spring中aop事务

事务：

* 事务特性：acid
* 事务并发问题：
* 脏读
* 不可重复读
* 幻读
* 事务的隔离级别：
* 读未提交
* 读已提交
* 可重复读
* 串行化

spring封装了事务管理代码：

* 事务操作：
* 打开事务
* 提交事务
* 回滚事务
* 事务操作对象：
* 因为在不同平台，操作事务的代码各不相同，spring提供了一个接口
* **PlatformTransactionManager（平台事务管理器）在spring中玩事务管理，最为核心的对象就是TransactionManager对象**
* **spring管理事务的属性介绍**
* **事务的隔离级别**
* 读未提交
* 读已提交
* 可重复读
* 串行化
* **是否只读**
* true只读
* false可操作
* **事务的传播行为**

**决定业务方法之间调用，业务应该如何处理的问题**

事务原本是数据库中的概念，在Dao层。但一般情况下，需要将事务提升到业务层，即Service层。这样做是为了能够使用事务的特性来管理具体的业务。

在Spring中通常可以通过以下三种方式来实现对事务的管理：

1. 使用Spring的事务代理工厂管理事务
2. 使用Spring的事务注解管理事务
3. 使用AspectJ的AOP配置管理事务
4. **Spring事务管理API**

Spring的事务管理，主要用到两个事务相关的接口：

1. **事务管理器接口**

事务管理器是PlatformTransactionManager接口对象。其主要用于完成事物的提交、回滚，及获取事务的状态信息。

1. 常用的两个实现类

PlatformTransactionManager接口有两个常用的实现类：

* DataSourceTransactionManager：使用JDBC或iBatis进行持久化数据时使用。
* HibernateTransactionManager：使用Hibernate进行持久化数据时使用。

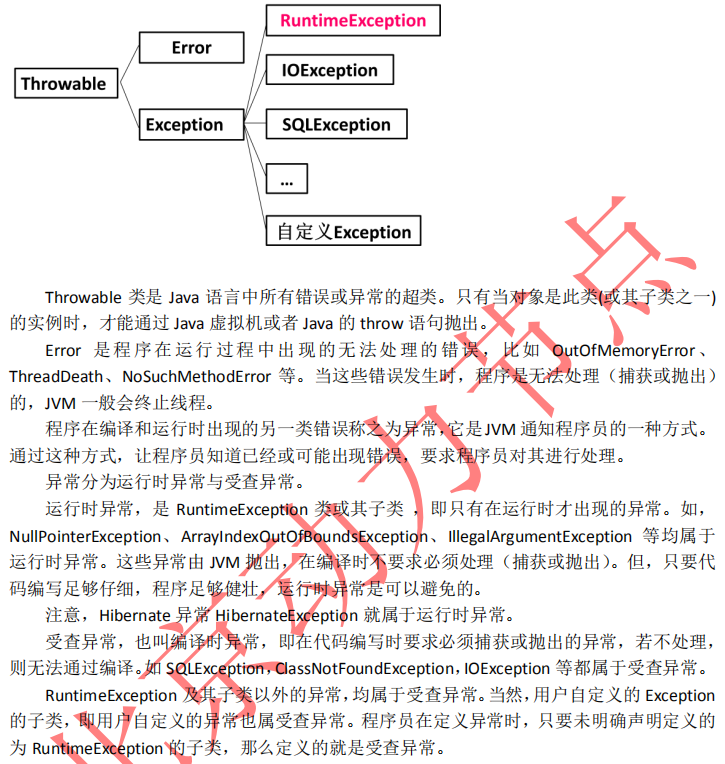
1. Spring的回滚方式

Spring事务的默认回滚方式是：发生运行异常（严重）时回滚，发生受查异常时提交。不过，对于受查异常，程序员也可以设置其回滚方法。

1. 回顾错误与异常

程序举例环境搭建

1. 使用Spring的事务代理工厂管理事务
2. 使用Spring的事务注解管理事务
3. 使用AspectJ的AOP配置管理事务（重点）



1. **事务定义接口**

事务定义接口TransactionDefinition中定义了事务描述相关的三类常量：事务的隔离级别、事务的传播行为、事务默认超时时限，及对他们的操作。

1. **定义了五个事务隔离级别的常量**

这些常量均为ISOLATION\_开头。

* DEFAULT：采用DB默认的事务隔离级别。MySql的默认级为REPEATABLE\_READ；Oracle默认为READ\_COMMITTED。
* READ\_UNCOMMITTED：读未提交。未解决任何并发问题。
* READ\_COMMITTED：读已提交。解决脏读，存在不可重复读与幻读。
* REPEATABLE\_READ：可重复读。解决脏读、不可重复读，存在幻读。
* SERIALIZABLE：串行化。不存在并发问题。

1. **定义了七个事务传播行为常量**

所谓事务传播行为是指，处于不同事务中的方法在相互调用时，执行期间事务的维护情况。如，A事务中的方法doSome()调用B事务中的方法doOther()，在调用执行期间事务的维护情况，就称为事务传播行为。事务传播行为是加在方法上的。

事务传播行为常量都是以PROPAGATION\_开头，形如PROPAGATION\_XXX。

**PROPAGATION\_REQUIRED（最常用）：支持当前事务，如果不存在，就新建一个（默认）。**

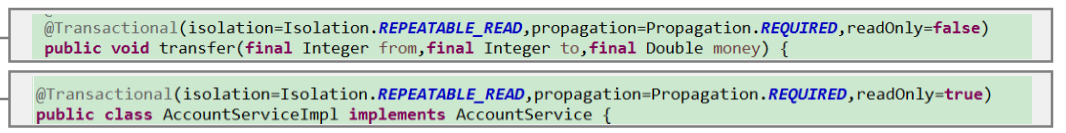


1. **定义了默认事务超时时限**

常量TIMEOUT\_DEFAULT定义了事务底层默认的超时时限，及不支持事务超时时限设置的more值。

注意，事务的超时时限起作用的条件比较多，且超时的时间计算点较复杂。所以，该值一般使用默认值即可。

1. **程序举例环境搭建**
2. **使用Spring的事务代理工厂管理事务**
3. **使用Spring的事务注解管理事务**



1. **使用AspectJ的AOP配置管理事务（重点）**