**任务描述**

本关任务： 编写一个银行卡转账的方法。

**相关知识**

为了完成本关任务，你需要掌握： 1.JDBC的基本知识(参见第1关); 2.JDBC的事务处理。

**JDBC的事务处理**

JDBC缺省情形下，Statement和PreapredStatement均自动为一个事务。因此，当一个Statement调了executeUpdate()方法后，所执行的SQL语句自动提交，其对数据库的修改不可再撤销。

但调用Connection.setAutoCommit(boolean autoCommit)方法可以改变缺省设置： connection.setAutoCommit(false); 将使本会话期内的语句不再自动提交，必须调用Connection的以下方法手动提交：

* commit()
* rollback()

前者为正常提交，后者表示事务回滚，撤销所有修改。

**JDBC的事务隔离级别**

JDBC通过调用Connetion的以下方法设置事务的隔离级别： void setTransactionIsolation(int level) 如果没有调用该方法设置隔离级别，将采用DBMS缺省的隔离级别，对MySQL而言，即repeatable read。JDBC支持以下隔离级别：

* static int TRANSACTION\_NONE
* static int TRANSACTION\_READ\_UNCOMMITTED
* static int TRANSACTION\_READ\_COMMITTED
* static int TRANSACTION\_REPEATABLE\_READ
* static int TRANSACTION\_SERIALIZABLE

它们对应的int值依次为0,1,2,4和8。 不是每个DBMS都支持以上所有的隔离级别。MySQL支持后面4个隔离级别，且缺省状态下为TRANSACTION\_REPEATABLE\_READ。Oracle支持其中的两个，缺省值为TRANSACTION\_READ\_COMMITTED。

**编程要求**

编写一个银行卡转账的方法transferBalance()。

transferBalance()在被调用前，柜台已经确认过转出帐号持有者身份，所以转帐方法只接受转出卡号，转入卡号和转账金额三个参数。由调用者保证转账金额为正数。 transferBalance()方法的每个参数都在注释中进行了详细的说明。 transferBalance()返回boolean值，true表示转帐成功，false表示转账失败，并不需要细分或解释失败的原因。 下列任一情形都不可转账(转账失败的原因)： 转出或转入帐号不存在 转出账号是信用卡 转出帐号余额不足

提示：

1.本方法需开启手工事务,并正确使用commit和rollback；

2.当转入卡是信用卡时，意指信用卡还款，还款可以超过透支款项； 3.对事务的隔离级别不作要求。

请不要修改main()方法。

附:bank\_card(银行卡)的表结构:

| **字段名称** | **数据类型** | **约束** | **说明** |
| --- | --- | --- | --- |
| b\_number | CHAR(30) | PRIMARY KEY | 银行卡号 |
| b\_type | CHAR(20) | 无 | 银行卡类型(储蓄卡/信用卡) |
| b\_c\_id | INTEGER | NOT NULL FOREIGN KEY | 所属客户编号,引用自client表的c\_id字段。 |
| b\_balance | NUMERIC(10,2) | NOT NULL | 余额,信用卡余额系指已透支的金额 |

**测试说明**

平台首先编译代码文件，然后执行，最后检查表数据的变化，如果程序对所有输入都能得出预期的结果，即可通关。