

## spring mvc 校验@NULL @notNULL等

2018年03月27日 09:45:41 durenniu 阅读数 9742

### 一、准备校验时使用的JAR

[java]

1. validation-api-1.0.0.GA.jar: JDK的接口;
2. hibernate-validator-4.2.0.Final.jar是对上述接口的实现;
3. log4j、slf4j、slf4j-log4j

### 二、编写需要校验的bean

```
@NotNull(message="名字不能为空")
private String userName;
@Max(value=120,message="年龄最大不能超过120")
private int age;
@Email(message="邮箱格式错误")
private String email;
```

### 三、校验方法

```
@RequestMapping("/login")
public String testValid(@Valid User user, BindingResult result){
    if (result.hasErrors()){
        List<ObjectError> errorList = result.getAllErrors();
        for(ObjectError error : errorList){
            System.out.println(error.getDefaultMessage());
        }
    }

    return "test";
}
```

**备注：**这里一个@Valid的参数后必须紧挨着一个BindingResult 参数，否则spring会在校验不通过时直接抛出异常

## 校验注解

### 1 空检查

[java]

1. @Null 验证对象是否为null
2. @NotNull验证对象是否不为null, 无法检查长度为0的字符串
3. @NotBlank 检查约束字符串是不是Null还有被Trim的长度是否大于0,只对字符串,且会去掉前后空格.
4. @NotEmpty 检查约束元素是否为NULL或者是EMPTY, 用在集合类上面

1. @NotNull: 不能为null, 但可以为empty(" ", " ", " ")
 2. @NotEmpty: 不能为null, 而且长度必须大于0(" ", " ")
 3. @NotBlank: 只能作用在String上, 不能为null, 而且调用trim()后, 长度必须大于0, ("test") 即: 必须有实际字符

@NotNull: The CharSequence, Collection, Map or Array object is not null, but can be empty.  
 @NotEmpty: The CharSequence, Collection, Map or Array object is not null and size > 0.

@NotBlank: The string is not null and the trimmed length is greater than zero.

#### Examples:

[java] □

1. String name = **null**;
2. @NotNull: **false**
3. @NotEmpty:**false**
4. @NotBlank:**false**
- 5.
6. String name = **" "**;
7. @NotNull:**true**
8. @NotEmpty: **false**
9. @NotBlank: **false**
- 10.
11. String name = **" ";**
12. @NotNull: **true**
13. @NotEmpty: **true**
14. @NotBlank: **false**
- 15.
16. String name = **"Great answer!"**;
17. @NotNull: **true**
18. @NotEmpty:**true**
19. @NotBlank:**true**

#### Booleans 检查

[java] □

1. @AssertTrue 验证 Boolean 对象是否为 **true**
2. @AssertFalse 验证 Boolean 对象是否为 **false**

#### 长度检查

[java] □

1. @Size(min=, max=) 验证对象 (Array, Collection, Map, String) 长度是否在给定的范围之内。可以验证集合内元素的多少。
2. @Length(min=, max=) Validates that the annotated string is between min and max included. 主要用于String类型

#### 日期检查

[java] □

1. @Past 验证 Date 和 Calendar 对象是否在当前时间之前
2. @Future 验证 Date 和 Calendar 对象是否在当前时间之后
3. @Pattern 验证 String 对象是否符合正则表达式的规则

#### 数值检查

建议使用在String, Integer类型，不建议使用在int类型上，  
因为表单值为“”时无法转换为int，但可以转换为String为“”，Integer为null

[html] □

1. @Min 验证 Number 和 String 对象是否大于等于指定的值
2. @Max 验证 Number 和 String 对象是否小于等于指定的值
3. @DecimalMax 被标注的值必须不大于约束中指定的最大值。这个约束的参数是一个通过BigDecimal定义的最大值的字符串表示。小数存在精度

4. @DecimalMin 被标注的值必须不小于约束中指定的最小值. 这个约束的参数是一个通过BigDecimal定义的最小值的字符串表示. 小数存在精度
5. @Digits 验证 Number 和 String 的构成是否合法
6. @Digits(integer=,fraction=) 验证字符串是否是符合指定格式的数字, integer指定整数精度, fraction指定小数精度。

## 其他

[java] □

1. @Valid 递归的对关联对象进行校验, 如果关联对象是个集合或者数组,那么对其中的元素进行递归校验,如果是一个map,则对其中的值部分进行校验.(是否进行递归验证)
2. @CreditCardNumber 信用卡验证
3. @Email 验证是否是邮件地址, 如果为null,不进行验证, 算通过验证。
4. @ScriptAssert(lang= ,script=, alias=)
5. @URL(protocol=,host=, port=,regexp=, flags=)
6. @Range(min=, max=) Checks whether the annotated value lies between (inclusive) the specified minimum and maximum.
7. @Range(min=10000,max=50000,message="range.bean.wage")

## 小结

枚举: @NotNull校验是否为空 值的正确性有框架判定

引用: 首先@NotNull判定是否为空, 然后@valid进行级联校验

数值: @NotNull判断是否为空, 使用@size/@Max/@Min进行大小的控制

日期: @NotNull校验是否为空 @DateTimeFormat结合joda可以固定日期的格式

对于日期的范围 注解解决不了 需要写方法判断了

日期类型输入纯文本数字也是可以通过的 值得注意

字符串: 使用@NotBlank,而不是@NotNull、@NotEmpty, @NotBlan是2者的结合; 使用@Length限制长度

对于其输入的具体内容的控制 目前没有好办法

@NotEmpty 用在集合类上面 不能为null, 而且长度必须大于0 ("","","")

@NotBlank 用在String上面 只能作用在String上, 不能为null, 而且调用trim()后, 长度必须大于0 ("test") 即: 必须有实际字符

@NotNull 用在基本类型上 不能为null, 但可以为empty ("","","","")

## 级联校验

[java] □

```

1. public class ReportVO {
2.
3.     @NotNull(message = "举报内容不能为空")
4.     @Valid
5.     private ReportContent content;
6.
7.     @NotNull(message = "举报信息不能为空")
8.     @Valid
9.     private ReportInfo info;
10. }
```

[java] □

```

1. public class ReportInfo extends BaseTenantDomain<String> {
2.
3.     private String reportContentId;
4.
5.     @NotBlank(message = "举报人id不能为空")
6.     private String reportId;
7.
8.     @NotNull(message = "举报类型不能为空")
9.     private ReportType reportType;
10. }
```

```
11. @NotBlank(message = "举报详细描述不能为空")
12. @Size(max=100, message = "举报详细描述不能超过100")
13. private String desc;
14.
15. private Date reportTime; // 举报时间
16. private Date dealTime; // 处理时间
17.
18. }
```

[java] ⚙️

```
1. public void report(@RequestBody @Valid ReportVO reportVo) {
2.     contentService.report(reportVo.getContent(), reportVo.getInfo());
3. }<span style="font-family:microsoft yahei;"><span style="font-size: 15px; line-height: 35px;">
4. </span></span>
```