# Spring基于XML配置的AOP

一般配置AOP的代码为公共代码,例如工具类

```
一、导入jar包
```

```
<?xml version="1.0" encoding="UTF-8"?>
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0"
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.itheima</groupId>
  <artifactId>day03 eesy 03springAOP</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>
  <dependencies>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>5.0.2.RELEASE</version>
    </dependency>
//解析切入点表达式
    <dependency>
      <groupId>org.aspectj</groupId>
      <artifactId>aspectiweaver</artifactId>
      <version>1.8.7</version>
    </dependency>
  </dependencies>
</project>
```

## 二、配置bean.xml

- 1、把通知Bean也交给spring来管理
- 2、使用aop:config标签表明开始AOP的配置
- 3、使用aop:aspect标签表明配置切面

id属性: 是给切面提供一个唯一标识

ref属性: 是指定通知类bean的Id。

## 4、在aop:aspect标签的内部使用对应标签来配置通知的类型

我们现在示例是让printLog方法在切入点方法执行之前之前: 所以是前置通知

aop:before: 表示配置前置通知

method属性: 用于指定Logger类中哪个方法是前置通知

pointcut属性:用于指定切入点表达式,该表达式的含义指的是对业务层中哪些

方法增强

## 5、切入点表达式的写法:

1) 关键字: execution(表达式)

2) 表达式:

访问修饰符 返回值 包名. 包名. .. 类名. 方法名(参数列表)

### 3) 表达式写法:

#### 1、标准的表达式写法:

public void com.itheima.service.impl.AccountServiceImpl.saveAccount()

- 2、全通配写法: (不建议)
- \* \* . . \* . \* (. . )
- 3、实际开发中切入点表达式的通常写法: (建议使用) 切到业务层实现类下的所有方法
- \* com. itheima. service. impl. \* . \* (...)

类名. 方法(参数)

#### 4、访问修饰符可以省略

void com. itheima. service. impl. AccountServiceImpl. saveAccount()

- 5、返回值可以使用通配符,表示任意返回值
- \* com. itheima. service. impl. AccountServiceImpl. saveAccount()
- 6、包名可以使用通配符,表示任意包。但是有几级包,就需要写几个 \*.
- \* \*. \*. \*. AccountServiceImpl. saveAccount())

- 7、包名可以使用..表示当前包及其子包(下面配置的是任意包下,没有一个包名限制)
- \* \* . . AccountServiceImpl.saveAccount()
- 8、类名和方法名都可以使用\*来实现通配
- \* \* . . \* . \* ()
- 9、参数列表:
  - 1)可以直接写数据类型: 基本类型直接写名称 int 引用类型写包名. 类名的方式 java. lang. String
  - 2) 可以使用通配符表示任意类型,但是必须有参数

```
3) 可以使用 .. 表示有无参数均可,有参数可以是任意类型
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"</p>
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:aop="http://www.springframework.org/schema/aop"
   xsi:schemaLocation="http://www.springframework.org/schema/beans
    http://www.springframework.org/schema/beans/spring-beans.xsd
    http://www.springframework.org/schema/aop
    http://www.springframework.org/schema/aop/spring-aop.xsd">
  <!-- 配置srping的loc,把service对象配置进来-->
  <bean id="accountService"</pre>
class="com.itheima.service.impl.AccountServiceImpl"/>
  <!-- 配置Logger类 -->
  <bean id="logger" class="com.itheima.utils.Logger"/>
  <!--配置AOP-->
  <aop:config>
    <!--配置切面 -->
    <aop:aspect id="logAdvice" ref="logger">
      <!-- 配置通知的类型,并且建立通知方法和切入点方法的关联-->
      <aop:before method="printLog" pointcut="execution(*
com.itheima.service.impl.*.*(..))"/>
    </aop:aspect>
  </aop:config>
</beans>
```