1. 事务
2. 概述

事务管理着一个包含多个步骤的业务操作，这些步骤要么同时成功，要么同时失败。当这些步骤中出现异常错误时，事务会及时地回滚到业务初始状态，防止出现逻辑错误。

1. 操作
2. 开启事务

start transaction;

1. 回滚

rollback; 回滚到开启事务的位置、

1. 提交

commit;

1. 事务自动提交与手动提交
2. 自动提交

一条DML（增删改）语句会自动提交一次事务

1. 手动提交

需要先开启事务，再提交

1. 修改事务的默认提交方式
2. 查看事务的默认提交方式：

SELECT @@autocommit；

1 代表自动提交， 0代表手动提交

1. 修改默认提交方式

SET @@autocommit = 0；

1. 事务原理

务开启之后, 所有的操作都会临时保存到事务日志中, 事务日志只有在得到 commit 命令才会同步到数据表

中，其他任何情况都会清空事务日志(rollback，断开连接)

1. 事务的步骤

1.客户端连接数据库服务器，创建连接时创建此用户临时日志文件

2． 开启事务以后，所有的操作都会先写入到临时日志文件中

3． 所有的查询操作从表中查询，但会经过日志文件加工后才返回

4． 如果事务提交则将日志文件中的数据写到表中，否则清空日志文件。

1. 回滚点
2. 回滚点概述

在某些成功的操作完成之后，后续的操作有可能成功有可能失败，但是不管成功还是失败，前面操作都已经成

功，可以在当前成功的位置设置一个回滚点。可以供后续失败操作返回到该位置，而不是返回所有操作，这个点称

之为回滚点

设置回滚点可以让我们在失败时回到回滚点，而不是回到事务开启的位置。

1. 回滚点操作

作用 语句

设置回滚点 savepoint 名字

回到回滚点 rollback to 名字

1. 事务的四大特征ACID
2. 原子性(Atomicity)

事务是一个整体，不可分割的最小操作单位，要么同时成功，要么同时失败。

1. 持久性(Durability)

当事务提交或回滚后，数据库会持久化地保存数据。

1. 隔离性(Isolation)

多个事务之间是相互独立的。

1. 一致性(Consistency)

事务在操作前后的数据总量是不变的，若变动了，则说明有异常出现。

1. 事务的隔离级别
2. 概述

多个事务之间是独立的，但当多个事务操作同一批数据时，会引发并发访问的问题，通过设置事务的隔离级别来解决。

1. 存在的问题
2. 脏读

一个事务读取到另一个事务中没有提交的数据。如一个事务A中改变的数据，被事务B读取了，但是事务A之后将事务回滚了，则事务B读取到的数据就是错误的。

1. 不可重复读（虚读）

在同一个事务中，两次读取到的数据内容不同，这是事务UPDATE时引发的问题。如事务A中改变的数据，事务B第一次读取结果为R1，事务A将业务回滚后，事务B第二次读取结果为R2，在事务B这一个事务中，读取同一数据结果不同，容易混淆、

1. 幻读

一个事务中两次读取的数据的 数量 不一致，要求在一个事务多次读取的数据的数量是一致的，这是 insert 或 delete时引发的问题。

1. 四种隔离级别
2. read uncommitted(读未提交)

产生的问题：脏读，不可重复读，幻读

1. read Committed(读已提交)

当一个事务提交后，该事务的数据才能够读取到。

产生的问题：不可重复读，幻读

解决的问题：脏读

Oracle默认

1. repeatable read(可重复读)

在一个事务中，可以重复地读取一个数据且该数据不会改变。

产生的问题：幻读

解决的问题：脏读，不可重复读

Mysql默认

1. serializable(串行化)

类似于锁的操作，当一个事务在操作一个表时，其他事务是无法操作这个表的，只能等待这个表被释放。

不会产生前面的三种问题。

从上到下，隔离级别越高，性能越差，安全性越高。

1. 查询隔离级别

SELECT @@tx\_isolation;

1. 设置隔离级别

set global transaction isolation level 级别字符串；