

NSD Devweb DAY03

1. [案例1：编写Bootstrap模板](#)
2. [案例2：使用全局 CSS 样式](#)
3. [案例3：使用常用组件](#)
4. [例4：使用常用插件](#)

1 案例1：编写Bootstrap模板

1.1 问题

【参见COOKBOOK】

1. 使用 Bootstrap 模版创建页面

1.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：使用内联样式为<p>元素设置样式

目录说明：

1)css目录用于存放Bootstrap框架使用的样式文件：

bootstrap.css文件：Bootstrap框架的样式文件

bootstrap.min.css文件：Bootstrap框架的样式压缩文件

bootstrap-theme.css文件：Bootstrap框架的主题文件

2)fonts目录用于存放Bootstrap框架使用的字体文件：

3)js目录用于存放Bootstrap框架使用的核心javascript文件：

bootstrap.js文件：Bootstrap框架的核心javascript文件

bootstrap.min.js文件：Bootstrap框架的核心javascript压缩文件

```
01.  <!-- 声明 -->
02.  <!DOCTYPE html>
03.  <html lang="en">
04.  <head>
05.      <meta charset="UTF-8">
06.      <meta name="viewport" content="width=device-width, initial-scale=1">
07.      <!-- 上述meta标签*必须*放在最前面，任何其他内容都*必须*跟随其后！ 元数据
08.      <title>my bootstrap test</title>
09.      <!-- Bootstrap 样式 -->
10.      <link rel="stylesheet" href="static/css/bootstrap.min.css">
11.  </head>
12.  <body>
```

[Top](#)

```

13.  <!-- 页面内容 -->
14.  <div class="container">
15.      <p>
16.          press <kbd>ctrl + C</kbd> to break.
17.      </p>
18.      <a href="http://www.bootcss.com">bootstrap中文网</a>
19.  </div>
20.  <!-- jQuery ( Bootstrap 的所有 JavaScript 插件都依赖 jQuery ) , 想要使用Bootstrap框架
21.  <script src="static/js/jquery.min.js"></script>
22.  <!-- 加载 Bootstrap 的所有 JavaScript 插件 -->
23.  <script src="static/js/bootstrap.min.js"></script>
24.  </body>
25.  </html>

```

步骤二：实例演示如图-1所示：

press ctrl + C to break.

[bootstrap中文网](http://www.bootcss.com)

图-1

2 案例2：使用全局 CSS 样式

2.1 问题

【参见COOKBOOK】

1. 使用全局 CSS 样式定义页面

2.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：HTML5 文档类型

Bootstrap 使用到的某些 HTML 元素和 CSS 属性需要将页面设置为 HTML5 文档类型。在你项目中的每个页面都要参照下面的格式进行设置

```

01.  <!DOCTYPE html>
02.  <html lang="zh-CN">
03.      ...
04.  </html>

```

[Top](#)

步骤二：移动设备优先

为了确保适当的绘制和触屏缩放，需要在<head>之中添加viewport元数据元素

```
01. <meta name="viewport" content="width=device-width, initial-scale=1">
```

步骤三：将Bootstrap模板补充完整

```
01. <!DOCTYPE html>
02. <html lang="en">
03. <head>
04.   <meta charset="UTF-8">
05.   <meta name="viewport" content="width=device-width, initial-scale=1">
06.   <title>my bootstrap test</title>
07.   <link rel="stylesheet" href="static/css/bootstrap.min.css">
08. </head>
09. <body>
10.   ...
11.   ...
12. <script src="static/js/jquery.min.js"></script>
13. <script src="static/js/bootstrap.min.js"></script>
14. </body>
15. </html>
```

步骤四：了解CSS全局样式的设置

1)为 body 元素设置 background-color: #fff;body元素背景色设置为白色

2)使用 @font-family-base、@font-size-base 和 @line-height-base 变量作为排版的基本参数

3)为所有链接设置了基本颜色 @link-color，并且当链接处于 :hover 状态时才添加下划线

步骤五：布局容器

.container 类用于固定宽度并支持响应式布局的容器

```
01. <div class="container">
02.   ...
03. </div>
```

[Top](#)

.container-fluid 类用于 100% 宽度，占据全部视口 (viewport) 的容器

```

01. <div class="container- fluid">
02.   ...
03. </div>

```

步骤五：按钮

1)按钮预定义样式

使用下面列出的类可以快速创建一个带有预定义样式的按钮：

```

01. <button type="button" class="btn btn- default"> (默认样式) Default</button>
02. <button type="button" class="btn btn- primary "> (首选项) Primary</button>
03. <button type="button" class="btn btn- success"> (成功) Success</button>
04. <button type="button" class="btn btn- info"> (一般信息) Info</button>
05. <button type="button" class="btn btn- warning"> (警告) Warning</button>
06. <button type="button" class="btn btn- danger"> (危险) Danger</button>
07. <!-- 链接按钮鼠标悬停才有下划线-->
08. <button type="button" class="btn btn- link"> (链接) Link</button>

```

实例演示如图-2所示：



图-2

2)按钮尺寸

使用 .btn-lg、.btn-sm 或 .btn-xs 可以获得不同尺寸的按钮

```

01. <p>
02.   <button type="button" class="btn btn- primary btn- lg"> (大按钮) Large button</button>
03.   <button type="button" class="btn btn- default btn- lg"> (大按钮) Large button</button>
04. </p>
05. <p>
06.   <button type="button" class="btn btn- primary "> (默认尺寸) Default button</button>
07.   <button type="button" class="btn btn- default"> (默认尺寸) Default button</button>
08. </p>
09. <p>
10.   <button type="button" class="btn btn- primary btn- sm"> (小按钮) Small button</button>
11.   <button type="button" class="btn btn- default btn- sm"> (小按钮) Small button</button>
12. </p>

```

13. `<p>`
14. `<button type="button" class="btn btn-primary btn-xs">` (超小尺寸) Extra small button
15. `<button type="button" class="btn btn-default btn-xs">` (超小尺寸) Extra small button
16. `</p>`



实例演示如图-3所示：



图-3

通过给按钮添加 `.btn-block` 类可以将其拉伸至父元素100%的宽度，而且按钮也变为了块级 (block) 元素。

01. `<button type="button" class="btn btn-primary btn-lg btn-block">` (块级元素) Block level
02. `<button type="button" class="btn btn-default btn-lg btn-block">` (块级元素) Block level



实例演示如图-4所示：



图-4

步骤六：文本

1)对齐

01. `<!-- 设定文本左对齐 -->`
02. `<p class="text-left">Left aligned text.</p>`
03. `<!-- 设定文本右对齐 -->`

[Top](#)

04. `<p class="text-center">Center aligned text.</p>`
05. `<!-- 设定文本居中对齐-->`
06. `<p class="text-right">Right aligned text.</p>`
07. `<!-- 设定文本对齐，段落超出屏幕部分文字自动换行-->`
08. `<p class="text-justify">Justified text.</p>`
09. `<!-- 段落超出屏幕部分不换行-->`
10. `<p class="text-nowrap">No wrap text.</p>`

2)大小写

01. `<!-- 小写（仅英文）-->`
02. `<p class="text-lowercase">Lowercased text.</p>`
03. `<!-- 文字大写（仅英文）-->`
04. `<p class="text-uppercase">Uppercased text.</p>`
05. `<!-- 首字母大写-->`
06. `<p class="text-capitalize">Capitalized text.</p>`

实例演示如图-5所示：

lowercased text.
UPPERCASED TEXT.
Capitalized Text.

图-5

3)情境文本颜色

通过颜色来展示意图，Bootstrap 提供了一组工具类。这些类可以应用于链接，并且在鼠标经过时颜色可以还可以加深，就像默认的连接一样

01. `<p class="text-muted">...</p>`
02. `<p class="text-primary">...</p>`
03. `<p class="text-success">...</p>`
04. `<p class="text-info">...</p>`
05. `<p class="text-warning">...</p>`
06. `<p class="text-danger">...</p>`

实例演示如图-6所示：

[Top](#)

Fusce dapibus, tellus ac cursus commodo, tortor mauris nibh.

Nullam id dolor id nibh ultricies vehicula ut id elit.

Duis mollis, est non commodo luctus, nisi erat porttitor ligula.

Maecenas sed diam eget risus varius blandit sit amet non magna.

Etiam porta sem malesuada magna mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

图-6

步骤七：列表

1)无样式列表

移除了默认的 list-style 样式和左侧外边距的一组元素

```
01. <ul class="list- unstyled">
02.   <li>...</li>
03. </ul>
```

实例演示如图-7所示：

Lorem ipsum dolor sit amet
 Consectetur adipiscing elit
 Integer molestie lorem at massa
 Facilisis in pretium nisl aliquet
 Nulla volutpat aliquam velit
 o Phasellus iaculis neque
 o Purus sodales ultricies
 o Vestibulum laoreet porttitor sem
 o Ac tristique libero volutpat at
 Faucibus porta lacus fringilla vel
 Aenean sit amet erat nunc
 Eget porttitor lorem

图-7

2)内联列表

通过设置 display: inline-block , 将所有元素放置于同一行。

```
01. <ul class="list- inline">
02.   <li>...</li>
03. </ul>
```

[Top](#)

实例演示如图-8所示：

Lorem ipsum Phasellus iaculis Nulla volutpat

图-8

3)描述

带有描述的短语列表

01. `<dl>`
02. `<dt>...</dt>`
03. `<dd>...</dd>`
04. `</dl>`

实例演示如图-9所示：

Description lists

A description list is perfect for defining terms.

Euismod

Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit.

Donec id elit non mi porta gravida at eget metus.

Malesuada porta

Etiam porta sem malesuada magna mollis euismod.

图-9

水平排列的描述

.dl-horizontal 可以让 <dl> 内的短语及其描述排在一行。开始是像 <dl> 的默认样式堆叠在一起，随着导航条逐渐展开而排列在一行。

01. `<dl class="dl-horizontal">`
02. `<dt>...</dt>`
03. `<dd>...</dd>`
04. `</dl>`

实例演示如图-10所示：

Description lists	A description list is perfect for defining terms.
Euismod	Vestibulum id ligula porta felis euismod semper eget lacinia odio sem nec elit. Donec id elit non mi porta gravida at eget metus.
Malesuada porta	Etiam porta sem malesuada magna mollis euismod.
Felis euismod semper...	Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus.

图-10

步骤八：表单

[Top](#)

1)基本表单

实现基本的表单样式

```

01. <form action="" role=" form" >
02. <div class="form-group">
03. <label>电子邮件 : </label>
04. <input type="email" class="form-control" placeholder="email地址">
05. </div>
06. <div class="form-group">
07. <label>密码</label>
08. <input type="password" class="form-control" placeholder="输入密码">
09. </div>
10. <input class=" btn btn- primary" type=" submit" value=" 提交" >
11. </form>

```

实例演示如图-11所示：

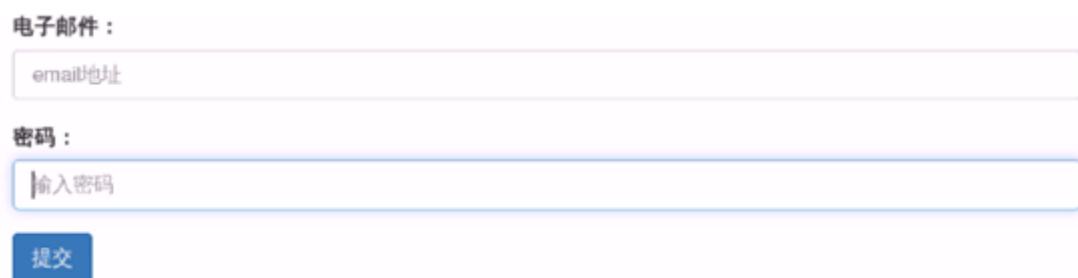


图-11

2)内联表单

让表单左对齐浮动，并表现为inline-block 内联块结构，为上面代码form元素增加类

```

01. <form class="form-inline">

```

实例演示如图-12所示：

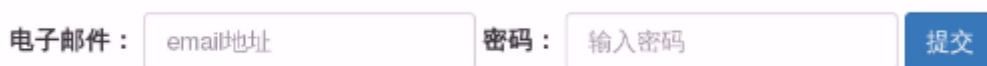


图-12

3)表单合组

前后增加片段

```

01. <form action="" role=" form" >
02. <div class="input-group">
03. <div class="input-group-addon">¥ </div>

```

[Top](#)

```

04.      <input type="text" class="form-control">
05.      <div class="input-group-addon">.00</div>
06.      </div>
07.  </form>

```

实例演示如图-13所示：



图-13

4)水平排列

让表单内的元素保持水平排列

```

01.  <form class="form-horizontal">
02.      <div class="form-group">
03.          <label class="col-sm-2 control-label">电子邮件</label>
04.          <div class="col-sm-10">
05.              <input type="email" class="form-control" placeholder="请输入您的电子邮件">
06.          </div>
07.      </div>
08.  </form>

```

注：这里用到了col-sm 栅格系统，后面章节会重点讲解，而control-label 表示和父元素样式同步。

5)复选框和单选框

设置复选框，在一行

```

01.  <div class="checkbox">
02.      <label>
03.          <input type="checkbox">体育
04.      </label>
05.  </div>
06.  <div class="checkbox">
07.      <label>
08.          <input type="checkbox">音乐
09.      </label>
10.  </div>

```

[Top](#)

设置禁用的复选框

```
01. <div class="checkbox disabled">
02.     <label>
03.         <input type="checkbox" disabled>音乐
04.     </label>
05. </div>
```

设置内联一行显示的复选框

```
01. <label class="checkbox- inline">
02.     <input type="checkbox">体育
03. </label>
04. <label class="checkbox- inline disabled">
05.     <input type="checkbox" disabled>音乐
06. </label>
```

设置单选框

```
01. <div class="radio disabled">
02.     <label>
03.         <input type="radio" name="sex" disabled>男
04.     </label>
```

6) 下拉列表

设置下拉列表

```
01. <select class="form- control">
02.     <option>1</option>
03.     <option>2</option>
04.     <option>3</option>
05.     <option>4</option>
06.     <option>5</option>
07. </select>
```

[Top](#)

3 案例3：使用常用组件

3.1 问题

【参见COOKBOOK】

1. 使用常用组件为页面添加元素

3.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：字体图标

1)Bootstrap提供了一套专用于Web开发/移动开发常用的一套图标字体——Glyphicons Halflings

基本样式类 `glyphicon`

其他修饰类

2)注意事项

引入 `fonts` 文件

不要和其他组件混合使用

只对内容为空的元素起作用

```

01. <button type="button" class="btn btn-default" aria-label="Left Align">
02.   <span class="glyphicon glyphicon-align-left" aria-hidden="true"></span>
03. </button>
04.
05. <button type="button" class="btn btn-default btn-lg">
06.   <span class="glyphicon glyphicon-star" aria-hidden="true"></span> Star
07. </button>
  
```

实例演示如图-14所示：

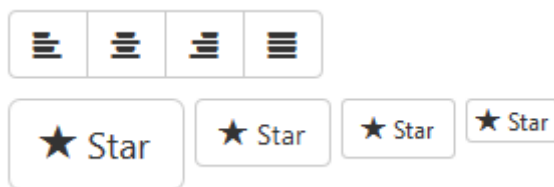


图-14

步骤二：下拉菜单

1)基本格式

将下拉菜单触发器和下拉菜单都包裹在 `.dropdown` 里，或者另一个声明了 `position: relative;` 的元素。然后加入组成菜单的 HTML 代码

[Top](#)

```
01. <div class="dropdown">
```

```

02. <button class="btn btn- default dropdown- toggle" type="button" id="dropdownMenu1" d
03.     Dropdown
04.     <span class="caret"></span>
05. </button>
06. <ul class="dropdown- menu" aria- labelledby ="dropdownMenu1">
07.     <li><a href="#">Action</a></li>
08.     <li><a href="#">Another action</a></li>
09.     <li><a href="#">Something else here</a></li>
10.     <li role="separator" class="div ider"></li>
11.     <li><a href="#">Separated link</a></li>
12. </ul>
13. </div>

```

实例演示如图-15所示：

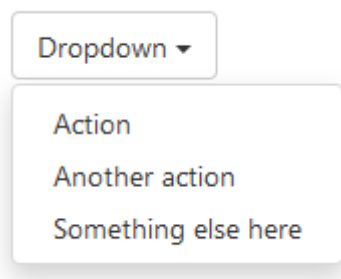


图-15

2)标题

```

01. <ul class="dropdown- menu" aria- labelledby ="dropdownMenu3">
02.     ...
03.     <li class="dropdown- header">Dropdown header</li>
04.     ...
05. </ul>

```

实例演示如图-16所示：

[Top](#)

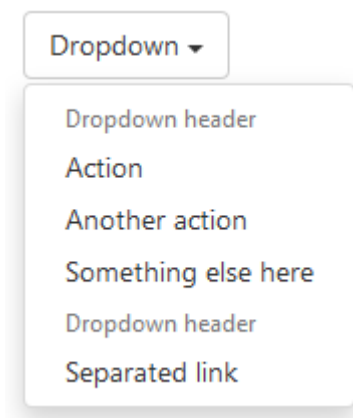


图-16

3)分割线

```

01. <ul class="dropdown-menu" aria-labelledby="dropdownMenuDivider">
02.   ...
03.   <li role="separator" class="divider"></li>
04.   ...
05. </ul>

```

实例演示如图-17所示：

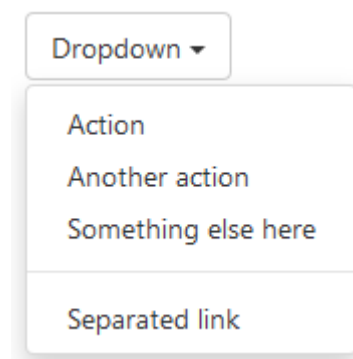


图-17

4)禁用的菜单项

为下拉菜单中的 `` 元素添加 `.disabled` 类，从而禁用相应的菜单项

```

01. <ul class="dropdown-menu" aria-labelledby="dropdownMenu4">
02.   <li><a href="#">Regular link</a></li>
03.   <li class="disabled"><a href="#">Disabled link</a></li>
04.   <li><a href="#">Another link</a></li>
05. </ul>

```

[Top](#)

实例演示如图-18所示：

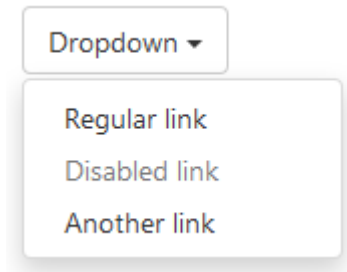


图-18

步骤三：按钮组

1)基本按钮组

```

01. <div class="btn-group" role="group" aria-label="...">
02.   <button type="button" class="btn btn-default">Left</button>
03.   <button type="button" class="btn btn-default">Middle</button>
04.   <button type="button" class="btn btn-default">Right</button>
05. </div>

```

实例演示如图-19所示：



图-19

2)按钮工具栏

```

01. <div class="btn-toolbar" role="toolbar" aria-label="...">
02.   <div class="btn-group" role="group" aria-label="...">...</div>
03.   <div class="btn-group" role="group" aria-label="...">...</div>
04.   <div class="btn-group" role="group" aria-label="...">...</div>
05. </div>

```

实例演示如图-20所示：



图-20

3)尺寸

[Top](#)

```

01. <div class="btn-group btn-group-lg" role="group" aria-label="...">...</div>
02. <div class="btn-group" role="group" aria-label="...">...</div>

```

03. `<div class="btn-group btn-group-sm" role="group" aria-label="...">...</div>`
04. `<div class="btn-group btn-group-xs" role="group" aria-label="...">...</div>`

实例演示如图-21所示：



图-21

4)垂直排列

01. `<div class="btn-group-vertical" role="group" aria-label="...">`
02. `...`
03. `</div>`

实例演示如图-22所示：

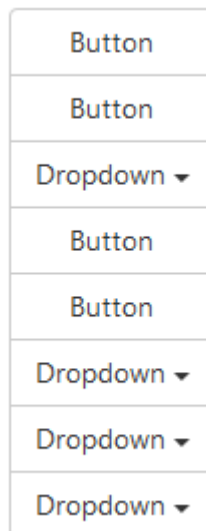


图-22

步骤四：警告框

警告框组件通过提供一些灵活的预定义消息，为常见的用户动作提供反馈消息

将任意文本和一个可选的关闭按钮组合在一起就能组成一个警告框，`.alert` 类是必须要设置的，另外我们还提供了有特殊意义的4个类（例如，`.alert-success`），代表不同的警告信息。

01. `<div class="alert alert- success" role="alert">...</div>`
02. `<div class="alert alert- info" role="alert">...</div>`
03. `<div class="alert alert- warning" role="alert">...</div>`
04. `<div class="alert alert- danger" role="alert">...</div>`

实例演示如图-23所示：

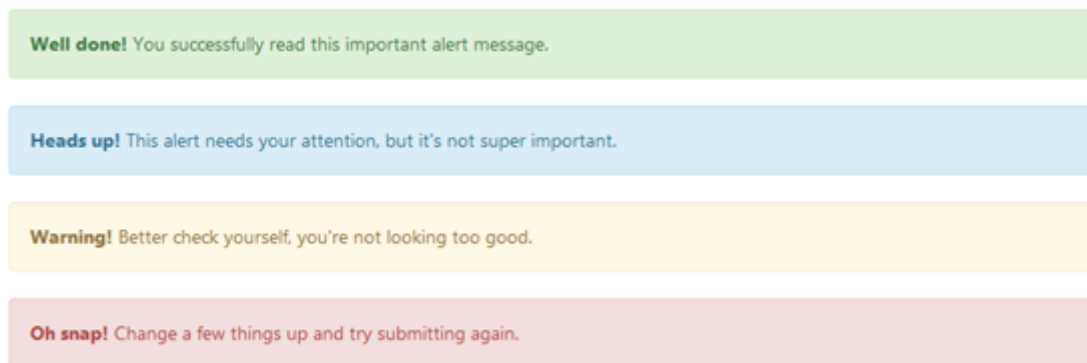


图-23

4 例4：使用常用插件

4.1 问题

【参见COOKBOOK】

1. 使用常用插件为页面添加元素

4.2 步骤

实现此案例需要按照如下步骤进行。

步骤一：标签页

1) 标签页

通过data属性:添加data-toggle="tab"或data-toggle="pill"到锚文本链接中

添加nav 和nav-tabs 类到ul中，将会应用Bootstrap标签样式，添加nav和 nav-pills类到ul中，将会应用Bootstrap胶囊式样式

01. `<ul class="nav nav- tabs">`
02. `<li class="active">Home`
03. `Prof ile`
04. `Dropdown`
05. ``

[Top](#)

实例演示如图-24所示：

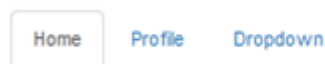


图-24

2) 淡入淡出效果

如果需要为标签页设置淡入淡出效果，请添加 `.fade` 到每个 `.tab-pane` 后面。第一个标签页必须添加 `.in` 类，以便淡入显示初始内容

```

01. <div class="tab-content">
02.   <div class="tab-pane fade in active" id="home">...</div>
03.   <div class="tab-pane fade" id="svn">...</div>
04.   <div class="tab-pane fade" id="ios">...</div>
05.   <div class="tab-pane fade" id="java">...</div>
06. </div>

```

3) 折叠框

`data-toggle="collapse"` 添加到您想要展开或折叠的组件的链接上。

`href` 或 `data-target` 属性添加到父组件，它的值是子组件的 `id`。

`data-parent` 属性把折叠面板 (`accordion`) 的 `id` 添加到要展开或折叠的组件链接上。

```

01. <div class="panel-group" id="accordion">
02.   <div class="panel panel-default">
03.     <div class="panel-heading">
04.       <h4 class="panel-title">
05.         <a data-toggle="collapse" data-parent="#accordion"
06.           href="#collapseOne">
07.           点击我进行展开，再次点击我进行折叠。第 1 部分
08.         </a>
09.       </h4>
10.     </div>
11.     <div id="collapseOne" class="panel-collapse collapse in">
12.       <div class="panel-body">
13.         Nihil anim keffiyeh helvetica, craft beer labore wes anderson
14.         cred nesciunt sapiente ea proident. Ad vegan excepteur butcher
15.         vice lomo.
16.       </div>
17.     </div>
18.   </div>
19.   <div class="panel panel-default">
20.     <div class="panel-heading">

```

[Top](#)

```
21.     <h4 class="panel- title">
22.         <a data- toggle="collapse" data- parent="#accordion"
23.             href="#collapseTwo">
24.             点击我进行展开，再次点击我进行折叠。第 2 部分
25.         </a>
26.     </h4>
27. </div>
28. <div id="collapseTwo" class="panel- collapse collapse">
29.     <div class="panel- body ">
30.         Nihil anim keffiy eh helvetica, craft beer labore wes anderson
31.         cred nesciunt sapiente ea proident. Ad vegan excepteur butcher
32.         vice lomo.
33.     </div>
34. </div>
35. </div>
36. <div class="panel panel- default">
37.     <div class="panel- heading">
38.         <h4 class="panel- title">
39.             <a data- toggle="collapse" data- parent="#accordion"
40.                 href="#collapseThree">
41.                 点击我进行展开，再次点击我进行折叠。第 3 部分
42.             </a>
43.         </h4>
44.     </div>
45.     <div id="collapseThree" class="panel- collapse collapse">
46.         <div class="panel- body ">
47.             Nihil anim keffiy eh helvetica, craft beer labore wes anderson
48.             cred nesciunt sapiente ea proident. Ad vegan excepteur butcher
49.             vice lomo.
50.         </div>
51.     </div>
52. </div>
53. </div>
```

实例演示如图-25所示：

[Top](#)

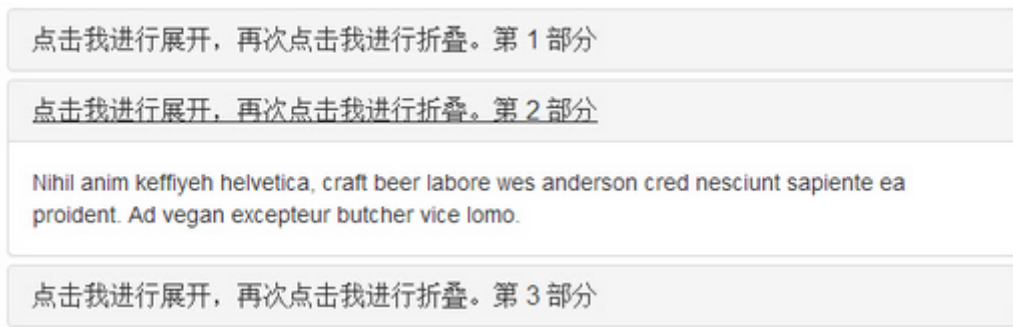


图-25

3)工具提示

提示工具 (Tooltip) 插件根据需求生成内容和标记，默认情况下是把提示工具 (tooltip) 放在它们的触发元素后面。

通过 data 属性：如需添加一个提示工具 (tooltip)，只需向一个标签添加 data-toggle="tooltip" 即可。标签的 title 即为提示工具 (tooltip) 的文本。默认情况下，插件把提示工具 (tooltip) 设置在顶部。

01. `<button data-toggle="tooltip" title="默认的 Tooltip"> 默认的 Tooltip</button>`
02. `<button data-toggle="tooltip" data-placement="left" title="左侧的 Tooltip">左侧的 Toolt`
03. `<button data-toggle="tooltip" data-placement="top" title="顶部的 Tooltip">顶部的 Toolt`
04. `<button data-toggle="tooltip" data-placement="bottom" title="底部的 Tooltip">底部的 T`
05. `<button data-toggle="tooltip" data-placement="right" title="右侧的 Tooltip">右侧的 Tool`



提示工具 (Tooltip) 插件不像之前所讨论的下拉菜单及其他插件那样，它不是纯 CSS 插件。如需使用该插件，您必须使用 jquery 激活它 (读取 javascript)。使用下面的脚本来启用页面中的所有的提示工具 (tooltip)：

01. `<script>`
02. `$(function() { $('[data-toggle="tooltip"]').tooltip(); });`
03. `</script>`

实例演示如图-26、图-27所示：

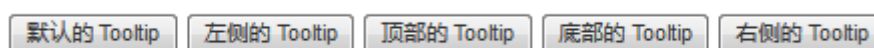


图-26



图-27

[Top](#)

4)弹出框

弹出框 (Popover) 插件根据需求生成内容和标记，默认情况下是把弹出框 (popover) 放在它们的触发元素后面。

通过 data 属性：如需添加一个弹出框 (popover)，只需向一个锚/按钮标签添加 data-toggle="popover" 即可。锚的 title 即为弹出框 (popover) 的文本。默认情况下，插件把弹出框 (popover) 设置在顶部。。

```
01. <a href="#" data-toggle="popover" title="请悬停在我的上面">
02.     请悬停在我的上面
03. </a>
```

弹出框 (Popover) 插件不像之前所讨论的下拉菜单及其他插件那样，它不是纯 CSS 插件。如需使用该插件，您必须使用 jquery 激活它 (读取 javascript)。使用下面的脚本来启用页面中的所有弹出框 (popover)：

```
01. <script>
02.     $(function() { $('[data-toggle="popover"]').popover(); });
03. </script>
```

实例演示如图-28所示：

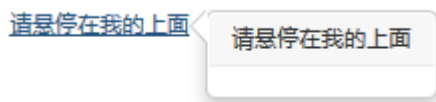


图-28

[Top](#)