transaction control language 事务控制语言

事务：一个或一组sql语句组成一个执行单元，这个执行单元要么全部执行，要么全部不执行。

案例：转账

存储引擎：

1.在mysql中的数据用各种不同的计数存储在文件（或内存）中

2.通过show engines来查看mysql支持的存储引擎

3.在mysql中用的最多的存储引擎有：innodb，myisam，memory等。其中innodb支持事务，而myisam，memory等不支持事务。

事务的ACID属性

1.原子性：Atomicity：原子性是指事务是一个不可分割的工作单位，事务中的操作要么都发生，要么都不发生

2.一致性：Consistency：事务必须使数据库从一个一致性状态到另一个一致性的状态。

3.隔离性：Isolation：事务的隔离性是指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。

4.持久性：Durability：持久性是指一个事务一旦被提交，它对数据库中的数据的改变就是永久性的，接下来的其他操作和数据库故障都不应该对其有任何影响。

事务的创建：

隐式事务：事务没有明显的开启和结束的标记。

比如：insert、update、delete

显示事务：事务具有明显的开启和结束的标记。

前提：比如先把自动提交功能关闭

set autocomit=0;

开启事务的语句：

步骤1：开启事务

set autocommit=0;

start transaction;可选的

步骤2：编写事务中的sql语句(select insert update delete)

语句1；

语句2；

……

步骤3：结束事务

commit；提交事务

rollback；回滚事务

savepoint;保存点，只搭配rollback使用

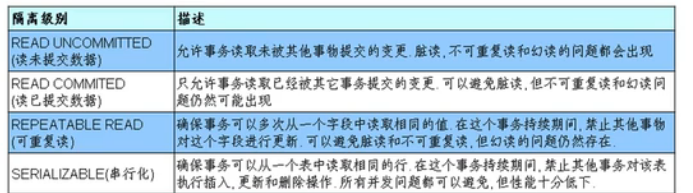
当多个事务同时访问数据库中相同的数据时，如果没有采取必要的隔离机制，就会导致各种并发的问题

脏读：对于两个事务T1，T2，T1读取了已经被T2更新但还没有被提交的字段之后，若T2回滚，T1读取的内容就是临时且无效的。

不可重复读：对于两个事务T1，T2，T1读取了一个字段，然后T2更新了该字段之后，T1再次读取他一字段，值就不同了。

幻读：对于两个事务T1，T2，T1从一个表中读取了一个字段，然后T2在该表中插入了一些新行。之后，如果T1再次读取同一个表，就会多出几行。

数据库提供的4种事务隔离级别



Oracle支持的2种事务隔离级别：read commited，serializable。Oracle默认的事务隔离级别为：read commited。

mysql支持4种事务隔离级别。Mysql默认的事务隔离级别为：repeatable read。

select @@tx\_isolation