

# jQuery Validate

jQuery Validate 是jQuery最常见的插件之一。

## 表单校验优点

---

- 1.提升用户体验
- 2.缩短用户等待时间
- 3.可以减少服务器的压力

## 基本用法（1.html）

---

### 引入jQuery Validate

```
<script src="jquery.js"></script>
<script src="jquery.validate.js"></script>
// 通过引入jquery.validate.js jQuery对象及原型被扩展了一些属性及方法。

// html code
<form id="commentForm" class="col-md-4 col-md-offset-4" novalidate="novalidate">
  <div class="form-group">
    <label>UserName</label>
    <input type="text" class="form-control" name="userName" placeholder="UserName">
  </div>
  <div class="form-group">
    <label>Password</label>
    <input type="password" class="form-control" name="password" placeholder="Password">
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>

// dom加载完后执行
// js code
$(function () {
  $("#commentForm").validate({
    rules: {
      userName: {
        required:true,
        minlength:2,
        maxlength:10
      },
      password: {
        required:true,
        minlength:2,
        maxlength:16
      }
    },
    messages: {
      userName: {
        required:"此项为必填项! ",
        minlength:"用户名最小为2位",
        maxlength:"用户名最大为10位"
      },
      password: {
        required:"此项为必填项! ",
        minlength:"用户名最小为2位",
        maxlength:"用户名最大为16位"
      }
    }
  });
});
```

ps:

1. rules及messages 中的userName、password对应input表单的name属性
2. 如果messages属性没有配置错误提示信息，将使用默认的提示信息
3. validate(): 是jQuery Validate插件的核心方法，定义了基本的校验规则和一些有用的配置项,参考上面代码 **method**: 验证方法，指的是校验的逻辑。 **eg**: 比如email方法，检查输入的文本是否符合email的规则； **rule**: 验证规则，指的是元素和验证方法的关联。 **eg**: 比如页面存在一个name为email的文本框，需要带有email的验证方法；

## 默认验证方法(2.html)

**required**: 必输字段 **minlength**: 输入长度最小是？的字符串(汉字算一个字符) **maxlength**: 输入长度最多是？的字符串(汉字算一个字符)

**rangelength**: 输入长度必须介于？和？之间的字符串)(汉字算一个字符) **min**: 输入值不能小于？ **max**: 输入值不能大于？ **range**: 输入值必须介于？和？之间 **email**: 必须输入正确格式的电子邮件

**url**: 必须输入正确格式的网址

**date**: 必须输入正确格式的日期 使用 new Date()进行时间转换 日期校验ie6出错,有效性较宽泛

**dateISO**: 必须输入正确格式的日期(ISO)，例如：2009-06-23，1998/01/22 验证ISO格式，不验证有效性

**number**: 必须输入合法的数字(负数，小数)

**digits**: 必须输入整数

**url**: 必须输入正确格式的网址

**creditcard**: 必须输入合法的信用卡号

**equalTo**: 输入值必须和对应表单元素值相同

**depends**: 所有的验证规则都有depends属性用于,规定该验证什么时候有效

**remote**: 使用ajax方法验证输入值

1. ajax 方式进行验证
2. 返回: **true**: 通过验证 **false**: 没有通过验证
3. 默认发送get请求，可以通过type属性配置
4. 默认会提交当前验证的值到远程地址，如果需要提交其他的值，可以使用 data 选项

```
//get
remote: "remote.json"

// post
remote: {
  url: "remote.json",
  type: "post",
  dataType: "json",
  data: {
    time1:111,
    test: function () {
      return $("[name='email']").val();
    }
  }
}
```

## Validate方法配置项(3.html)

**debug**: 表单不会提交，只进行检查，用于调试

**submitHandler**: 表单提交通过验证后运行的函数,函数内需执行表单提交的函数,否则表单不会提交。 **br> invalidHandler**: 无效表单提交后运行的函数

**ignore**: 对某些元素不进行验证

**rules**: 定义校验规则

**messages**: 定义提示信息

**groups**: 对一组元素的验证，用一个错误提示，用errorPlacement控制把错误信息放在哪里

**onsubmit**: 是否在提交时验证

**onfocusout**: 是否在获取焦点时验证

**onkeyup**: 是否在敲击键盘时验证

**onclick**: 是否在鼠标点击时验证，一般用于checkbox或者radio

**focusInvalid**: 提交表单后，未通过验证的表单（第一个或提交之前获得焦点的未通过验证的表单）是否会获得焦点

**focusCleanup:** 当未通过验证的元素获得焦点时，是否移除错误提示

**errorClass:** 指定错误提示的css类名，可以自定义错误提示的样式

**validClass:** 指定验证通过的css类名

**errorElement:** 使用什么标签标记错误

**wrapper:** 使用什么标签将errorElement包起来

**errorLabelContainer:** 把错误信息统一放在一个容器里面

**errorContainer:** 显示或隐藏验证信息，可以自动实现有错误出现时把容器变为显示，无错误时隐藏

**showErrors:** 可以显示总共有多少个未通过验证的元素

**errorPlacement:** 自定义错误信息放到哪里

**highlight:** 给未通过验证的元素添加效果

**unhighlight:** 去除未通过验证的元素的效果，一般和highlight一起使用

## Validator对象(4.html)

**validate():** 方法返回的对象，Validator对象的方法：

```
var validator = $("#commentForm").validate();
```

**Validator.form():** 验证表单是否有效，返回true/false

**Validator.element(element):** 验证某个元素是否有效 返回true/false

**Validator.resetForm():** 将表单恢复到验证前原来的状态

**Validator.showErrors(errors):** 针对某个元素显示特定的错误信息

**Validator.numberOfInvalids():** 返回无效的表单元素数量

## jQuery.Validator对象(5.html)

**jQuery.validator.addMethod(name,method[,message]):** 增加自定义方法 **jQuery.validator.format(template,argument,argumentN...):** 格式化字符串，用参数代替模板中的{n}，主要用作验证规则的提示语。

**jQuery.validator.setDefaults(options):** 修改插件默认配置 如： debug: true

**jQuery.validator.addClassRules(name,rules):** 为某些名为name的class增加组合验证类型

## 常用方法(6.html)

**valid:** 检查表单或表单元素是否有效，返回true/false

```
// 表单校验
$("#commentForm").valid();
// 表单元素校验
$("#[name='userName']").valid();
```

**rules:** 针对表单元素的方法

1. rules() 获取表单元素的校验规则
2. rules("add",rules) 向表单元素添加校验规则
3. rules("remove",rules) 删除表单元素校验规则

```
// 1.rules()
console.log($userName.rules());
// 2.rules("add",rules)
$userName.rules("add",{
    number:true
});
console.log($userName.rules());
// 3.rules("remove",rules)
$userName.rules("remove",{
    number:true
})
```

## 自定义验证方法（7.html）

---

`jQuery.validator.addMethod(name,method[,message]):` 增加自定义方法

## 选择器的扩展与其他（8.html）

---

**:blank:** 选择所有值为空的元素（trim除全角空格）

**:filled:** 选择所有值不为空的元素(trim后)

**:unchecked:**选择所有没有被选中的元素 (checkbox radio)

**radio:** radio的required表示必须选中一个

**checkbox:** checkbox的required表示必须选中 checkbox的minlength表示必须选中的最小个数,maxlength表示最大的选中个数,rangelength:[2,3]表示选中个数区间

**select:** select的required表示选中的value不能为空