

10. 验证提示和数据写入

学习要点：

1. 验证提示
2. 数据写入

本节课我们来开始进入学习一下验证提示模版和用户新增入库的方法。

一. 验证提示

1. 因为不做前端模态框的提示方案，后端提示我们采用模版提示；
2. 适当结合 Bootstrap 本身的提示控件，让显示效果好一些；

```
<!--简易提示-->
<div class="d-flex vw-100 vh-100 bg-light justify-content-center align-items-center">
    <div class="toast show mb-5 pl-3 pr-3">
        <div class="toast-header">
            <strong>● 系统提示</strong>
        </div>
        <div class="toast-body">
            <ol class="text-danger">
                {volist name="infos" id="item"}
                <li>{$item}</li>
            {/volist}
            </ol>
            <a href="{$_url_path}" class="btn btn-primary btn-sm w-100">{$url_text}</a>
        </div>
    </div>
</div>
```

3. 过滤方面，粗暴的先搞个字段前后空格去除，在 app/request.php;

```
// 应用请求对象类
class Request extends \think\Request
{
    protected $filter = ['trim'];
}
```

4. 获取数据，进行验证提示；

```
//获取请求数据
$data = $request->param();

//将提交过来的数据交给验证类验证
//支持批量验证和提醒
```

```
try {
    validate(UserValidate::class)
        ->batch(true)
        ->check($data);
} catch (ValidateException $exception) {
    return view('./view/public/toast.html', [
        'infos' => $exception->getError(),
        'url_text' => '返回上一页',
        'url_path' => url('/user/create'),
    ]);
}
```

二. 数据写入

1. 写入成功，也引用这个模版，设置不同的提示信息即可，如要个性化，重新建；

```
//密码加密
$data['password'] = sha1($data['password']);
//新增
$id = UserModel::create($data)->getData('id');
//提示
return $id ? view('./view/public/toast.html', [
    'infos' => ['恭喜，注册成功！'],
    'url_text' => '去首页',
    'url_path' => url('/')
]) : '注册失败！';
```