

43. 表单快速验证

学习要点：

1. 快速验证

本节课我们来开始学习表单提交之后的快速验证功能。

一. 快速验证

1. 首先，创建两个路由：一个表单，一个处理表单；

```
//表单页
Route::get('/task/form','TaskController@form');
//接收页
Route::post('/task/receive','TaskController@receive');
```

2. 表单，我们先采用用户名和密码来演示验证；

```
<form action="/task/receive" method="post">
    @csrf
    用户名 : <input type="text" name="username">
    密码 : <input type="password" name="password">
    <button type="submit">提交</button>
</form>
```

3. 表单处理页，通过 Request 对象来实现字段验证，注释写了执行流程；

```
public function receive(Request $request)
{
    //验证请求方式
    //左边为验证字段，右边为验证规则，每个规则竖线隔开
    //比如 required 不得为空，min 不得小于，max 不得大于
    $request->validate([
        'username' => 'required|min:2|max:10',
        'password' => 'required|min:6',
    ]);

    //只有通过才能执行下面的语句，否则返回表单页
    return '恭喜验证通过！';
}
```

4. 如果验证没有被通过，会自动重定向到提交页，并将错误信息存储到 session 中；

```
@if ($errors->any())
    <div>
        <ul>
            @foreach ($errors->all() as $error)
                <li>{{ $error }}</li>
            @endforeach
        </ul>
    </div>
```

```
        @endforeach
    </ul>
</div>
@endif
```

PS: 内部操作是通过中间件: Illuminate\View\Middleware\ShareErrorsFromSession;

5. 也可以单独设置错误信息, 使用@error 指令;

```
@error('username')
    <div>{{ $message }} : 用户名非法! </div>
@enderror

@error('password')
    <div>{{ $message }} : 密码非法! </div>
@enderror
```

6. 关于规则, 还有其它写法, 具体如下:

```
//数组方式验证
$request->validate([
    'username'    => ['required', 'min:2', 'max:10'],
    'password'    => ['required', 'min:6']
]);
```