if**(*checkUsername*(username)){  **if**(*checkPassword*(password)){  **if**(*checkRePassword*(repassword,password)){  **if**(*checkEmail*(email)){  **if**(*checkMobile*(mobile)){  flag=**true**;  }  }  }  }  }   **return** flag;  }  </**script**> </**body**> </**html**> |

## 7.2轮播图【carousel】

### 7.2.1什么是轮播图

这个组件用于轮播显示一组图片，类似于旋转木马，页面加载就自动运行，也可以通过点击左右的两个箭头向左或向右翻页。

|  |
| --- |
|  |

### 7.2.2轮播图的样式

|  |  |
| --- | --- |
| **类样式名字** | **作用** |
| **.carousel slide** | **轮播图的容器** |
| **.carousel-indicators** | **中间的小圆点** |
| **.carousel-inner** | **要轮播的图片** |
| **.carousel-caption** | **图片的名字** |
| **.carousel-control** | **向左向右按钮** |

### 7.2.3轮播图的属性

|  |  |
| --- | --- |
| **属性名** | **作用** |
| **data-ride="carousel"** | **指定当前是一个轮播图，加载时自动运行** |
| **data-interval="2000"** | **间隔的时间间隔，单位是毫秒** |
| **data-target="#myPic"** | **指定轮播图div容器的id值** |

### 7.2.4案例：自制轮播图组件

#### 7.2.4.1案例需求

参考文档，自制轮播图。

#### 7.2.4.2案例效果

|  |
| --- |
|  |

#### 7.2.4.3案例代码

15\_lun.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="zh-CN"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**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！ -->* <**title**>自制轮播图组件</**title**>  <**link href="css/bootstrap.min.css" type="text/css" rel="stylesheet"**>  <**script src="js/jquery-3.2.1.min.js" type="text/javascript"**></**script**>  <**script src="js/bootstrap.min.js" type="text/javascript"**></**script**> </**head**> <**body**>  <**div id="carousel-example-generic" class="carousel slide" data-ride="carousel"**>  *<!-- Indicators -->* <**ol class="carousel-indicators"**>  <**li data-target="#carousel-example-generic" data-slide-to="0" class="active"**></**li**>  <**li data-target="#carousel-example-generic" data-slide-to="1"**></**li**>  <**li data-target="#carousel-example-generic" data-slide-to="2"**></**li**>  </**ol**>   *<!-- Wrapper for slides -->* <**div class="carousel-inner" role="listbox"**>  <**div class="item"**>  <**img src="img/1.jpg"**>  </**div**>  <**div class="item"**>  <**img src="img/2.jpg"**>  </**div**>  <**div class="item active"**>  <**img src="img/3.jpg"**>  </**div**>  </**div**>   *<!-- Controls -->* <**a class="left carousel-control" href="#carousel-example-generic" role="button" data-slide="prev"**>  <**span class="glyphicon glyphicon-chevron-left" aria-hidden="true"**></**span**>  <**span class="sr-only"**>Previous</**span**>  </**a**>  <**a class="right carousel-control" href="#carousel-example-generic" role="button" data-slide="next"**>  <**span class="glyphicon glyphicon-chevron-right" aria-hidden="true"**></**span**>  <**span class="sr-only"**>Next</**span**>  </**a**>  </**div**>   <**script**>  *//轮播时间，单位毫秒* $(**".carousel"**).**carousel**({  **interval**: 1000  })  </**script**> </**body**> </**html**> |

# 第08章 案例：完成黑马旅游的网站首页

## 8.1案例需求

使用我们学过的 HTML、CSS 和 Bootstrap 制作黑马旅游的首页。

## 8.2案例效果

|  |
| --- |
|  |

|  |
| --- |
|  |

## 8.3案例代码

index.css

|  |
| --- |
| .**navbar a**{  **margin**: 0**px** 32**px**; } .**navbar a**:**link**{  **text-decoration**: **underline**; } .**navbar ul**{  **background-color**: **yellow**;  **line-height**: 10**px**; } .**company**{  **background-color**: **yellow**;  **color**: **gray**;  **font-size**: 8**px**;  **text-align**: **center**; } |

16\_heimalvyou\_app.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="utf-8"**>  <**meta http-equiv="X-UA-Compatible" content="IE=edge"**>  <**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！ -->* <**title**>黑马旅游首页</**title**>  <**link href="css/bootstrap.min.css" type="text/css" rel="stylesheet"**>  <**link href="css/index.css" type="text/css" rel="stylesheet"**>  <**script src="js/jquery-3.2.1.min.js" type="text/javascript"**></**script**>  <**script src="js/bootstrap.min.js" type="text/javascript"**></**script**> </**head**> <**body**>  *<!--头部-->* <**header**>   *<!--第1行-->* <**div class="container-fluid"**>  <**div class="row"**>  <**div class="col-md-12"**>  <**img src="imgs/top\_banner.jpg" class="img-responsive"**/>  </**div**>  </**div**>  </**div**>   *<!--第2行-->* <**div class="container-fluid"**>  <**div class="row"**>  <**div class="col-md-2"**>  <**img src="imgs/logo.jpg" class="img-responsive"**/>  </**div**>  <**div class="col-md-8"**>  <**img src="imgs/search.png" class="img-responsive"**/>  </**div**>  <**div class="col-md-2"**>  <**img src="imgs/hotel\_tel.png" class="img-responsive"**/>  </**div**>  </**div**>  </**div**>   *<!--第3行-->* <**div class="container-fluid"**>  <**div class="row"**>  <**div class="col-md-12"**>  *<!--嵌入导航条-->* <**nav class="navbar navbar-default"**>  <**div class="container-fluid"**>  <**div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1"**>  *<!--主体内容-->* <**ul class="nav navbar-nav"**>  <**li**><**a href="#"**>首页<**span class="sr-only"**>(current)</**span**></**a**></**li**>  <**li**><**a href="#"**>国内</**a**></**li**>  <**li**><**a href="#"**>国际</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  <**li**><**a href="#"**>首页</**a**></**li**>  </**ul**>  </**div**>*<!-- /.navbar-collapse -->* </**div**>*<!-- /.container-fluid -->* </**nav**>  </**div**>  </**div**>  </**div**>   *<!--第3行-->* <**div class="container-fluid"**>  <**div class="row"**>  <**div class="col-md-12"**>  *<!--嵌入轮播图-->* <**div id="carousel-example-generic" class="carousel slide" data-ride="carousel"**>   *<!-- Indicators -->* <**ol class="carousel-indicators"**>  <**li data-target="#carousel-example-generic" data-slide-to="0" class="active"**></**li**>  <**li data-target="#carousel-example-generic" data-slide-to="1"**></**li**>  <**li data-target="#carousel-example-generic" data-slide-to="2"**></**li**>  </**ol**>   *<!-- 图片 -->* <**div class="carousel-inner" role="listbox"**>  <**div class="item active"**>  <**img src="imgs/banner\_1.jpg" alt="..." title=""**>  <**div class="carousel-caption"**>  1  </**div**>  </**div**>  <**div class="item"**>  <**img src="imgs/banner\_2.jpg" alt="..."**>  <**div class="carousel-caption"**>  2  </**div**>  </**div**>  <**div class="item"**>  <**img src="imgs/banner\_3.jpg" alt="..."**>  <**div class="carousel-caption"**>  3  </**div**>  </**div**>  </**div**>   *<!-- Controls -->* <**a class="left carousel-control" href="#carousel-example-generic" role="button" data-slide="prev"**>  <**span class="glyphicon glyphicon-chevron-left" aria-hidden="true"**></**span**>  <**span class="sr-only"**>Previous</**span**>  </**a**>  <**a class="right carousel-control" href="#carousel-example-generic" role="button" data-slide="next"**>  <**span class="glyphicon glyphicon-chevron-right" aria-hidden="true"**></**span**>  <**span class="sr-only"**>Next</**span**>  </**a**>  </**div**>  </**div**>  </**div**>  </**div**>   </**header**>   *<!--作业：体部-->* <**main**></**main**>   *<!--尾部-->* <**footer class="container-fluid"**>  <**div class="col-md-12"**>  <**img src="imgs/footer\_service.png" class="img-responsive"**/>  </**div**>   <**div class="col-md-12"**>  江苏传智播客教育科技股份有限公司<**br**/>  版权所有Copyright 2006-2018, All Rights Reserved 苏ICP备16007882  </**div**>  </**footer**>  </**body**> </**html**> |

# 随堂测试

# 附：日期对象

### 日期对象的创建

作用：Date 对象用于处理日期和时间。

### 创建 Date 对象的语法：

|  |
| --- |
| **var 变量名 =new Date(); // Date 对象会自动把当前日期和时间保存为其初始值。** |

### 日期对象的方法

|  |  |
| --- | --- |
| **方法名** | **作用** |
| **getFullYear()** | **从 Date 对象以四位数字返回年份。** |
| **getMonth()** | **从 Date 对象返回月份 (0 ~ 11)。** |
| **getDate()** | **从 Date 对象返回一个月中的某一天 (1 ~ 31)。** |
| **getDay()** | **从 Date 对象返回一周中的某一天 (0 ~ 6)。其中：0表示周日，1~6周一到周六。** |
| **getHours()** | **返回 Date 对象的小时 (0 ~ 23)。** |
| **getMinutes()** | **返回 Date 对象的分钟 (0 ~ 59)。** |
| **getSeconds()** | **返回 Date 对象的秒数 (0 ~ 59)。** |
| **getMilliseconds()** | **返回 Date 对象的毫秒(0 ~ 999)。** |
| **getTime()** | **返回 1970 年 1 月 1 日至今的毫秒数。类似于Java中的System.currentTimeMillis()** |
| **toLocaleString()** | **根据本地时间格式，把 Date 对象转换为字符串。** |

### 与计时有关的方法

|  |  |
| --- | --- |
| **window中的方法** | **作用** |
| **setTimeout(函数名, 间隔毫秒数)** | **在指定的时间后调用1次函数，只执行1次，单位是毫秒。**  **返回值：返回一个整数类型的计时器**  **函数调用有两种写法：**  **1) setTimeout("函数名(参数)", 1000);**  **2) setTimeout(function(){},1000);**  **注意方式二：没有引号，没有括号** |
| **setInterval(函数名, 间隔毫秒数)** | **每过指定的时间调用1次函数，不停的调用函数，单位是毫秒。返回值：返回一个整数类型的计时器。**  **调用方式也是两种，格式同上。如：var timerID = setInterval();** |
| **clearInterval(计时器)** | **清除setInterval()方法创建的计时器，如clearInterval(timerID);** |
| **clearTimeout(计时器)** | **清除setTimeout()创建的计时器** |