

## 61. 独立验证和内置规则

学习要点：

1. 独立验证
2. 内置规则

本节课我们来学习一下数据验证的知识，这节课了解独立验证和内置规则。

### 一. 独立验证

1. 之前，已经了解了一些独立验证的地方，系统还提供了 **make** 方法实现独立验证；
2. 只不过这个方法，并且在 TP6 就废弃了，所以，简单了解一下即可；

```
$data = [
    'name'      =>  '',
    'email'     =>  'bnbbs163.com'
];

$validate = Validate::make([
    'name'      =>  'require|max:25',
    'email'     =>  'email'
]);

if (!$validate->batch()->check($data)) {
    dump($validate->getError());
}
```

### 二. 内置规则

1. 内置的规则内容较多，并且严格区分大小写，这里按照类别一一列出；
2. 静态方法支持两种形式，比如 **::number()** 或者 **isNumber()** 均可；
3. 而 **require** 是 PHP 保留字，那么就必须用 **isRequire()** 或 **must()**；
4. 格式验证类：

|         |    |              |                           |
|---------|----|--------------|---------------------------|
| 'field' | => | 'require',   | //不得为空::isRequire 或::must |
| 'field' | => | 'number',    | //是否是纯数字，非负数非小数点          |
| 'field' | => | 'integer',   | //是否是整数                   |
| 'field' | => | 'float',     | //是否是浮点数                  |
| 'field' | => | 'boolean',   | //是否是布尔值，或者 bool          |
| 'field' | => | 'email',     | //是否是 email               |
| 'field' | => | 'array',     | //是否是数组                   |
| 'field' | => | 'accepted',  | //是否是“yes”“no”“1”这三个值     |
| 'field' | => | 'date',      | //是否是有效日期                 |
| 'field' | => | 'alpha',     | //是否是纯字母                  |
| 'field' | => | 'alphaNum',  | //是否是字母和数字                |
| 'field' | => | 'alphaDash', | //是否是字母和数字以及_-(下划线和破折号)   |
| 'field' | => | 'chs',       | //是否是纯汉字                  |

```

'field'    =>    'chsAlpha',        //是否是汉字字母
'field'    =>    'chsAlphaNum',      //是否是汉字字母数字
'field'    =>    'chsDash',          //是否是汉字字母数字以及_(下划线和破折号)
'field'    =>    'cntrl',            //是否是控制字符(换行、缩进、空格)
'field'    =>    'graph',            //是否是可打印字符(空格除外)
'field'    =>    'print',            //是否是可打印字符(包含空格)
'field'    =>    'lower',            //是否是小写字符
'field'    =>    'upper',            //是否是大写字符
'field'    =>    'space',            //是否是空白字符
'field'    =>    'xdigit',           //是否是十六进制
'field'    =>    'activeUrl',        //是否是有效域名或 IP 地址
'field'    =>    'url',              //是否是有效 URL 地址
'field'    =>    'ip',               //是否是有效 IP(支持 ipv4,ipv6)
'field'    =>    'dateFormat:Y-m-d', //是否是指定日期格式
'field'    =>    'mobile',           //是否是有效手机
'field'    =>    'idCard',           //是否是有效身份证
'field'    =>    'macAddr',          //是否是有效 MAC 地址
'field'    =>    'zip',              //是否是有效邮编

```

#### 5. 长度和区间验证类:

```

'field'    =>    'in:1,2,3',         //是否有某个值
'field'    =>    'notIn:1,2,3',      //是否是没有某个值
'field'    =>    'between:1,100',    //是否是在区间中
'field'    =>    'notBetween:1,100', //是否是不在区间中
'field'    =>    'length:2,20',      //是否字符长度在范围中
'field'    =>    'length:4',         //是否字符长度匹配
'field'    =>    'max:20',           //是否字符最大长度
'field'    =>    'min:5',            //是否字符最小长度
//length、max、min 也可以判断数组长度和 File 文件大小

```

```

'field'    =>    'after:2020-1-1',   //是否在指定日期之后
'field'    =>    'before:2020-1-1', //是否在指定日期之前

```

//是否在当前操作是否在某个有效期内

```

'field'    =>    'expire:2019-1-1,2020-1-1',
//验证当前请求的 IP 是否在某个范围之间,
'field'    =>    'allowIp:221.221.78.1, 192.168.0.1',
//验证当前请求的 IP 是否被禁用
'field'    =>    'denyIp:221.221.78.1, 127.0.0.1',

```

#### 6. 字段比较类:

```

'field'    =>    'confirm:password', //是否和另一个字段匹配
'field'    =>    'differnet:password', //是否和另一个字段不匹配
'field'    =>    'eq:100',            //是否等于某个值, =、same 均可
'field'    =>    'gt:100',            //是否大于某个值, 支持>

```

```
'field'    =>    'egt:100',        //是否大于等于某个值，支持>=  
'field'    =>    'lt:100',        //是否小于某个值，支持<  
'field'    =>    'elt:100',       //是否小于等于某个值，支持<=  
//比较方式也支持字段比较，比如：'field'=>'lt:price'
```

#### 7. 其它验证类：

```
'field'    =>    '\d{6}',          //正则表达式验证  
'field'    =>    'regex:\d{6}',    //正则表达式验证  
'field'    =>    'file',           //判断是否是上传文件  
'field'    =>    'image:150,150,gif', //判断图片(参数可选)  
'field'    =>    'fileExt:jpg,txt',  //判断文件允许的后缀  
'field'    =>    'fileMime:text/html', //判断文件允许的文件类型  
'field'    =>    'fileSize:2048',    //判断文件允许的字节大小  
  
'field'    =>    'behavior:\app\...', //判断行为验证  
'field'    =>    'unique:user',      //验证 field 字段的值是否在 user 表  
'field'    =>    'requireWith:account' //当 account 有值时,requireWidth 必须
```