[bootstrapValidator使用总结](https://www.cnblogs.com/huiy/p/9583633.html)

1.根据条件动态增删检查条件

参考文章：<https://blog.csdn.net/sxlzs_/article/details/78211928>

#去除某字段的验证

$("#formResource").bootstrapValidator('removeField', 'permission');

#增加某字段的验证

[复制代码](javascript:void(0);)

$("#formResource").bootstrapValidator("addField", "permission", {

validators : {

notEmpty : {

message : '数据标识不能为空'

}

}

});

[复制代码](javascript:void(0);)

2.select等值会发生变化时处理

[复制代码](javascript:void(0);)

sjklx : {

trigger : 'change',

validators : {

notEmpty : {

message : '数据库类型不能为空'

}

}

}

[复制代码](javascript:void(0);)

3.输入字符长度控制，添加正则校验

以输入密码为例：

[复制代码](javascript:void(0);)

pass: {

validators: {

stringLength: {

min: 6,

max: 18,

message: '密码长度至少6位'

},

regexp: {

regexp: /^(?![0-9]+$)(?![a-zA-Z]+$)[0-9A-Za-z]{6,18}$/,

message: '必须含有大写字母，小写字母和数字'

}

}

}

[复制代码](javascript:void(0);)

4.两次密码校验

[复制代码](javascript:void(0);)

pass: {

validators: {

stringLength: {

min: 6,

max: 18,

message: '密码长度至少6位'

},

regexp: {

regexp: /^(?![0-9]+$)(?![a-zA-Z]+$)[0-9A-Za-z]{6,18}$/,

message: '必须含有大写字母，小写字母和数字'

}

}

},

// 密码确认

passConfirm: {

message: '密码确认验证失败',

validators: {

identical: {

field: 'pass',

message: '两次密码不相同'

}

}

}

[复制代码](javascript:void(0);)

4.ip 校验

[复制代码](javascript:void(0);)

ip: {

validators: {

ip: {

message: '请输入正确的IP地址'

}

}

}

[复制代码](javascript:void(0);)

5. emailAddress校验

[复制代码](javascript:void(0);)

mail: {

validators: {

emailAddress: {

message: '请输入正确的Email地址'

}

}

}

[复制代码](javascript:void(0);)

6. between：值范围校验

[复制代码](javascript:void(0);)

level: {

validators: {

between: {

message: '输入值必须在1到10之间'

}

}

}

[复制代码](javascript:void(0);)

7.base64校验

[复制代码](javascript:void(0);)

base64Test: {

validators: {

base64: {

message: '不是合法的base64编码'

}

}

},

[复制代码](javascript:void(0);)

7.身份证校验

[复制代码](javascript:void(0);)

creditCardTest: {

validators: {

creditCard: {

message: '不是合法的身份证'

}

}

},

[复制代码](javascript:void(0);)

8.date日期校验

[复制代码](javascript:void(0);)

dateTest: {

validators: {

date: {

message: '不是合法的日期'

}

}

}

[复制代码](javascript:void(0);)

9.numeric校验

[复制代码](javascript:void(0);)

numericTest: {

validators: {

numeric: {

message: '不是合法的数字'

}

}

}

[复制代码](javascript:void(0);)

9.uri校验

[复制代码](javascript:void(0);)

dateTest: {

validators: {

uri: {

message: '不是合法的URI'

}

}

}

[复制代码](javascript:void(0);)

10.电话号码校验

[复制代码](javascript:void(0);)

phoneTest: {

validators: {

phone: {

message: '不是合法的电话'

}

}

}

[复制代码](javascript:void(0);)

11.校验范围：**隐藏域（:hidden）、禁用域（:disabled）、那啥域（:not(visible)）**

**bootstrap默认不对上述三种类型进行校验，如果需要改变，见下：**

[复制代码](javascript:void(0);)

$('#form').bootstrapValidator({

message: '您的输入有误',

excluded: [":disabled"],

fields: {

zjxh: {

validators: {

notEmpty: {

message: '闸机型号不能为空'

}

}

}

}

});

[复制代码](javascript:void(0);)

12.input=file校验

[复制代码](javascript:void(0);)

zp : {

trigger : 'change',

validators : {

notEmpty : {

message : '照片不能为空'

},

regexp : {

regexp : /jpg$/,

message : '格式必须为.jpg'

}

}

}

[复制代码](javascript:void(0);)

13.以上规则如果都不满足，可以采用回调函数实现规则

[复制代码](javascript:void(0);)

card: {

validators: {

notEmpty: {

message: '身份证证明号码不能为空'

},

callback: {

message: '身份证号码不符合要求',

callback: function(value, validator) {

return IdentityCard(value);

}

}

}

}

[复制代码](javascript:void(0);)

IdentityCard 为自定义函数，只要返回boolean值就可以。