

# 一、validate插件

## 1.1 validate概述

- validate: jquery的一个插件,依赖jquery使用, Validation是历史最悠久的jQuery插件之一, 经过了全球范围内不同项目的验证, 并得到了许多Web开发者的好评。作为一个标准的验证方法库, Validation拥有如下特点:
- 内置验证规则: 拥有必填、数字、Email、URL和信用卡号码等19类内置验证规则
- 自定义验证规则: 可以很方便地自定义验证规则
- 简单强大的验证信息提示: 默认了验证信息提示, 并提供自定义覆盖默认的提示信息的功能
- 实时验证: 可能通过keyup或blur事件触发验证, 而不仅仅在表单提交的时候验证

## 1.2 validate使用步骤

- 导入jquery文件
- 导入validate.js
- 页面加载成功后!对表单进行验证! \$("选择器").validate();
- 在validate中编写校验规则

```
$("选择器").validate(  
    rules:{},  
    messages:{}  
)
```

# 二、Bootstrap

## 2.1 bootstrap

- bootstrap简介:  
bootstrap 是一个用于快速开发 Web 应用程序和网站的前端框架。bootstrap 是基于 HTML、CSS、JAVASCRIPT 的。bootstrap 是由 Twitter 的 Mark Otto 和 Jacob Thornton 开发的。bootstrap 是 2011 年八月在 GitHub 上发布的开源产品。
- bootstrap特点: 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。
- 浏览器支持:
  - 所有的主流浏览器都支持 bootstrap。
  - Internet Explorer Firefox Opera Google Chrome Safari
  - 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。
  - 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。

## 2.2案例:响应式的页面

- 需求:
  - 创建一套页面!根据上网设备的不同自动调节显示的效果!
  - m.taobao.com
  - taobao.com
- bootstrap的使用步骤:
  - 下载bootstrap: <http://www.bootcss.com/>
  - 导入bootstrap.css文件
  - 导入jquery.js
  - 导入bootstrap.js
  - 将所有的内容放到布局容器中

```
.container 类用于固定宽度并支持响应式布局的容器。  
<div class="container">  
    ...  
</div>  
.container-fluid 类用于 100% 宽度, 占据全部视口 (viewport) 的容器。  
<div class="container-fluid">  
    ...  
</div>
```

## 2.3bootstrap的组成

- 全局css的样式
- 组件 I
- js插件

## 2.4栅格系统

什么是栅格系统?

- 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。下面就介绍一下 Bootstrap 栅格系统的工作原理：
  - “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（alignment）和内补（padding）。
  - 通过“行（row）”在水平方向创建一组“列（column）”。
  - 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。
  - 类似 .row 和 .col-xs-4 这种预定义的类，可以用来快速创建栅格布局。Bootstrap 源码中定义的 mixin 也可以用来创建语义化的布局。
  - 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。
  - 负值的 margin 就是下面的示例为什么是向外突出的原因。在栅格列中的内容排成一行。
- 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。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。
- 栅格类适用于与屏幕宽度大于或等于分界点大小的设备，并且针对小屏幕设备覆盖栅格类。因此，在元素上应用任何 .col-md-\* 栅格类适用于与屏幕宽度大于或等于分界点大小的设备，并且针对小屏幕设备覆盖栅格类。因此，在元素上应用任何 .col-lg-\* 激活 Windows

## 媒体查询功能？

- 判断是什么上网设备
  - 小屏幕（平板，大于等于 768px） @media (min-width: @screen-sm-min) { ... }
  - 中等屏幕（桌面显示器，大于等于 992px） @media (min-width: @screen-md-min) { ... }
  - 大屏幕（大桌面显示器，大于等于 1200px） @media (min-width: @screen-lg-min) { ... } col-lg-2
  - 大屏幕 大于1200 col-lg-2
  - 中屏幕 大于992<1200 col-md-3 I
  - 小屏幕 大于768<922 col-sm-6
  - 最小屏 小于768 col-xs-12
  - 超小屏幕 手机 (<768px)
  - 小屏幕 平板 (≥768px)
  - 中等屏幕 桌面显示器 (≥992px)
  - 大屏幕 大桌面显示器 (≥1200px)
- 栅格系统行为 总是水平排列，开始是堆叠在一起的，当大于这些阈值时将变为水平排列
- .container 最大宽度 None (自动) 750px 970px 1170px
- 类前缀 .col-xs- .col-sm- .col-md- .col-lg-
- 可以在 class 中拼接多个列数限制

## 2.5排版

标题: HTML 中的所有标题标签, `<h1>` 到 `<h6>` 均可使用。另外, 还提供了 `.h1` 到 `.h6` 类, 为的是给内联 (inline) 属性的文本赋予标题的样式。

```
<h1>我是h1</h1>我是跟随者  
<!-- 要写成行内快-->  
<span class="h1">我是h1</span>我是跟随者
```

对齐: 通过文本对齐类, 可以简单方便的将文字重新对齐。

```
<p class="text-left">Left aligned text.</p>    I  
<p class="text-center">Center aligned text.</p>  
<p class="text-right">Right aligned text.</p>  
<p class="text-justify">Justified text.</p>  
<p class="text nowrap">No wrap text.</p>
```

改变大小写: `text-lowercase` 或 `text-uppercase` 或 `text-capitalize`

```
<p class="text-uppercase">Uppercased text.</p>
```

## 2.6表格

- `table` 表格
- `table-striped` 表格隔行变色
- `table-hover` 悬浮变色

```
<div class="table-responsive">  
  <table class="table">  
    ...  
  </table>
```

行状态: 通过这些状态类可以为行或单元格设置颜色。

class类名	描述
<code>.active</code>	鼠标悬停在行或单元格上时所设置的颜色
<code>.success</code>	标识成功或积极的动作
<code>.info</code>	标识普通的提示信息或动作
<code>.warning</code>	标识警告或需要用户注意
<code>.danger</code>	标识危险或潜在的带来负面影响的动作

## 2.7表单

- 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。
- 单独的表单控件会被自动赋予一些全局样式。所有设置了 `.form-control` 类的 `<input>`、`<textarea>` 和 `<select>`

-把标签和控件放在一个带有 class .form-group 的 `<div>` 中。这是获取最佳间距所必需的。  
-向所有的文本元素 `<input>`、`<textarea>` 和 `<select>` 添加 class = "form-control" 。  
form-group 会将label和input上下排列  
form-control 会自动将input充满屏幕 并添加点击高亮效果

```
<form>
  <div class="form-group">
    <label for="exampleInputEmail1">Email address</label>
    <input type="email" class="form-control" id="exampleInputEmail1"
placeholder="Email">
  </div>
  <div class="form-group">
    <label for="exampleInputPassword1">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1"
placeholder="Password">
  </div>
  <div class="form-group">
    <label for="exampleInputFile">File input</label>
    <input type="file" id="exampleInputFile">
    <p class="help-block">Example block-level help text here.</p>
  </div>
  <div class="checkbox">
    <label>
      <input type="checkbox"> Check me out
    </label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```

激活 Windows 10  
转到“设置”以启动

内联表单：为 `<form>` 元素添加 `.form-inline` 类可使其内容左对齐并且表现为 inline-block 级别的控件。只适用于视口 (viewport) 至少在 768px 宽度时 (视口宽度再小的话就会使表单折叠)。

```
<form class="form-inline">
  <div class="form-group">
    <label for="exampleInputName2">Name</label>
    <input type="text" class="form-control" id="exampleInputName2" placeholder="Jane Doe">
  </div>
  <div class="form-group">
    <label for="exampleInputEmail2">Email</label>
    <input type="email" class="form-control" id="exampleInputEmail2"
placeholder="jane.doe@example.com">
  </div>
  <button type="submit" class="btn btn-default">Send invitation</button>
</form>
```

激活 Windows 10

水平表单：通过为表单添加 `.form-horizontal` 类，并联合使用 Bootstrap 预置的栅格类，可以将 label 标签和控件组水平并排布局。这样做将改变 `.form-group` 的行为，使其表现为栅格系统中的行 (row)，因此就无需再额外添加 `.row` 了。

```
<form class="form-horizontal">
  <div class="form-group">
    <label for="inputEmail3" class="col-sm-2 control-label">Email</label>
    <div class="col-sm-10">
      <input type="email" class="form-control" id="inputEmail3" placeholder="Email">
    </div>
  </div>
  <div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-label">Password</label>
    <div class="col-sm-10">
      <input type="password" class="form-control" id="inputPassword3"
placeholder="Password">
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <div class="checkbox">
        <label>
          <input type="checkbox"> Remember me
        </label>
      </div>
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-10">
      <button type="submit" class="btn btn-default">Sign in</button>
    </div>
  </div>
</form>
```

激活 Windc  
转到“设置”以激

## 案例 显示带引导的

```
<form class="form-inline">
  <div class="form-group">
    <label class="sr-only" for="exampleInputAmount">Amount (in dollars)</label>
    <div class="input-group">
      <div class="input-group-addon">$</div>
      <input type="text" class="form-control" id="exampleInputAmount"
placeholder="Amount">
      <div class="input-group-addon">.00</div>
    </div>
    <button type="submit" class="btn btn-primary">Transfer cash</button>
  </div>
</form>
```

通过将 .checkbox-inline 或 .radio-inline 类应用到一系列的多选框（checkbox）或单选框（radio）控件上，可以使这些控件排列在一行。

```
<label class="checkbox-inline">
    <input type="checkbox" id="inlineCheckbox1" value="option1"> 1
</label>
<label class="checkbox-inline">
    <input type="checkbox" id="inlineCheckbox2" value="option2"> 2
</label>
<label class="checkbox-inline">
    <input type="checkbox" id="inlineCheckbox3" value="option3"> 3
</label>
<label class="radio-inline">
    <input type="radio" name="inlineRadioOptions" id="inlineRadio1" value="option1"> 1
</label>
<label class="radio-inline">
    <input type="radio" name="inlineRadioOptions" id="inlineRadio2" value="option2"> 2
</label>    I
<label class="radio-inline">
    <input type="radio" name="inlineRadioOptions" id="inlineRadio3" value="option3"> 3
</label>
```

## 下拉列表

```
<select class="form-control">
    <option>1</option>
    <option>2</option>
    <option>3</option>
    <option>4</option>
    <option>5</option>
</select>

<select multiple class="form-control">
    <option>1</option>
    <option>2</option>
    <option>3</option>
    <option>4</option>
    <option>5</option>
</select>
```

## 带分割线的下拉框

```
<div class="btn-group">
    <button id="show" type="button" class="btn btn-danger">Action</button>
    <button type="button" class="btn btn-danger dropdown-toggle" data-toggle="dropdown" aria-
haspopup="true" aria-expanded="false">
        <span class="caret"></span>
        <span class="sr-only">Toggle Dropdown</span>
    </button>
    <ul id="ul" class="dropdown-menu">
        <li><a href="#">Action</a></li>
        <li><a href="#">Another action</a></li>
        <li><a href="#">Something else here</a></li>
        <li role="separator" class="divider"></li>
        <li><a href="#">Separated link</a></li>
    </ul>
</div>
</div>
```

激活 Windows

```
<div class="dropdown">
    <!-- 下拉框 触发按钮 -->
    <button class="btn btn-success dropdown-toggle" type="button" id="dropdownMenu1" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="true">
        下拉框
        <!-- 下拉框图片-->
        <span class="glyphicon glyphicon-arrow-down"></span>
    </button>
    <!-- 下拉菜单-->
    <ul class="dropdown-menu" aria-labelledby="dropdownMenu1">
        <li><a href="#">Action</a></li>
        <li><a href="#">Another action</a></li>
        <li><a href="#">Something else here</a></li>
        <li role="separator" class="divider"></li>
        <li><a href="#">Separated link</a></li>
    </ul>
</div>
```

## 静态控件

```
<form class="form-horizontal">
    <div class="form-group">
        <label class="col-sm-2 control-label">Email</label>
        <div class="col-sm-10">
            <p class="form-control-static">email@example.com</p>
        </div>
    </div>
    <div class="form-group">
        <label for="inputPassword" class="col-sm-2 control-label">Password</label>
        <div class="col-sm-10">
            <input type="password" class="form-control" id="inputPassword"
placeholder="Password">
        </div>
    </div>
</form>
```

激活 Windows

- Bootstrap 对表单控件的校验状态，如 error、warning 和 success 状态，都定义了样式。使用时，添加 .has-warning、.has-error 或 .has-success 类到这些控件的父元素即可。
- 任何包含在此元素之内的 .control-label、.form-control 和 .help-block 元素都将接受这些校验状态的样式。

## 2.8按钮

class = btn 将任何东西变成按钮 需要配合 btn-default a标签也可以

### 1.按钮颜色

```
<!-- Standard button -->
<button type="button" class="btn btn-default"> (默认样式) Default</button>

<!-- Provides extra visual weight and identifies the primary action in a set of buttons -->
<button type="button" class="btn btn-primary"> (首选项) Primary</button>

<!-- Indicates a successful or positive action -->
<button type="button" class="btn btn-success"> (成功) Success</button>

<!-- Contextual button for informational alert messages -->
<button type="button" class="btn btn-info"> (一般信息) Info</button>

<!-- Indicates caution should be taken with this action -->
<button type="button" class="btn btn-warning"> (警告) Warning</button>

<!-- Indicates a dangerous or potentially negative action -->
<button type="button" class="btn btn-danger"> (危险) Danger</button>

<!-- Deemphasize a button by making it look like a link while maintaining button behavior -->
<button type="button" class="btn btn-link"> (链接) Link</button>
```

### 2.成组button

```
<div class="btn-group" data-toggle="buttons">
  <label class="btn btn-primary">
    <input type="radio" name="options" id="option1"> 选项 1
  </label>
  <label class="btn btn-primary">
    <input type="radio" name="options" id="option2"> 选项 2
  </label>
  <label class="btn btn-primary">
    <input type="radio" name="options" id="option3"> 选项 3
  </label>
</div>
```

需要让按钮具有不同尺寸吗？使用 .btn-lg、.btn-sm 或 .btn-xs 就可以获得不同尺寸的按钮

```
<p>
  <button type="button" class="btn btn-primary btn-lg"> (大按钮) Large button</button>
  <button type="button" class="btn btn-default btn-lg"> (大按钮) Large button</button>
</p>
<p>
  <button type="button" class="btn btn-primary"> (默认尺寸) Default button</button>
  <button type="button" class="btn btn-default"> (默认尺寸) Default button</button>
</p>
<p>
  <button type="button" class="btn btn-primary btn-sm"> (小按钮) Small button</button>
  <button type="button" class="btn btn-default btn-sm"> (小按钮) Small button</button>
</p>
<p>
  <button type="button" class="btn btn-primary btn-xs"> (超小尺寸) Extra small button</button>
  <button type="button" class="btn btn-default btn-xs"> (超小尺寸) Extra small button</button>
</p>
```

- 按钮组：把一系列的.btn按钮放入.btn-group。
- btn-group 内部嵌套 btn

```
<div class="btn-group" role="group" aria-label="...>
  <button type="button" class="btn btn-default">Left</button>
  <button type="button" class="btn btn-default">Middle</button>
  <button type="button" class="btn btn-default">Right</button>
</div>
```

按钮工具栏：把一组 `<div class="btn-group">` 组合进一个 `<div class="btn-toolbar">` 中就可以做成更复杂的组件

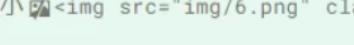
```
<div class="btn-toolbar" role="toolbar" aria-label="...>
  <div class="btn-group" role="group" aria-label="...>...</div>
  <div class="btn-group" role="group" aria-label="...>...</div>
  <div class="btn-group" role="group" aria-label="...>...</div>
</div>
```

尺寸：只要给 .btn-group 加上 .btn-group-\* 类，就省去为按钮组中的每个按钮都赋予尺寸类了，如果包含了多个按钮组时也适用。

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

激活 Windows  
转到“设置”以激活

## 2.9图片形状

- 响应式图片!随着窗体大小改变大小
- 通过为元素添加以下相应的类,可以让图片呈现不同的形状。

```
  
  

```

## 2.10 导航栏

data-toggle="tab"

倒航条自带响应效果 缩小显示成手机效果

```
<li role="presentation" class="active"><a href="#" data-toggle="tab">Home</a></li>
```

### 标签式导航栏

```
<li role="presentation" class="active"><a href="#" data-toggle="tab">Home</a></li>  
<ul class="nav nav-tabs">  
    <li role="presentation" class="active"><a href="#" data-toggle="tab">Home</a></li>  
    <li role="presentation"><a href="#" data-toggle="tab">Profile</a></li>  
    <li role="presentation"><a href="#" data-toggle="tab">Messages</a></li>  
</ul>    [  
  
<div id="myTabContent" class="tab-content">  
    <div class="tab-pane fade in active" id="h5">  
        <p>Html5最近比较火</p>  
    </div>  
    <div class="tab-pane fade" id="java">  
        <p>java是高级语言,是最好的语言</p>  
    </div>  
    <div class="tab-pane fade" id="android">  
        <p>android是最受大众欢迎的智能机品牌</p>  
    </div>  
</div>
```

激活 Windo

## 2.11 分页

```

<nav aria-label="Page navigation">
  <ul class="pagination">
    <li>
      <a href="#" aria-label="Previous">
        <span aria-hidden="true">&laquo;</span>
      </a>
    </li>
    <li><a href="#">1</a></li>
    <li><a href="#">2</a></li>
    <li><a href="#">3</a></li>
    <li><a href="#">4</a></li>
    <li><a href="#">5</a></li>
    <li>|</li>
      <a href="#" aria-label="Next">
        <span aria-hidden="true">&raquo;</span>
      </a>
    </li>
  </ul>
</nav>

```

## 2.12缩略图

```

<div class="container" style="margin-top: 30px;">
  <div class="row">
    <div class="col-md-4">
      <a href="#" class="thumbnail"> </img></a>
      <div class="caption">
        <h4>HTML入门</h4>
        <h6>html是最好的静态网页语言</h6>
      </div>
    </div>
  </div>
</div>

```

## 2.13模态框

### 动态模态框

```

1  <!-- Button trigger modal -->
2  <button type="button" class="btn btn-primary btn-lg" data-toggle="modal"
   data-target="#myModal">
3    Launch demo modal
4  </button>
5
6  <!-- Modal -->
7  <div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-
   labelledby="myModalLabel">
8    <div class="modal-dialog" role="document">
9      <div class="modal-content">
10        <div class="modal-header">
11          <button type="button" class="close" data-dismiss="modal" aria-
   label="Close"><span aria-hidden="true">&times;</span></button>
12          <h4 class="modal-title" id="myModalLabel">Modal title</h4>
13        </div>
14        <div class="modal-body">
15          ...

```

```
16      </div>
17      <div class="modal-footer">
18          <button type="button" class="btn btn-default" data-
dismiss="modal">Close</button>
19          <button type="button" class="btn btn-primary">Save changes</button>
20      </div>
21  </div>
22 </div>
23 </div>
```