# 问题解决项目SpringMybatis

## 配置文件

### [1.解决@value](mailto:1.解决@value)注解获取值问题

解决 IllegalArgumentException: Could not resolve placeholder in string value "${XXXXXX}"

如题：

导致这一问题的原因：使用了重复的property-placeholder

如一个配置文件中使用了

<context:property-placeholder location="classpath:aa.properties" />

而另一处使用了

<bean id="propertyConfigurer">

<!--class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">-->

<property name="locations">-->

<list>

<value>classpath:bb.properties</value>

</list>

</property>

</bean>

解决：

主要从以下几个地方去解决：

1. 去掉一处的propertyConfigurer配置，替换成context:property-placeholder

2. 两处都添加ignore-unresolvable="true"

配置文件1：

<context:property-placeholder location="classpath:aa.properties" ignore-unresolvable="true" />

配置文件2：

<context:property-placeholder location="classpath:bb.properties" ignore-unresolvable="true" />

### ２.Spring定时任务调度问题

在Spring配置文件中进行配置基本信息

在xmlns中添加

xmlns:task="http://www.springframework.org/schema/task"

在xsi中添加

http://www.springframework.org/schema/task

<http://www.springframework.org/schema/task/spring-task-3.2.xsd>

添加

*<!--配置定时任务-->*<**task:annotation-driven scheduler="qbScheduler" mode="proxy"**/>  
<**task:scheduler id="qbScheduler"** />

然后是必须的一项，因为这个是spring提供的定时任务，所以需要进行扫描类下的包才能执行相应的操作

<!-- 配置扫描的包 我把相应的测试类放到**com.task** 包下，这个一定要写-->

<**context:component-scan base-package="com.task"**/>

然后是设置定时任务

<!-- 设置定时任务 -->

<!--<task:annotation-driven/>-->

开始在测试类中写测试代码

@Component

public class TimerTask {

@Scheduled(cron = "\* \* \* \* \* ?")//每分钟都执行

public void test(){

System.out.println("执行");

}

}

Cron配置

CRON表达式 含义

“0 0 12 \* \* ?” 每天中午十二点触发

“0 15 10 ? \* \*” 每天早上10：15触发

“0 15 10 \* \* ?” 每天早上10：15触发

“0 15 10 \* \* ? \*” 每天早上10：15触发

“0 15 10 \* \* ? 2005” 2005年的每天早上10：15触发

“0 \* 14 \* \* ?” 每天从下午2点开始到2点59分每分钟一次触发

“0 0/5 14 \* \* ?” 每天从下午2点开始到2：55分结束每5分钟一次触发

“0 0/5 14,18 \* \* ?” 每天的下午2点至2：55和6点至6点55分两个时间段内每5分钟一次触发

“0 0-5 14 \* \* ?” 每天14:00至14:05每分钟一次触发

“0 10,44 14 ? 3 WED” 三月的每周三的14：10和14：44触发

“0 15 10 ? \* MON-FRI” 每个周一、周二、周三、周四、周五的10：15触发