

44. 验证类的使用

学习要点：

1. 创建验证类

本节课我们来开始学习验证类的创建和使用方法。

一. 创建验证类

1. 上节课，使用的快速验证，适合小规模单独的验证；
2. 也创建一个验证类，把验证规则写到验证类里，然后调用；

```
php artisan make:request Form
```

3. 验证类 `Http\Request\Form.php` 创建好之后，会提供两个固定方法；

```
public function authorize()  
{  
    //默认 false，为关闭授权，关闭状态会 403  
    //比如判断这个用户是否有操作权限  
    return true;  
}  
  
public function rules()  
{  
    return [  
        //规则  
        'username' => 'required|min:2|max:10',  
        'password' => 'required|min:6',  
    ];  
}
```

4. 在控制器端，直接使用验证类进行验证即可，其它和快速验证一样；

```
//通过验证类验证  
$request->validated();
```

5. 默认错误提示是英文的，如果想设置成中文提示，可在验证类创建 `message()`；

```
//验证器类方法  
public function messages()  
{  
    return [  
        'username.required' => '用户名不得为空~',  
        'username.min' => '用户名不得小于 2 位~',  
        'username.max' => '用户名不得大于 10 位~',  
  
        'password.required' => '密码不得为空~',  
    ];  
}
```

```
        'password.min' => '密码不得小于 6 位~',  
    ];  
}
```

6. 也可以自定义属性名，而其它采用默认提示；

```
public function attributes()  
{  
    return [  
        'username' => '用户名',  
    ];  
}
```

PS: 提示语言包在: resources\lang\en\validation.php

7. 在请求验证之前，修改提交的请求数据；

```
protected function prepareForValidation()  
{  
    $this->merge([  
        'username' => 'Mr.Lee'  
    ]);  
}
```