

58. 单个验证和注解验证

学习要点：

1. 单个验证
2. 注解验证

本节课我们来学习一下数据验证的知识，这节课了解单个验证和注解验证。

一. 单个验证

1. 静态调用，即使用 **facade** 模式进行调用验证，非常适合单个数据的验证；
2. 引入 **facade** 中的 **Validate** 时，和其它 **Validate** 会冲突，要特别注意；

```
//验证邮箱是否合法
dump(Validate::isEmail('xiaoxin@163.com'));
//验证是否为空
dump(Validate::isRequire(''));
//验证是否为数值
dump(Validate::isNumber(10));
```

3. 静态调用返回的结果是 **false** 和 **true**，方便你进行条件判断；
4. 静态调用，也是支持多规则验证的，使用 **checkRule()** 方法实现；

```
//验证数值合法性
dump(Validate::checkRule(10, 'number|between:1,10'));
```

5. **checkRule()** 不支持错误信息，需要自己实现，但支持对象化规则定义；
- ```
dump(Validate::checkRule(10, ValidateRule::isNumber()->between('1,10')));
```

### 二. 注解验证

1. 可以结合之前课程中注解路由的传参，使用验证方式，对其进行验证；

```
use think\annotation\Route;
use think\annotation\route\Validate;

/**
 * @param $id
 * @return string
 * @route("vr/:id")
 * @Validate(User::class)
 */
```