# 异常测试

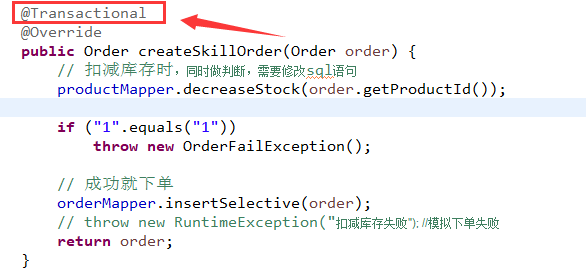
## 异常



**结果库存扣除成功，但是订单没创建**

## 解决

使用事务



# 传播机制

## 定义

事务传播行为用来描述由某一个事务传播行为修饰的方法被嵌套进另一个方法的时事务如何传播。

用伪代码说明：

public void methodA(){

methodB();

//doSomething

}

@Transaction(Propagation=XXX)

public void methodB(){

//doSomething

}

代码中methodA()方法嵌套调用了methodB()方法，methodB()的事务传播行为由@Transaction(Propagation=XXX)设置决定。这里需要注意的是methodA()并没有开启事务，某一个事务传播行为修饰的方法并不是必须要在开启事务的外围方法中调用。

## 分类



## 代码

测试参考：<https://segmentfault.com/a/1190000013341344>

# 简单总结

**NESTED和REQUIRED修饰的内部方法都属于外围方法事务，如果外围方法抛出异常，这两种方法的事务都会被回滚。但是REQUIRED是加入外围方法事务，所以和外围事务同属于一个事务，一旦REQUIRED事务抛出异常被回滚，外围方法事务也将被回滚。而NESTED是外围方法的子事务，有单独的保存点，所以NESTED方法抛出异常被回滚，不会影响到外围方法的事务。**

**NESTED和REQUIRES\_NEW都可以做到内部方法事务回滚而不影响外围方法事务。但是因为NESTED是嵌套事务，所以外围方法回滚之后，作为外围方法事务的子事务也会被回滚。而REQUIRES\_NEW是通过开启新的事务实现的，内部事务和外围事务是两个事务，外围事务回滚不会影响内部事务。**

# **业务测试**

