# 事务

1,什么是事务?

在业务代码发生异常时,保护数据库一致性的一种技术.

2,JDBC里怎么做事务的手动提交和回滚

**try** {

conn = DriverManager.*getConnection*(url, "s1", "s1");

conn.setAutoCommit(**false**); //true 自动提交(默认) false取消自动提交

stat = conn.createStatement();

//写业务代码

conn.commit();//事务提交

//代码出问题时,事务用来保护数据库的完整性,原子型

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

conn.rollback();//事务回滚

}

3,在web项目中添加事务(声明式)

<!-- TransactionManager 事务管理 -->

<bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTransactionManager">

<property name="dataSource" ref="dataSource" />

</bean>

<aop:config>

<aop:pointcut id=*"point1"* expression=*"execution(\* com.turing.\*.service.impl.\*.\*(..))"* />

<aop:advisor advice-ref=*"txAdvice"* pointcut-ref=*"point1"* />

</aop:config>

<aop:aspectj-autoproxy proxy-target-class=*"true"* />

<!-- 事务的拓展性,主要是针对类中的方法,绝对哪些方法有事务,哪些方法不需要事务 🡪

<tx:advice id=*"txAdvice"* transaction-manager=*"transactionManager"*>

<tx:attributes>

<tx:method name=*"query\*"* propagation=*"NOT\_SUPPORTED"* read-only=*"true"* />

<tx:method name=*"get\*"* propagation=*"NOT\_SUPPORTED"* read-only=*"true"* />

<tx:method name=*"select\*"* propagation=*"NOT\_SUPPORTED"* read-only=*"true"* />

<tx:method name=*"\*"* propagation=*"REQUIRED"* />

</tx:attributes>

</tx:advice>

4, 事务使用的类型

PROPAGATION\_REQUIRED

支持当前事务，如果当前没有事务，就新建一个事务。这是最常见的选择。

PROPAGATION\_SUPPORTS

支持当前事务，如果当前没有事务，就以非事务方式执行。

PROPAGATION\_MANDATORY

支持当前事务，如果当前没有事务，就抛出异常。

PROPAGATION\_REQUIRES\_NEW

新建事务，如果当前存在事务，把当前事务挂起。

PROPAGATION\_NOT\_SUPPORTED

以非事务方式执行操作，如果当前存在事务，就把当前事务挂起。

PROPAGATION\_NEVER

以非事务方式执行，如果当前存在事务，则抛出异常。

PROPAGATION\_NESTED

如果当前存在事务，则在嵌套事务内执行。如果当前没有事务，则进行与

PROPAGATION\_REQUIRED类似的操作。

5,声明式事务,必须看懂的两个地方

5.1 ,横切的路径,我们要把自己的模型层放在横切的包内.

5.2,事务的拓展性,必须安装要求写自己的方法名.

6, 在web项目中添加事务(注解式)

6.1 在spring容器内添加启动事务注解的驱动

<tx:annotation-driven transaction-manager="transactionManager" />

6.2 在需要添加事务的方法上,添加事务注解 @Transactional

@Transactional可以放在方法上,也可以放在类上

如果放在类上,则类里所有方法都具有事务的功能.

7,事务的特性

7.1 原子性

个事务是一个不可分割的工作单位，事务中包括的诸操作要么都做，要么都不做

7.2 一致性

事务必须是使数据库从一个一致性状态变到另一个一致性状态

7.3 隔离性

一个事务的执行不能被其他事务干扰。即一个事务内部的操作及使用的数据对并发的其他事务是隔离的，并发执行的各个事务之间不能互相干扰

7.4 持久性

持久性也称永久性, 指一个事务一旦提交，它对数据库中数据的改变就应该是永久性的,不可逆的.

10,面试题

1.什么是事务

2.事务的特性有哪些

3.什么是事务的隔离性

4.在SSM框架中，什么地方需要配置事务