#### Spring Boot

##### 之面向切面编程（AOP）-事务管理 背景

问题描述

Spring Boot 2.2.5

JDK 8

接下来的一篇技术笔记将讨论利用AOP特性实现事务管理注解@Transactional。这篇笔记将对@Transactional注解做一个简单的介绍。

解决

1. 建表并导入测试数据

/\*

Navicat MySQL Data Transfer

Source Server : localdb

Source Server Version : 50727

Source Host : localhost:3306

Source Database : transactional

Target Server Type : MYSQL

Target Server Version : 50727

File Encoding : 65001

Date: 2020-05-22 08:19:53

\*/

SET FOREIGN\_KEY\_CHECKS=0;

-- ----------------------------

-- Table structure for products

-- ----------------------------

DROP TABLE IF EXISTS `products`;

CREATE TABLE `products` (

`id` int(11) NOT NULL,

`name` varchar(255) NOT NULL,

`description` varchar(255) DEFAULT NULL,

`price` decimal(10,2) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- ----------------------------

-- Records of products

-- ----------------------------

INSERT INTO `products` VALUES ('1001', 'ProductA', 'Product A description', '1.00');

INSERT INTO `products` VALUES ('1002', 'ProductB', 'Product B description', '2.00');

INSERT INTO `products` VALUES ('1003', 'ProductC', 'Product C description', '3.00');

INSERT INTO `products` VALUES ('1004', 'ProductD', 'Product D description', '4.00');

INSERT INTO `products` VALUES ('1005', 'ProductE', 'Product E description', '5.00');

1. 添加@Transactional注解

@Transactional(transactionManager = "getTx")  
 public void queryAnimal() {  
// 可以正常执行  
 jdbcTemplate.execute("insert into products (id, name, description, price) values(1006,'ProductE', 'Product E description', 5)");  
// 抛出异常Duplicate entry '1005' for key 'PRIMARY'  
 jdbcTemplate.execute("insert into products (id, name, description, price) values(1005,'ProductF', 'Product F description', 6)");  
 }

1. 观察结果

java.sql.SQLIntegrityConstraintViolationException: Duplicate entry '1005' for key 'PRIMARY'

打印的异常堆栈从顶到底大致分为三块：

{ jdbc }

{ 动态代理}

{ Tomcat }

笔记

1. Transactional注解加在直接调用的方法上面才可以，否则不会回滚。原因在于Spring Framework会过滤@Transactional注解，并自动生成代理类。只有调用到代理类中的相应方法才会进入事务（Transaction）

□