

昵称: Rex.He
园龄: 8年3个月
粉丝: 59
关注: 4
+加关注

< 2011年11月 >						
日	一	二	三	四	五	六
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

搜索

找找看

谷歌搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签
更多链接

我的标签

11/22(1)

随笔分类(80)

ASP.NET(30)
DIV/CSS(4)
Flash/Flex(3)
IIS(3)
jQuery/Javascript(15)
Linux(2)
PHP(4)
SEO(3)
SilverLight
工具-软件(5)
经验-分析(1)
其他备忘
数据库(10)

随笔档案(80)

2014年8月 (1)
2013年9月 (1)
2013年8月 (1)
2013年5月 (1)
2013年1月 (1)
2012年12月 (1)
2012年10月 (2)
2012年8月 (2)
2012年5月 (1)
2012年4月 (2)
2012年3月 (4)

jQuery验证控件jquery.validate.js使用说明+中文API

官网地址: <http://bassistance.de/jquery-plugins/jquery-plugin-validation>

jQuery plugin: Validation 使用说明

转载自: http://blog.sina.com.cn/s/blog_608475eb0100h3h1.html

一导入js库

```
<script src="../js/jquery.js" type="text/javascript"></script>  
<script src="../js/jquery.validate.js" type="text/javascript"></script>
```

二、默认校验规则

(1)required:true	必填字段
(2)remote:"check.php"	使用ajax方法调用check.php验证输入值
(3)email:true	必须输入正确格式的电子邮件
(4)url:true	必须输入正确格式的网址
(5)date:true	必须输入正确格式的日期 日期校验ie6出错, 慎用
(6)dateISO:true	必须输入正确格式的日期(ISO), 例如: 2009-06-23, 1998/01/22 只验证格式, 不验证有效性
(7)number:true	必须输入合法的数字(负数, 小数)
(8)digits:true	必须输入整数
(9)creditcard:	必须输入合法的信用卡号
(10)equalTo:"#field"	输入值必须和#field相同
(11)accept:	输入拥有合法后缀名的字符串(上传文件的后缀)
(12)maxlength:5	输入长度最多是5的字符串(汉字算一个字符)
(13)minlength:10	输入长度最小是10的字符串(汉字算一个字符)
(14)rangelength:[5,10]	输入长度必须介于 5 和 10 之间的字符串(汉字算一个字符)
(15)range:[5,10]	输入值必须介于 5 和 10 之间
(16)max:5	输入值不能大于5
(17)min:10	输入值不能小于10

三、默认的提示

```
messages: {  
    required: "This field is required.",  
    remote: "Please fix this field.",  
    email: "Please enter a valid email address.",  
    url: "Please enter a valid URL.",  
    date: "Please enter a valid date.",  
    dateISO: "Please enter a valid date (ISO).",  
    dateDE: "Bitte geben Sie ein g\u00fcltiges Datum ein.",  
    number: "Please enter a valid number.",  
    numberDE: "Bitte geben Sie eine Nummer ein.",  
    digits: "Please enter only digits",  
    creditcard: "Please enter a valid credit card number.",  
    equalTo: "Please enter the same value again.",  
    accept: "Please enter a value with a valid extension.",  
    maxlength: $.validator.format("Please enter no more than {0} characters."),  
    minlength: $.validator.format("Please enter at least {0} characters."),
```

2012年2月 (2)
2012年1月 (1)
2011年11月 (5)
2011年10月 (7)
2011年9月 (11)
2011年8月 (5)
2011年5月 (8)
2011年4月 (2)
2011年3月 (1)
2011年1月 (1)
2010年1月 (19)
2009年12月 (1)

最新评论

1. Re:jQuery验证控件jquery.validate.js
使用说明+中文API
很详细，挺好的
--xiaonuan筱暖
2. Re:jQuery验证控件jquery.validate.js
使用说明+中文API
@rainFish05用“”引号包括起来，就可以运行正常...
--刘彦佐
3. Re:jQuery验证控件jquery.validate.js
使用说明+中文API
看这个文章就好了，官网看不懂。
--狷特头
4. Re:jQuery验证控件jquery.validate.js
使用说明+中文API
您好，firstname: "required",这个firstn
ame指的是id吗？在线等。
--虚竹_风清扬
5. Re:jQuery验证控件jquery.validate.js
使用说明+中文API
关于对bootstrap-datetimepicker的验证，
如何恰当使用：时间 ' name="entity.star
ttimevalu.....
--崔俊宁

阅读排行榜

1. jQuery验证控件jquery.validate.js使
用说明+中文API(290421)
2. jquery.cookie.js[保存](7253)
3. Hyper-V虚拟机安装及网卡无法找到，
驱动无法安装解决办法(7222)
4. jquery.each读取json数据(6822)
5. IIS7-URL重写(URL Rewrite Module)
(5390)

评论排行榜

1. jQuery验证控件jquery.validate.js使
用说明+中文API(36)
2. Hyper-V虚拟机安装及网卡无法找到，
驱动无法安装解决办法(3)
3. xampp无法启动apache，提示termin
ating worker thread 0(3)
4. 使用纯真版IP地址库，根据IP判断所在
地(1)
5. Assembly.Load()不能加载程序集的问题(1)

推荐排行榜

1. jQuery验证控件jquery.validate.js使
用说明+中文API(63)
2. xampp无法启动apache，提示termin
ating worker thread 0(2)
3. IIS7-URL重写(URL Rewrite Module)
(1)
4. .NET下URL重写及PostBack处理(1)
5. jquery.cookie.js[保存](1)

```
    rangelength: $.validator.format("Please enter a value between {0} and {1} characters long."),  
    range: $.validator.format("Please enter a value between {0} and {1}."),  
    max: $.validator.format("Please enter a value less than or equal to {0}."),  
    min: $.validator.format("Please enter a value greater than or equal to {0}."),  
},
```

如需要修改，可在js代码中加入：

```
jQuery.extend(jQuery.validator.messages, {  
    required: "必选字段",  
    remote: "请修正该字段",  
    email: "请输入正确格式的电子邮件",  
    url: "请输入合法的网址",  
    date: "请输入合法的日期",  
    dateISO: "请输入合法的日期 (ISO).",  
    number: "请输入合法的数字",  
    digits: "只能输入整数",  
    creditcard: "请输入合法的信用卡号",  
    equalTo: "请再次输入相同的值",  
    accept: "请输入拥有合法后缀名的字符串",  
    maxlength: jQuery.validator.format("请输入一个 长度最多是 {0} 的字符串"),  
    minlength: jQuery.validator.format("请输入一个 长度最少是 {0} 的字符串"),  
    rangelength: jQuery.validator.format("请输入 一个长度介于 {0} 和 {1} 之间的字符串"),  
    range: jQuery.validator.format("请输入一个介于 {0} 和 {1} 之间的值"),  
    max: jQuery.validator.format("请输入一个最大为{0} 的值"),  
    min: jQuery.validator.format("请输入一个最小为{0} 的值")  
});
```

推荐做法，将此文件放入messages_cn.js中，在页面中引入

```
<script src="../../js/messages_cn.js" type="text/javascript"></script>
```

四、使用方式

1. 将校验规则写到控件中



```
<script src="../../js/jquery.js" type="text/javascript"></script>  
<script src="../../js/jquery.validate.js" type="text/javascript"></script>  
<script src="../../js/jquery.metadata.js" type="text/javascript"></script>  
  
$(function() {  
    $("#signupForm").validate();  
});  
  
<form id="signupForm" method="get" action="">  
    <p>  
        <label for="firstname">Firstname</label>  
        <input id="firstname" name="firstname" class="required" />  
    </p>  
    <p>  
        <label for="email">E-Mail</label>  
        <input id="email" name="email" class="required email" />  
    </p>  
    <p>  
        <label for="password">Password</label>  
        <input id="password" name="password" type="password" class="{required:true,minlength:5}" />  
    </p>  
    <p>  
        <label for="confirm_password">确认密码</label>  
        <input id="confirm_password" name="confirm_password" type="password" class="{  
required:true,minlength:5,equalTo:'#password'}" />  
    </p>  
    <p>  
        <input class="submit" type="submit" value="Submit"/>  
    </p>  
</form>
```

```
</p>
</form>
```



使用class="{}"的方式，必须引入包：jquery.metadata.js

可以使用如下的方法，修改提示内容：

class="{required:true,minlength:5,messages:{required:'请输入内容'}}"

在使用equalTo关键字时，后面的内容必须加上引号，如下代码：

class="{required:true,minlength:5,equalTo:'#password'}"

2.将校验规则写到js代码中



```
$(function() {
    $('#signupForm').validate({
        rules: {
            firstname: "required",
            email: {
                required: true,
                email: true
            },
            password: {
                required: true,
                minlength: 5
            },
            confirm_password: {
                required: true,
                minlength: 5,
                equalTo: "#password"
            }
        },
        messages: {
            firstname: "请输入姓名",
            email: {
                required: "请输入Email地址",
                email: "请输入正确的email地址"
            },
            password: {
                required: "请输入密码",
                minlength: jQuery.format("密码不能小于{0}个字符")
            },
            confirm_password: {
                required: "请输入确认密码",
                minlength: "确认密码不能小于5个字符",
                equalTo: "两次输入密码不一致"
            }
        }
    });
});
```



//messages处，如果某个控件没有message，将调用默认的信息



```
<form id="signupForm" method="get" action="">
    <p>
        <label for="firstname">Firstname</label>
        <input id="firstname" name="firstname" />
    </p>
    <p>
        <label for="email">E-Mail</label>
        <input id="email" name="email" />
    </p>
```

```

<p>
  <label for="password">Password</label>
  <input id="password" name="password" type="password" />
</p>
<p>
  <label for="confirm_password">确认密码</label>
  <input id="confirm_password" name="confirm_password" type="password" />
</p>
  <p>
    <input class="submit" type="submit" value="Submit"/>
  </p>
</form>

```



required:true 必须有值

required:"#aa:checked"表达式的值为真，则需要验证

required:function(){ }返回为真，表时需要验证

后边两种常用于，表单中需要同时填或不填的元素

五、常用方法及注意事项

1.用其他方式替代默认的SUBMIT

```

$.ready(function() {
  $("#signupForm").validate({
    submitHandler:function(form){
      alert("submitted");
      form.submit();
    }
  });
});

```

使用ajax方式

```

$(".selector").validate({
  submitHandler: function(form)
  {
    $(form).ajaxSubmit();
  }
})

```

可以设置validate的默认值，写法如下：

```

$.validator.defaults({
  submitHandler: function(form) { alert("submitted!");form.submit(); }
});

```

如果想提交表单，需要使用form.submit()而不要使用\$(form).submit()

2.debug，只验证不提交表单

如果这个参数为true，那么表单不会提交，只进行检查，调试时十分方便

```

$.ready(function() {
  $("#signupForm").validate({
    debug:true
  });
});

```

如果一个页面中有多个表单都想设置成为debug，用

```

$.validator.defaults({
  debug: true
})

```

3.ignore：忽略某些元素不验证

ignore: ".ignore"

4.更改错误信息显示的位置

errorPlacement : Callback

Default: 把错误信息放在验证的元素后面

指明错误放置的位置，默认情况是：error.appendTo(element.parent());即把错误信息放在验证的元素后面

```
errorPlacement: function(error, element) {  
    error.appendTo(element.parent());  
}
```

//示例：

```
  
<tr>  
    <td class="label"><label id="lfirstname" for="firstname">First Name</label></td>  
    <td class="field">  
<input id="firstname" name="firstname" type="text" value="" maxlength="100" /></td>  
    <td class="status"></td>  
</tr>  
<tr>  
    <td style="padding-right: 5px;">  
        <input id="dateformat_eu" name="dateformat" type="radio" value="0" />  
        <label id="ldateformat_eu" for="dateformat_eu">14/02/07</label>  
    </td>  
    <td style="padding-left: 5px;">  
        <input id="dateformat_am" name="dateformat" type="radio" value="1" />  
        <label id="ldateformat_am" for="dateformat_am">02/14/07</label>  
    </td>  
</tr>  
<tr>  
    <td class="label">&nbsp;</td>  
    <td class="field" colspan="2">  
        <div id="termswrap">  
            <input id="terms" type="checkbox" name="terms" />  
            <label id="lterms" for="terms">I have read and accept the Terms of Use.</label>  
        </div>  
    </td>  
</tr>  
  
errorPlacement: function(error, element) {  
    if ( element.is(":radio") )  
        error.appendTo( element.parent().next().next() );  
    else if ( element.is(":checkbox") )  
        error.appendTo ( element.next() );  
    else  
        error.appendTo( element.parent().next() );  
}
```



代码的作用是：一般情况下把错误信息显示在<td class="status"></td>中，如果是radio显示在<td></td>中，如果是 checkbox显示在内容的后面

errorClass : String Default: "error"

指定错误提示的css类名，可以自定义错误提示的样式

errorElement : String Default: "label"

用什么标签标记错误，默认的是label你可以改成em

errorContainer : Selector

显示或者隐藏验证信息，可以自动实现有错误信息出现时把容器属性变为显示，无错误时隐藏，用处不大

errorContainer: "#messageBox1, #messageBox2"

errorLabelContainer : Selector

把错误信息统一放在一个容器里面。

wrapper : String

用什么标签再把上边的errorElement包起来

一般这三个属性同时使用，实现在一个容器内显示所有错误提示的功能，并且没有信息时自动隐藏

```
errorContainer: "div.error",
errorLabelContainer: $("#signupForm div.error"),
wrapper: "li"
```

5更改错误信息显示的风格

设置错误提示的风格，可以增加图标显示，在该系统中已经建立了一个validation.css专门用于维护校验文件的风格

```
input.error { border: 1px solid red; }
label.error {
    background:url("../demo/images/unchecked.gif") no-repeat 0px 0px;

    padding-left: 16px;

    padding-bottom: 2px;

    font-weight: bold;

    color: #EA5200;
}
label.checked {
    background:url("../demo/images/checked.gif") no-repeat 0px 0px;
}
```

6每个字段验证通过执行函数

success : String,Callback

要验证的元素通过验证后的动作，如果跟一个字符串，会当做一个css类，也可跟一个函数

```
success: function(label) {
    // set &nbsp; as text for IE
    label.html("&nbsp;").addClass("checked");
    //label.addClass("valid").text("Ok!")
}
```

添加"valid" 到验证元素, 在CSS中定义的风格<style>label.valid {}</style>

success: "valid"

7验证的触发方式修改

下面的虽然是boolean型的，但建议除非要改为false,否则别乱添加。

onsubmit : Boolean Default: true

提交时验证. 设置唯false就用其他方法去验证

onfocusout : Boolean Default: true

失去焦点是验证(不包括checkboxes/radio buttons)

onkeyup : Boolean Default: true

在keyup时验证.

onclick : Boolean Default: true

在checkboxes 和 radio 点击时验证

focusInvalid : Boolean Default: true

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)会获得焦点

focusCleanup : Boolean Default: false

如果是true那么当未通过验证的元素获得焦点时，移除错误提示。避免和 focusInvalid 一起用

// 重置表单

```
$(document).ready(function() {
    var validator = $("#signupForm").validate({
        submitHandler:function(form){
            alert("submitted");
            form.submit();
        }
    });
});
```

```

    $("#reset").click(function() {
        validator.resetForm();
    });
});

```

8异步验证

remote : URL

使用ajax方式进行验证，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

remote: "check-email.php"

```

remote: {
    url: "check-email.php",    //后台处理程序
    type: "post",             //数据发送方式
    dataType: "json",         //接受数据格式
    data: {                   //要传递的数据
        username: function() {
            return $("#username").val();
        }
    }
}

```

远程地址只能输出 "true" 或 "false"，不能有其它输出

9添加自定义校验

addMethod : name, method, message

自定义验证方法

```

// 中文字两个字节
jQuery.validator.addMethod("byteRangeLength", function(value, element, param) {
    var length = value.length;
    for(var i = 0; i < value.length; i++){
        if(value.charCodeAt(i) > 127){
            length++;
        }
    }
    return this.optional(element) || ( length >= param[0] && length <= param[1] );
}, $.validator.format("请确保输入的值在{0}-{1}个字节之间(一个中文字算2个字节)"));

```

```

// 邮政编码验证
jQuery.validator.addMethod("isZipCode", function(value, element) {
    var tel = /^[0-9]{6}$/;
    return this.optional(element) || (tel.test(value));
}, "请正确填写您的邮政编码");

```

1.要在additional-methods.js文件中添加或者在jquery.validate.js添加

建议一般写在additional-methods.js文件中

2.在messages_cn.js文件添加：isZipCode: "只能包括中文字、英文字母、数字和下划线",

调用前要添加对additional-methods.js文件的引用。

10radio和checkbox、select的验证

```
1.radio的required表示必须选中一个
<input type="radio" id="gender_male" value="m" name="gender" class="{required:true}" />
<input type="radio" id="gender_female" value="f" name="gender"/>
```

```
2.checkbox的required表示必须选中
<input type="checkbox" class="checkbox" id="agree" name="agree" class="{required:true}" />

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
```

```
<input type="checkbox" class="checkbox" id="spam_email" value="email" name="spam[]"
class="{required:true, minlength:2}" />
<input type="checkbox" class="checkbox" id="spam_phone" value="phone" name="spam[]" />
<input type="checkbox" class="checkbox" id="spam_mail" value="mail" name="spam[]" />
```

```
3.select的required表示选中的value不能为空
<select id="jungle" name="jungle" title="Please select something!" class="{required:true}">
  <option value=""></option>
  <option value="1">Buga</option>
  <option value="2">Baga</option>
  <option value="3">Oi</option>
</select>
```

```
select的minlength表示选中的最小个数（可多选的select）,maxlength表示最大的选中个数,rangelength:
[2,3]表示选中个数区间
<select id="fruit" name="fruit" title="Please select at least two fruits" class="{required:true,
minlength:2}" multiple="multiple">
  <option value="b">Banana</option>
  <option value="a">Apple</option>
  <option value="p">Peach</option>
  <option value="t">Turtle</option>
</select>
```

jQuery.validate 中文API

名称	返回类型	描述
validate(options)	返回:Validator	验证所选的FORM
valid()	返回:Boolean	检查是否验证通过
rules()	返回:Options	返回元素的验证规则
rules("add",rules)	返回:Options	增加验证规则
rules("remove",rules)	返回:Options	删除验证规则
removeAttrs(attributes)	返回:Options	删除特殊属性并且返回他们
Custom selectors		
:blank	返回:Validator	没有值的筛选器

:filled	返回:Array <Element >	有值的筛选器
:unchecked	返回:Array <Element >	没选择的元素的筛选器
Utilities		
jQuery.format (template,argument ,argumentN...)	返回:String	用参数代替模板中的 {n}

Validator:		
<p>validate方法返回一个Validator对象,它有很多方法, 让你能使用引发校验程序或者改变form的内容. validator对象有很多方法,但下面只是列出常用的</p>		
form()	返回:Boolean	验证form返回成功还是失败
element(element)	返回:Boolean	验证单个元素是成功还是失败
resetForm()	返回:undefined	把前面验证的FORM恢复到验证前原来的状态
showErrors(errors)	返回:undefined	显示特定的错误信息
Validator functions:		
setDefault(defaults)	返回:undefined	改变默认的设置
addMethod(name,method,message)	返回:undefined	添加一个新的验证方法. 必须包括一个独一无二的名字, 一个JAVASCRIPT的方法和一个默认的信息
addClassRules(name,rules)	返回:undefined	增加组合验证类型 在一个类里面用多种验证方法里比较有用
addClassRules(rules)	返回:undefined	增加组合验证类型 在一个类里面用多种验证方法里比较有用, 这个是一下子加多个

内置验证方式:		
required()	返回:Boolean	必填验证元素
required(dependency-expression)	返回:Boolean	必填元素依赖于表达式的结果
required(dependency-callback)	返回:Boolean	必填元素依赖于回调函数的结果
remote(url)	返回:Boolean	请求远程校验。url通常是一个远程调用方法
minlength(length)	返回:Boolean	设置最小长度
maxlength(length)	返回:Boolean	设置最大长度

rangeLength (range)	返回: Boolean	设置一个长度范围 [min, max]
min (value)	返回: Boolean	设置最大值
max (value)	返回: Boolean	设置最小值
email ()	返回: Boolean	验证电子邮箱格式
range (range)	返回: Boolean	设置值的范围
url ()	返回: Boolean	验证URL格式
date ()	返回: Boolean	验证日期格式 (类似30/30/2008的格式, 不验证日期准确性只验证格式)
dateISO ()	返回: Boolean	验证ISO类型的日期格式
dateDE ()	返回: Boolean	验证德式的日期格式 (29. 04. 1994 or 1. 1. 2006)
number ()	返回: Boolean	验证十进制数字 (包括小数的)
digits ()	返回: Boolean	验证整数
creditcard ()	返回: Boolean	验证信用卡号
accept (extension)	返回: Boolean	验证相同后缀名的字符串
equalTo (other)	返回: Boolean	验证两个输入框的内容是否相同
phoneUS ()	返回: Boolean	验证美式的电话号码

validate () 的可选项:	
debug:进行调试模式 (表单不提交):	<pre> \$ ("selector").validate ({ debug: true }) </pre>
把调试设置为默认:	<pre> \$.validator.defaults({ debug: true }) </pre>
submitHandler: 通过验证后运行的函数, 里面要加上表单提交的函数, 否则表单不会提交	<pre> \$ ("selector").validate({ submitHandler: function (form) { \$ (form).ajaxSubmit (); } } </pre>

	<pre> } })</pre>
<p>ignore:</p> <p>对某些元素不进行验证</p>	<pre>\$("#myform").validate({ ignore:".ignore" })</pre>
<p>rules:</p> <p>自定义规则, key:value的形式, key是要验证的元素, value可以是字符串或对象</p>	<pre>\$(".selector").validate({ rules: { name:"required", email: { required:true, email:true } } })</pre>
<p>messages:</p> <p>自定义的提示信息key:value的形式key是要验证的元素, 值是字符串或函数</p>	<pre>\$(".selector").validate({ rules: { name:"required", email: { required:true, email:true } }, messages: { name:"Name不能为空", email: { required:"E-mail不能为空", email:"E-mail地址不正确" } } })</pre>

	<pre>)}</pre>
<p>groups:</p> <p>对一组元素的验证, 用一个错误提示, 用error Placement控制把出错信息放在哪里</p>	<pre>\$("#myform").validate({ groups:{ username:"fname lname" }, errorPlacement:function(error, element) { if (element.attr("name") == "fname" element.attr("name") == "lname") error.insertAfter("#lastname"); else error.insertAfter(element); }, debug:true })</pre>
<p>Onubmit Boolean 默认:true</p> <p>是否提交时验证</p>	<pre>\$(".selector").validate({ onsubmit:false })</pre>
<p>onfocusout Boolean 默认:true</p> <p>是否在获取焦点时验证</p>	<pre>\$(".selector").validate({ onfocusout:false })</pre>
<p>onkeyup Boolean 默认:true</p> <p>是否在敲击键盘时验证</p>	<pre>\$(".selector").validate({ onkeyup:false })</pre>
<p>onclick Boolean 默认:true</p> <p>是否在鼠标点击时验证 (一般验证 checkbox, radiobox)</p>	<pre>\$(".selector").validate({ onclick:false })</pre>

<p>focusInvalid Boolean 默认:true</p> <p>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)会获得焦点</p>	<pre> \$("<code>.selector</code>").validate({ focusInvalid:false }) </pre>
<p>focusCleanup Boolean 默认:false</p> <p>当未通过验证的元素获得焦点时, 并移除错误提示(避免和 focusInvalid. 一起使用)</p>	<pre> \$("<code>.selector</code>").validate({ focusCleanup:true }) </pre>
<p>errorClass String 默认:"error"</p> <p>指定错误提示的css类名, 可以自定义错误提示的样式</p>	<pre> \$("<code>.selector</code>").validate({ errorClass:"invalid" }) </pre>
<p>errorElement String 默认:"label"</p> <p>使用什么标签标记错误</p>	<pre> \$("<code>.selector</code>").validate errorElement:"em" }) </pre>
<p>wrapper String</p> <p>使用什么标签再把上边的errorElement包起来</p>	<pre> \$("<code>.selector</code>").validate({ wrapper:"li" }) </pre>
<p>errorLabelContainer Selector</p> <p>把错误信息统一放在一个容器里面</p>	<pre> \$("<code>#myform</code>").validate({ errorLabelContainer:"<code>#messageBox</code>", wrapper:"li", submitHandler:function() { alert("Submitted!") } }) </pre>
<p>showErrors:</p> <p>跟一个函数, 可以显示总共有多少个未通过验证的元素</p>	<pre> \$("<code>.selector</code>").validate({ showErrors:function(errorMap,errorList) { \$("<code>#summary</code>").html("Your form contains " + this.numberOfInvalids() + " errors,see details below."); this.defaultShowErrors(); } }) </pre>

	<pre> }) </pre>
<p>errorPlacement:</p> <p>跟一个函数, 可以自定义错误放到哪里</p>	<pre> \$("#myform").validate({ errorPlacement:function(error,element) { error.appendTo(element.parent("td").next("td")); }, debug:true }) </pre>
<p>success:</p> <p>要验证的元素通过验证后的动作, 如果跟一个字符串, 会当做一个css类, 也可跟一个函数</p>	<pre> \$("#myform").validate({ success:"valid", submitHandler:function() { alert("Submitted!") } }) </pre>
<p>highlight:</p> <p>可以给未通过验证的元素加效果, 闪烁等</p>	

addMethod(name, method, message) 方法:
参数name是添加的方法的名字
<p>参数method是一个函数, 接收三个参数(value, element, param)</p> <p>value是元素的值, element是元素本身</p> <p>param是参数, 我们可以用addMethod来添加除built-in Validation methods之外的验证方法 比如有一个字段, 只能输一个字母, 范围是a-f, 写法如下:</p> <pre> \$.validator.addMethod("af", function(value, element, params) { if(value.length>1) { return false; } if(value>=params[0] && value<=params[1]) { return true; }else{ return false; </pre>

```
    }
```

}, "必须是一个字母, 且a-f");

用的时候, 比如有个表单字段的id="username", 则在rules中写

```
username: {  
  
    af: ["a", "f"]  
  
}
```

addMethod的第一个参数, 就是添加的验证方法的名子, 这时是af

addMethod的第三个参数, 就是自定义的错误提示, 这里的提示为: "必须是一个字母, 且a-f"

addMethod的第二个参数, 是一个函数, 这个比较重要, 决定了用这个验证方法时的写法

如果只有一个参数, 直接写, 如果af: "a", 那么a就是这个唯一的参数, 如果多个参数, 用在[]里, 用逗号分开

meta String方式:

```
$("#myform").validate({  
  
    meta: "validate",  
  
    submitHandler: function() {  
alert("Submitted!") }  
  
})
```

```
<script type="text/javascript"  
src="js/jquery.metadata.js"></script>  
  
<script type="text/javascript"  
src="js/jquery.validate.js"></script>  
  
<form id="myform">  
  
    <input type="text"  
name="email" class="{validate:{ required:true, email:true }}" />  
  
    <input type="submit"  
value="Submit" />  
  
</form>
```

分类: [jQuery/Javascript](#)

好文要顶

关注我

收藏该文



Rex.He

关注 - 4

粉丝 - 59

+加关注

发表评论

#1楼 2012-05-22 15:23 | nuki

非常好

支持(0) 反对(0)

#2楼 2012-10-24 09:45 | 要有好的心情

mark

支持(0) 反对(0)

#3楼 2013-01-17 09:36 | Johnny Ding

MARK 顶!

支持(0) 反对(0)

#4楼 2013-02-17 10:31 | 虔城墨客

很全，谢谢了。

支持(0) 反对(0)

#5楼 2013-02-18 10:59 | 站在我的肩上

正在学习使用中，谢谢

支持(0) 反对(0)

#6楼 2013-09-15 13:20 | 张张开

非常好，正在学习。

支持(0) 反对(0)

#7楼 2013-10-08 18:56 | tinyphp

顶~

支持(0) 反对(0)

#8楼 2013-10-15 09:09 | fang8580

mark

支持(0) 反对(0)

#9楼 2013-11-07 14:44 | 银杏叶儿

不知道为什么，一直没有提示错误信息，

支持(0) 反对(0)

#10楼 2013-11-28 18:57 | gyy_Jade

把<input>标签的name属性值命名为连字符的形式，即如register-pwd，控件就会失效。~attention

支持(0) 反对(0)

#11楼 2013-12-05 17:34 | 清新每一天

mark

支持(0) 反对(

#12楼 2014-01-03 11:28 | 虞筱瑶

英文提示改为中文提示那块 楼主建议把中文提示另存为一个js文件。想问问 不可以在原有的JQury.validate库中更改吗？

支持(0) 反对(

#13楼 2014-01-11 13:19 | 懒鬼5317

很详细，学习了

支持(0) 反对(

#14楼[楼主] 2014-01-11 13:21 | Rex.He

@ 员？猿！
当然可以，只是建议，会比较后支持多语言

支持(0) 反对(

#15楼 2014-02-07 16:28 | wxl1990721

你好：我的部分代码如下 邮箱可以正常验证 但是密码没有验证 为什么呢？

```
<script type="text/javascript" src="js/front/jquery.js" charset="gbk"></script>
<script type="text/javascript" src="js/front/jquery.validate.js" charset="gbk"></script>
<script type="text/javascript" src="js/front/jquery.metadata.js" charset="gbk"></script>
<script type="text/javascript" src="js/front/message_cn.js" charset="gbk"></script></head>

<script type="text/javascript">
$.ready(function() {
$("#signupForm").validate();
});
</script>

<!-- 访问链接 http://192.168.9.33:8080/1005\_ibook/pages/front/register.jsp -->
<body>
<form id="signupForm" action="front/register_test!save.action" method="get">
邮箱：<input type="text" name="email" id="email" class="required email"/><br>
密码：<input id="password" name="password" type="password" class="{required:true, minlength:5}" />
r>
确认密码：<input id="rePassword" name="rePassword" type="password" class="{required:true, minlength:5, equalTo:'#password'}" /><br>
<input class="submit" type="submit" value="注册"/>
</form>
</body>
```

支持(1) 反对(

#16楼 2014-02-22 16:22 | 云霄n

@ 银杏叶儿
jQuery和jQuery validate有个版本对应的关系。我的就是不对应搞了好久。

支持(0) 反对(

#17楼 2014-03-30 14:52 | 晨雨零稀

addClassRules这个方法无法自定义错误信息啊，楼主可知道如何搞？

支持(0) 反对(

#18楼 2014-04-10 17:10 | 立即出发

学习 学习

支持(0) 反对(

#19楼 2014-04-16 11:31 | 特雷西one

学习了，mark

支持(0) 反对(0)

#20楼 2014-04-25 15:51 | 小方。

不错，细致

支持(0) 反对(0)

#21楼 2014-05-30 17:53 | 开源控Meteor(OSMeteor)

新版不支持jquery.metadata.js 1.8 的才支持

支持(0) 反对(0)

#22楼 2014-05-30 17:55 | 开源控Meteor(OSMeteor)

被版本纠结死。。。

支持(0) 反对(0)

#23楼 2014-06-11 08:54 | 例维亚的杰洛特

mark

支持(0) 反对(0)

#24楼 2014-07-23 17:58 | 素和墨如

请问，这个插件能用于非from容器么？

支持(0) 反对(0)

#25楼 2014-10-29 09:23 | lovexyw

好文。

支持(0) 反对(0)

#26楼 2014-11-27 14:19 | 维京师

非常不错。谢谢博主

支持(0) 反对(0)

#27楼 2014-12-05 10:44 | 一对一软件

mark

支持(0) 反对(0)

#28楼 2014-12-26 09:19 | bugong

mark

支持(0) 反对(0)

#29楼 2015-01-12 21:16 | 船头尺

楼主，我想让错误信息在一个div里面显示，并且是从上到下只显示第一条错误

信息进行较验，请问这个该怎么做？

比如，现在有两个字段，第一个是用户名，第二个是手机号，如果用户名和手

机号都不对，则显示用户名不对。进待楼主回信。谢谢

支持(0) 反对(0)

#30楼 2015-01-30 15:19 | 王赫赫

给力 感谢！

支持(0) 反对(0)

#31楼 2015-02-12 16:30 | super-summer

校验银行卡的正则好像不对 有的就校验不过去

支持(0) 反对(0)

#32楼 2015-04-07 14:25 | 崔俊宁

关于对bootstrap-datetimepicker的验证,如何恰当使用：

```
<label for="telephone" class="col-sm-3 control-label">时间</label>
<div class="input-group date form_datetime col-sm-3 pl15" data-date-format="yyyy-mm-dd hh:ii" data-lit
-field="dtp_input1" id="stime">
<input class="form-control " style="width:60% !important" size="16" type="text" value='<fmt:formatDat
value="$ {entity.starttime }" pattern="yyyy-MM-dd HH:mm"/>' name="entity.starttimevalue" id="starttim
readonly >
<span class="input-group-addon dateAddValidateRemove"><span class="glyphicon glyphicon-remove"><
pan></span>
<span class="input-group-addon dateAddValidateCalendar"><span class="glyphicon glyphicon-calendar">
</span></span>
</div>
```

支持(0) 反对(0)

#33楼 2015-04-28 14:24 | KevinGarnett

您好，firstname: "required",这个firstname指的是id吗？在线等。

支持(0) 反对(0)

#34楼 2015-10-15 16:11 | 狷特头

看这个文章就好了，官网看不懂。

支持(0) 反对(0)

#35楼 2015-11-14 20:43 | 刘彦佐

@ rainFish05
用""引号包括起来，就可以运行正常

支持(0) 反对(0)

#36楼 2016-07-20 15:58 | xiaonuan筱暖

很详细，挺好的

支持(0) 反对(0)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】超50万VC++源码：大型工控、组态\仿真、建模CAD源码2018！

【推荐】微信小程序一站式部署 多场景模板定制



开发 **在线Excel** 不再难
用 **SpreadJS** 一分钟搞定

[了解详情](#)

最新IT新闻:

- 为庆祝中国春节 谷歌Waze导航添加中文语音
 - 美团外卖除夕大数据：均价2000元以上年夜饭受青睐
 - 比特币是又一朵郁金香？已成威胁金融安全的毒瘤
 - 春节留守程序员：最期望负责的业务一切正常
 - 外媒指苹果音箱平庸 一次对亚马逊不成功的模仿？
- » [更多新闻...](#)



告别高昂运维费用 云计算全面助力

40+款核心产品免费半年 再+8000津贴任意采购

[立即申请](#)

最新知识库文章:

- 领域驱动设计在互联网业务开发中的实践
 - 步入云计算
 - 以操作系统的角度述说线程与进程
 - 软件测试转型之路
 - 门内门外看招聘
- » [更多知识库文章...](#)