Bootstrap的JavaScript插件

bootstrap.js 和 bootstrap.min.js 都包含了所有插件，你在使用时，只需选择一个引入页面就可以了。jQuery必须在所有插件**之前**引入页面

# 动画过渡效果

对于简单的过渡效果，只需将 transition.js 和其它 JS 文件一起引入即可。如果你使用的是编译（或压缩）版的 bootstrap.js 文件，就无需再单独将其引入了。

# 模态弹窗

|  |
| --- |
| <!-- Button trigger modal -->  <button type="button" class="btn btn-primary btn-lg" data-toggle="modal" data-target="#myModal">  Launch demo modal  </button>  <!-- Modal -->  <div class="**modal** fade" id="myModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">  <div class="**modal-dialog**">  <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**">  ...  </div>  <div class="modal-footer">  <button type="button" class="btn btn-default" **data-dismiss**="modal">Close</button>  <button type="button" class="btn btn-primary">Save changes</button>  </div>  </div>  </div>  </div> |

## 声明式用法

|  |
| --- |
| <!—触发元素-->  <button type="button" class="btn btn-primary " **data-toggle= "modal"** **data-target="#myModal"**>  提交  </button>  <a class="btn btn-primary " **data-toggle="modal"** **data-target="#myModal"**>  提交  </a>  //必须将data-toggle设置为modal，data-target设置为选择器(  #id或者唯一的css样式)  <!—弹窗内容-->  <div class="**modal** fade" **id="myModal**">  …  </div>  说明：属性设置可以在触发元素上也可以在弹窗最外层div元素上。常见属性有data-backdrop(true默认单击背景时候关闭弹窗，static单击背景不关闭弹窗)，data-keyboard(true默认按Esc关闭弹窗,false不关闭弹窗)，  data-show(默认true初始化时候是否显示) |

## JavaScript用法

### 属性

| **名称** | **类型** | **默认值** | **描述** |
| --- | --- | --- | --- |
| backdrop | boolean | true |  |
| keyboard | boolean | true | 键盘上的 esc 键被按下时关闭模态框。 |
| show | boolean | true | 模态框初始化之后就立即显示出来。 |

### 方法

|  |
| --- |
| $('#myModal').modal('toggle');//反转弹窗状态  $('#myModal').modal('show');//显示弹窗  $('#myModal').modal('hide');//关闭弹窗 |

### 事件

| **事件类型** | **描述** |
| --- | --- |
| show.bs.modal | show 方法调用之后立即触发该事件。如果是通过点击某个作为触发器的元素， 则此元素可以通过事件的 relatedTarget 属性进行访问。 |
| shown.bs.modal | 此事件在模态框已经显示出来（并且同时在 CSS 过渡效果完成）之后被触发。 如果是通过点击某个作为触发器的元素，则此元素可以通过事件的 relatedTarget  属性进行访问。 |
| hide.bs.modal | hide 方法调用之后立即触发该事件。 |
| hidden.bs.modal | 此事件在模态框被隐藏（并且同时在 CSS 过渡效果完成）之后被触发。 |
| loaded.bs.modal | 从远端的数据源加载完数据之后触发该事件。 |

|  |
| --- |
| $('#myModal').on('hidden.bs.modal', function (e) {  // do something...  }) |

# 下拉菜单

# 滚动侦测

# 选项卡

# 提示框

默认触发事件：hover和focus

## 声明式用法

出于性能考虑，Bootstrap默认不支持提示框使用声明式用法。可以使用如下代码启用： $('[data-toggle=tooltip]').tooltip();

|  |
| --- |
| <button type="button" class="btn btn-default" **data-toggle= "tooltip"**  **data-placement="left"** title="Tooltip on left">Tooltip on left</button>  <button type="button" class="btn btn-default" data-toggle="tooltip" data-placement="top" title="Tooltip on top">Tooltip on top</button>  //常用属性有**data-placement(**tooltip显示位置**top|left|right|bottom)**  **title,data-original-title(**设置提示语显示内容**)** |

## JavaScript用法

### 属性和方法

|  |
| --- |
| #属性使用  #('#element').tooltip(options);  $('#element').tooltip({  title:"提示语内容",  placement:'top'  });  #方法  #('#element').tooltip('show');  #('#element').tooltip('hide');  #('#element').tooltip('toggle'); |

### 事件

| **属性名称** | **描述** |
| --- | --- |
| show.bs.tooltip | Show方法调用时候立即触发（未显示前） |
| shown.bs.tooltip | 提示框完全显示给用户是后触发 |
| hide.bs.tooltip | Hide方法调用时立即触发（未关闭时） |
| hidden.bs.tooltip | 提示窗完全隐藏后触发 |

例子：$('#myTooltip').on('hidden.bs.tooltip', function () { // 逻辑… })

# 弹出窗

默认触发事件click，popover插件继承tooltip

## 声明式用法

由于性能的原因，工具提示和弹出框的 data 编程接口（data api）是必须要手动初始化的。**$('[data-toggle="popover"]').popover()**

|  |
| --- |
| <button type="button" class="btn btn-lg btn-danger" **data-toggle="popover"** **title**="Popover title" **data-content**="And here's some amazing content. It's very engaging. Right?">  点我弹出/隐藏弹出框  </button>  //data-toggle=**"popover"**说明是弹出窗，**title**表示弹出窗标题，**data-content**表示弹出窗内容。**data-trigger**表示如何触发弹出窗(click,hover.) |

## JavaScript用法

### 属性和方法

|  |
| --- |
| #属性使用  #('#element').popover(options);  $('#element').popover({  title:"弹出窗标题",  content: "弹出窗内容",  placement:'top'  });  #方法  #('#element').popover('show');  #('#element').popover('hide');  #('#element'). popover('toggle'); |

### 事件

| **类型** | **描述** |
| --- | --- |
| show.bs.popover | Show方法调用时候立即触发（未显示前） |
| shown.bs.popover | 弹出框完全显示给用户是后触发 |
| hide.bs.popover | Hide方法调用时立即触发（未关闭时） |
| hidden.bs.popover | 弹出窗完全隐藏后触发 |

$('#myPopover').on('hidden.bs.popover', function () { // 代码逻辑… })

# 警告框插件

## 声明式用法

|  |
| --- |
| <div **class="alert alert-danger alert-dismissible fade in"** role= "alert">  <button type="button" class="close" **data-dismiss="alert"** aria-label="Close"><span aria-hidden="true">×</span></button>  <h4>Oh snap! You got an error!</h4>  <p>Change this and that and try again. Duis mollis, est non commodo luctus, nisi erat porttitor ligula, eget lacinia odio sem nec elit. Cras mattis consectetur purus sit amet fermentum.</p>  <p>  <button type="button" class="btn btn-danger" **data-dismiss="alert"**>关闭</button>  </p>  </div> |

## JavaScript用法

### 方法

|  |
| --- |
| **$().alert()**：让警告框监听具有 data-dismiss="alert" 属性的后裔元素的点击（click）事件。（如果是通过 data 属性进行的初始化则无需使用）  **$().alert('close')**：关闭警告框并从 DOM 中将其删除。如果警告框被赋予了 .fade 和 .in 类，那么，警告框在淡出之后才会被删除。 |

### 事件

| **事件类型** | **描述** |
| --- | --- |
| close.bs.alert | 当 close 方法被调用后立即触发此事件。 |
| closed.bs.alert | 当警告框被关闭后（也即 CSS 过渡效果完毕之后）立即触发此事件。 |

$('#myAlert').on('closed.bs.alert', function () { // do something… })

# 按钮

## 声明式用法

|  |
| --- |
| #支持按钮，复选框和单选框  <button type="button" id="myButton" **data-loading-text="Loading..."** **data-toggle="button"** class="btn btn-primary" autocomplete="off">  Loading state  </button> |

## JavaScript用法

|  |
| --- |
| $( '#id').button('toggle')  $( '#id').button('reset') //重置状态 |

# 折叠

# 旋转轮播