《数据库系统概论》

课程设计报告

《银行储蓄业务管理系统》

班级：

学号：

姓名：

目录

[一、系统需求分析 5](#_Toc487448818)

[（一）需求概述 5](#_Toc487448819)

[（二）业务流分析 5](#_Toc487448824)

[（三）数据流分析 6](#_Toc487448831)

[（四）数据字典 8](#_Toc487448833)

[二、数据库概念结构设计 11](#_Toc487448961)

[（一）实体分析 11](#_Toc487448962)

[（二）属性分析 错误!未定义书签。](#_Toc487448963)

[（三）联系分析 13](#_Toc487448964)

[（四）概念模型分析（.PDM图） 15](#_Toc487448965)

[三、数据库逻辑逻辑结构设计 17](#_Toc487448968)

[（一）概念模型转化为逻辑模型 17](#_Toc487448969)

[1. 一对一关系的转化 17](#_Toc487448970)

[2. 一对多关系的转化 17](#_Toc487448971)

[3. 多对多关系的转化 17](#_Toc487448972)

[（二）逻辑模型设计（.PDM图） 18](#_Toc487448973)

[四、数据库物理实现 19](#_Toc487448975)

[（一）表设计 19](#_Toc487448976)

[（二）创建表和完整性约束代码设计 21](#_Toc487449098)

[（三）创建视图、索引、存储过程和触发器 23](#_Toc487449105)

[1. 创建视图 23](#_Toc487449106)

[2. 创建索引 23](#_Toc487449109)

[3. 创建存储过程 24](#_Toc487449111)

[五、数据库功能调试 25](#_Toc487449114)

[（一）储户管理模块 25](#_Toc487449115)

[1. 查询储户信息 25](#_Toc487449116)

[2. 查询存款单信息 错误!未定义书签。](#_Toc487449117)

[3. 查询取款单信息 错误!未定义书签。](#_Toc487449118)

[（二）工程负责人管理模块 26](#_Toc487449119)

[1. 查询工程信息 26](#_Toc487449120)

[2. 查询工程所需设备信息 27](#_Toc487449122)

[3. 查询工程供应商信息 28](#_Toc487449124)

[4. 查询参与工程的职工信息 28](#_Toc487449126)

[（三）系统管理员管理模块 30](#_Toc487449128)

[1. 对储户信息管理 30](#_Toc487449129)

[2. 对存款单管理 32](#_Toc487449130)

[3. 对取款单信息管理 34](#_Toc487449131)

[6. 对供应关系的信息管理 错误!未定义书签。](#_Toc487449134)

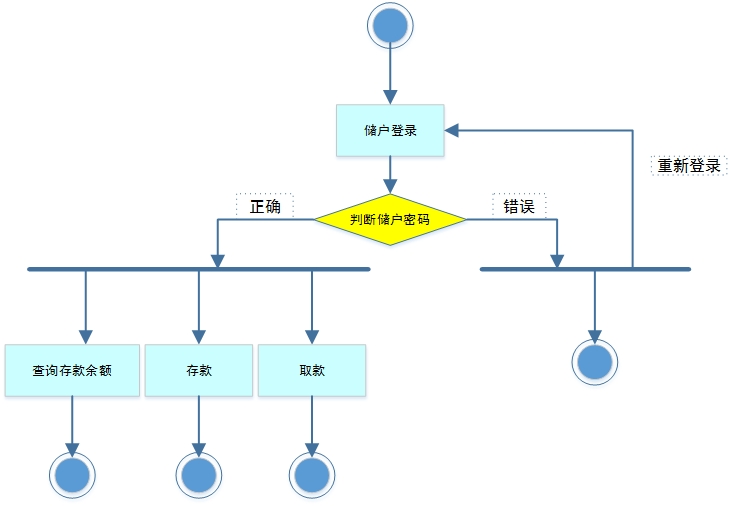
[六、设计总结 39](#_Toc487449135)

# 系统需求分析

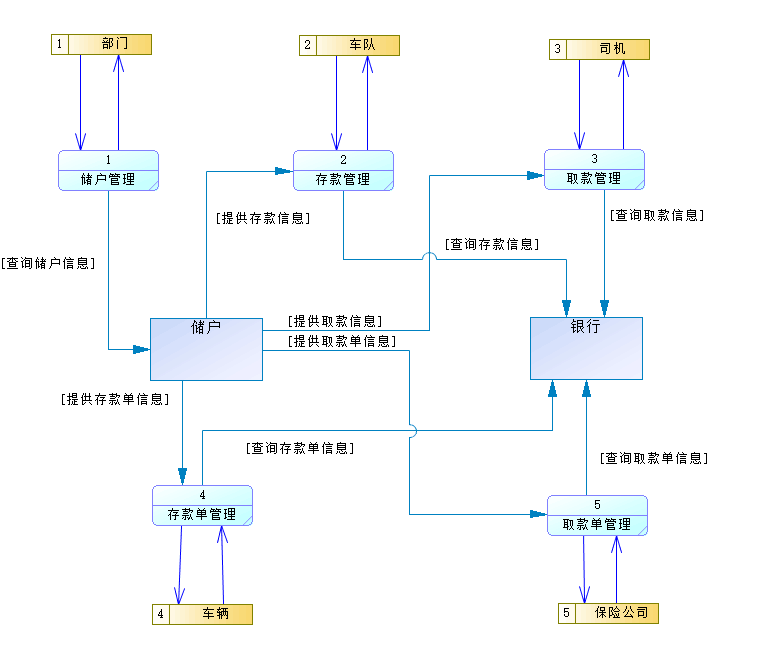
1. 需求概述

本次课程设计的题目是银行储蓄业务管理系统，业务流分析:

本软件实现银行的活期储蓄存款业务,整套软件由客户端程序和服务端程序构成,,客户端运行在面向个人业务的银行柜台的电脑上,银行职员输入相关信息并通过网络通讯向服务器发送请求,并接受服务器端程序的应答显示处理的结果.服务器端处理客户端的各种业务请求.主要实现银行的开户、销户、存款、取款、余额查询、转账、修改密码等功能。



1. 数据流分析



1. 数据字典

数据项：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据项名 | 含义说明 | 数据类型 | 长度 | 取值范围 | 数据项联系 |
| 储户帐号 | 区别储户 | vchar | 20 |  | 一个储户对应一个账号 |
| 身份证号 | 区别储户 | vchar | 20 |  | 一个储户名对应一个身份证号 |
| 姓名 | 区别储户 | vchar | 20 |  | 一个储户有一个姓名 |
| 地址 | 区别储户 | vchar | 20 |  | 一个储户对应一个地址名 |
| 存款余额 | 储户信息 | vchar | 20 |  |  |
| 存款单号 | 标注存款单 | vchar | 20 |  | **一个存款单号对应一个取款单** |
| 存款金额 | 存款信息 | vchar | 20 |  |  |
| 存款方式 | **存款信息** | vchar | 20 |  |  |
| 取款单号 | 标注取款单 | vchar | 20 |  | 一个取款单号对应一个取款单 |
| 取款金额 |  | vchar | 20 |  |  |
| 取款方式 |  | vchar | 20 |  |  |
| 存款日期 |  | vchar | 20 |  | 一个存款日期对应一个存款 |
| 取款日期 |  | vchar | 20 |  | 一个取款日期对应一个取款 |

数据结构：

|  |  |  |
| --- | --- | --- |
| 数据结构名 | 含义名 | 组成 |
| 储户 | 储户信息 | 账号、身份证号、姓名、地址、存款余额 |
| 存款单 | 存款信息 | 存款单号、金额、存款方式 |
| 取款单 | 取款信息 | 取款单号、金额、取款方式 |
| 存款 | 日期信息 | 存款日期 |
| 取款 | 日期信息 | 取款日期 |

数据

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 数据流来源 | 数据流去向 | 组成 |
| 储户管理 | 储户自己 | 银行负责人 | 储户信息 |
| 存款管理 | 储户自己 | 储户 | 存款单信息 |
| 取款管理 | 储户自己 | 储户 | 取款单信息 |
| 存款日期 | 银行 | 银行、储户 | 存款信息 |
| 取款日期 | 银行 | 银行、储户 | 取款信息 |

数据存储：

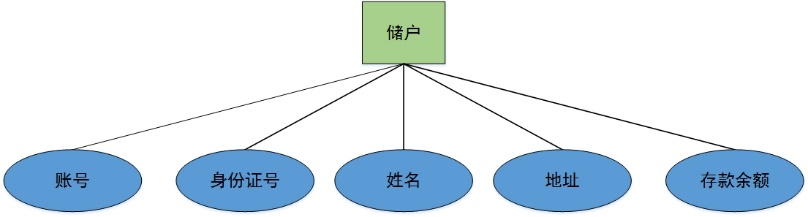
|  |  |  |  |
| --- | --- | --- | --- |
| 数据存储名 | 输入的数据流 | 输出的数据流 | 组成 |
| 储户表 | 储户信息 | 职工信息 | 储户信息 |
| 存款表 | 储户信息、存款单 | 存款日期 | 存款日期、储户信息、存款单 |
| 取款表 | 储户信息、取款单 | 取款日期 | 取款日期、储户信息、取款单 |
| 存款单表 |  | 取款单 | 取款单 |
| 取款单表 |  | 取款单 | 取款单 |

# 数据库概念结构设计

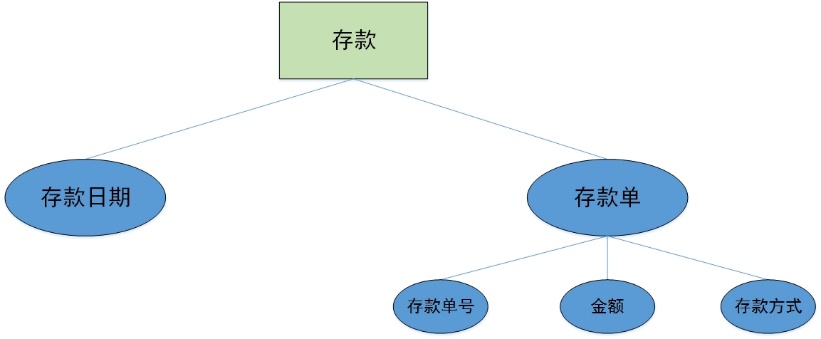
1. 实体分析

经需求分析，本次课程设计中包含五个实体，他们分别是储户实体、取款实体、取款单实体、存款实体、存款单实体

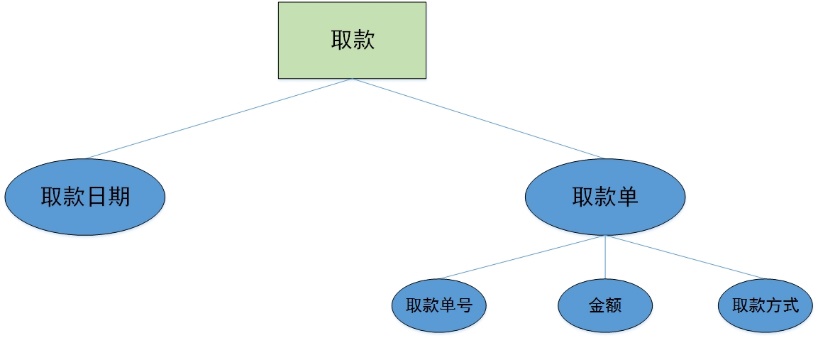
储户属性：账号、身份证号、姓名、地址、存款余额如下：



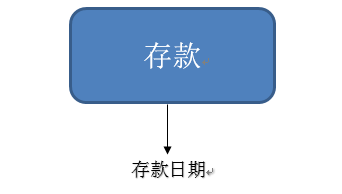
存款单属性：存款单号、存款金额、存款方式如下：



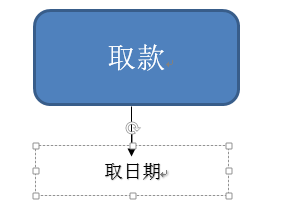
取款单号属性：取款单号、取款金额、取款方式如下：



存款属性：存款日期。如下:



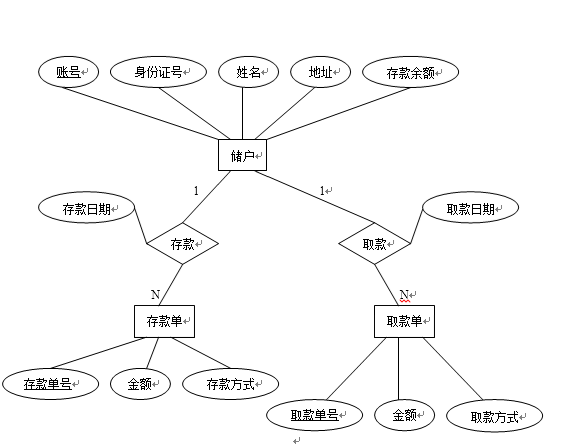
取款属性：取款日期。如下：



1. 联系分析

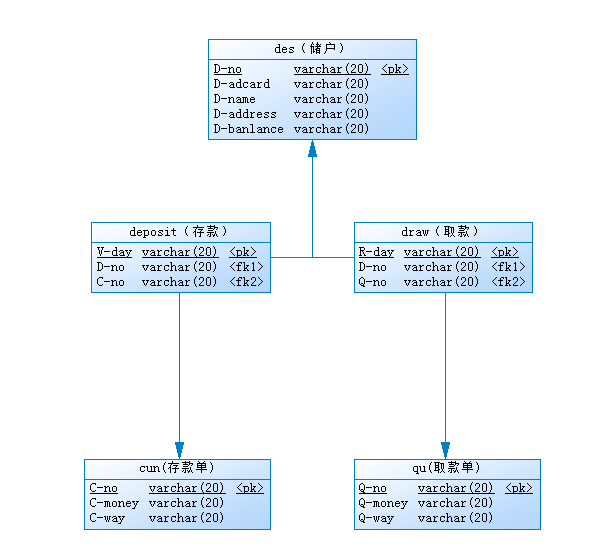
一个储户在一次操作中只能用一个账号、身份证号姓名、地址、以及只能输入一个存款金额，储户可以进行存款和取款两项业务，在进行存款和取款当中又有单号、金额、和方式以及时间，每个储户对应的业务操作都是一条线一对一下去的，但是每个业务都是可以有很多储户来进行业务办理的。

E-R图分析如下：



1. 概念模型分析（.PDM图）

在经过需求分析和实体属性的分析，以及各实体之间的关系，最终得到概念模型如下：



# 三、数据库逻辑逻辑结构设计

（一）概念模型转化为逻辑模型

1. 一对一关系的转化

在银行储蓄业务系统中没有一对一关系的转化

1. 一对多关系的转化

一个账号只属于一个储户，但一个储户有多个账号。

一个存款单只能参加一次存款，存款需要多个存款单参与。

一个取单只能参加一次取款取款，取款需要多个取款单参与。

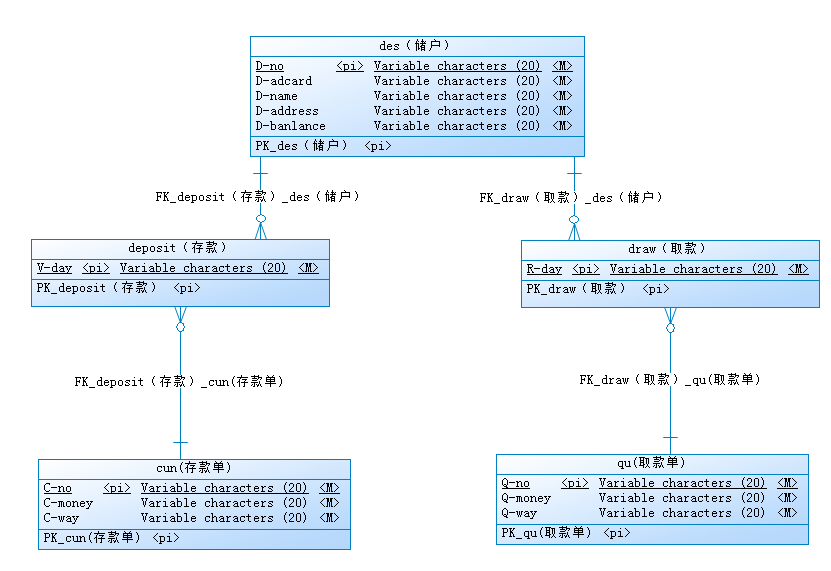
存款：存款日期（主键）、储户账号（外键）、存款单号（外键）

取款：取款日期（主键）、储户账号（外键）、取款单号（外键）

1. 多对多关系的转化

银行储蓄业务管理系统不存在多对多关系

（二）逻辑模型设计（CDM图）



# 数据库物理实现

1. 表设计

1.储户信息表设计

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段解释 | 数据类型 | 备注 |
| D-no | 储户帐号 | vchar(20) | 主键 |
| D-adcard | 身份证号 | vchar(20) | 非空 |
| D-name | 姓名 | vchar(20) | 非空 |
| D-address | 地址 | vchar(20) | 非空 |
| D-banlance | 存款余额 | Vchar(20) | 非空 |

2.存款单信息表设计

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段解释 | 数据类型 | 备注 |
| C-no | 存款单号 | vchar(20) | 主键 |
| C-money | 存款金额 | vchar(20) | 非空 |
| C-way | 存款方式 | vchar(20) | 非空 |

3.取款单信息表设计：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段解释 | 数据类型 | 备注 |
| Q-no | 存款单号 | vchar(20) | 主键 |
| Q-money | 存款金额 | vchar(20) | 非空 |
| Q-way | 存款方式 | vchar(20) | 非空 |

4.存款信息表设计：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段解释 | 数据类型 | 备注 |
| V-day | 存款日期 | vchar(20) | 主键 |

5.取款信息表设计：

|  |  |  |  |
| --- | --- | --- | --- |
| 字段名称 | 字段解释 | 数据类型 | 备注 |
| R-day | 取款日期 | vchar(20) | 主键 |

1. 创建表和完整性约束代码设计
2. 创建储户表及完整性约束：

CREATE TABLE des（储户）(

D-No varchar(20) primary key,

D-adcard varchar(20) NOT NULL,

D-name varchar(20) NOT NULL,

D-banlance varchar(20) NOT NULL,

D-address varchar(20) NOT NULL,)

1. 创建存款单表及完整性约束：

CREATE TABLE cun（存款单）(

C-no varchar(20) primary key,

C-money varchar(20) not null,

C-way varchar(20) not null,)

1. 创建取款单及完整性约束：

CREATE TABLE qu（取款单）(

Q-no varchar(20) primary key,

Q-money varchar(20) not null,

Q-way varchar(20) not null,)

1. 创建存款表及完整性约束：

CREATE TABLE deposit（存款）(

V-day varchar(20) primary key,)

1. 创建取款表及完整性约束:

CREATE TABLE draw（取款）(

R-day varchar(20) primary key,)

1. 创建视图、索引、存储过程和触发器
2. 创建视图
3. 创建储户的信息的视图。

create view P1

as

select des.D-no ,D-adcard ,D-name，D-address，D-banlance

from des

WHERE C-No = 1

1. 创建储户存款单的信息的视图。

create view P2

as

select .C-no ,C-money ,C-way

from des,

WHERE C-No = 1 and Supplier.C-No =Supply.C-No

1. 创建索引

为改数据库中的Supply和officer两个表建立索引。其中Supply表按照设备数量升序建立唯一索引。Officer表按照职工工资升序建立唯一索引。

CREATE UNIQUE INDEX des on despit;

CREATE UNIQUE INDEX cun on cunkuan;

//我不会的而且不重要的

创建存储过程

1. 建立存储过程，输入职工号得到该职工的所有信息。

create procedure zhigong

@officerNo char (4)

as

select \*

from officer

where officer.oNo = @officerNo

1. 建立存储过程，输入工程号得到该工程需要的设备信息。

create procedure shebei

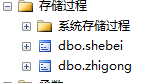
@pNo char (4)

as

select Equipment.eNo ,eName ,price

from Supply ,Equipment

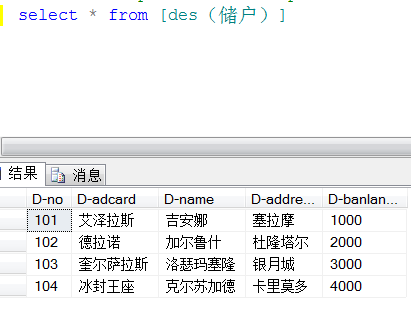
where Supply .eNo = Equipment .eNo and Supply .pNo =@pNo



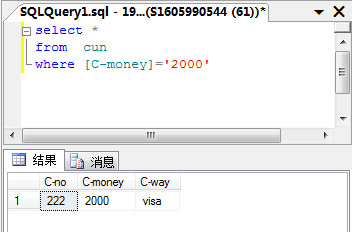
# //到这

# 五、数据库功能调试

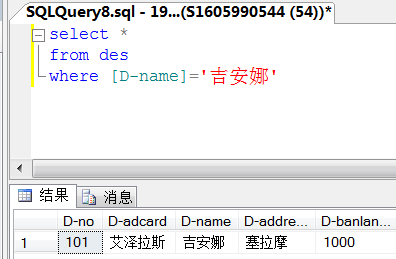
1. 存款管理模块
2. 查询储户信息



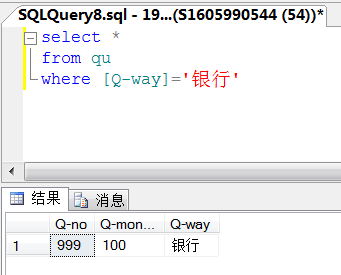
1. 工程负责人管理模块
2. 查询存款单信息



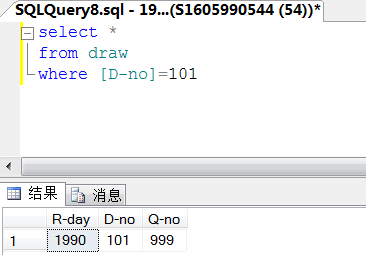
1. 查询储户信息



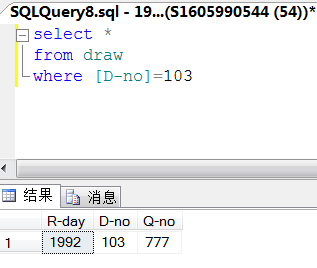
1. 查询取款单信息



1. 查询存款信息

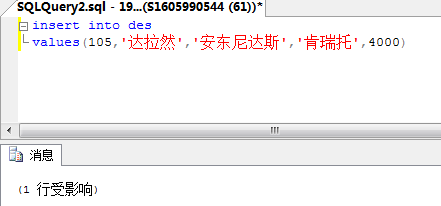


1. 查询取款信息

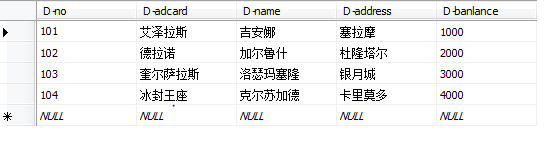
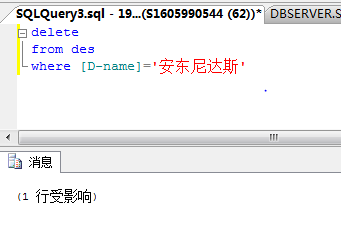


1. 系统管理员管理模块
2. 对储户的信息管理

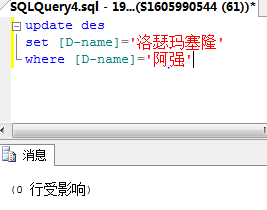
1.1插入储户信息

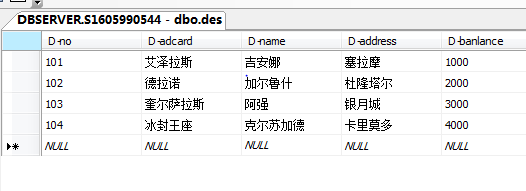


1.2删除储户信息



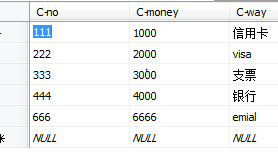
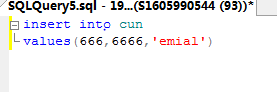
1.3修改储户信息



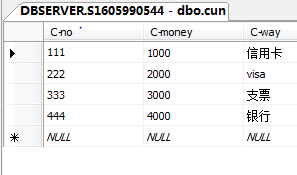
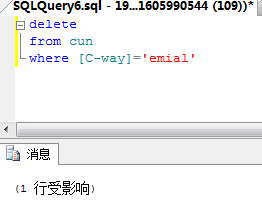


1. 对存款单的信息管理

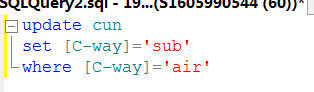
2.1插入存款单信息



2.2删除存款单信息

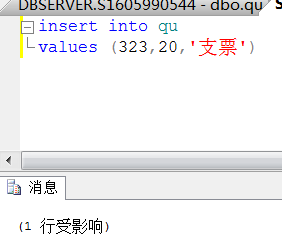


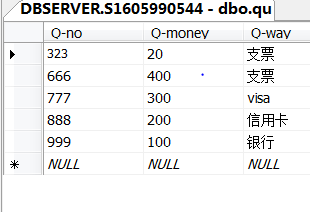
2.3修改存款单信息



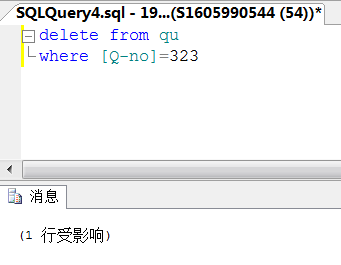
1. 对取款单的信息管理

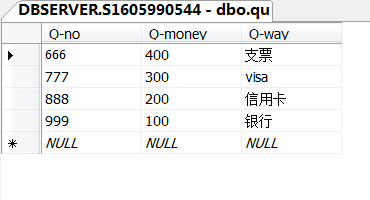
3.1插入取款单信息



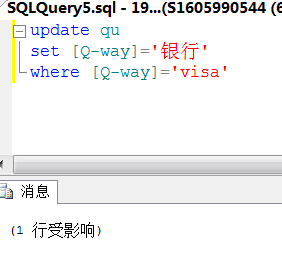


3.2删除取款单信息



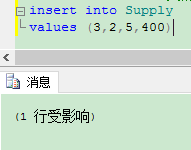


3.3修改取款单信息

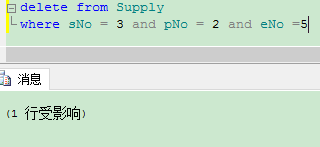
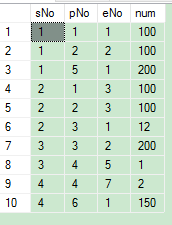




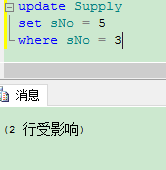
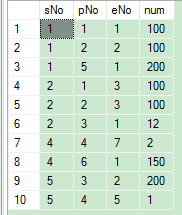
6.1插入供应关系信息

6.2删除供应关系信息

6.3修改供应关系信息

# 六、设计总结

通过这次项目，我意识到自己所需的知识远远超出了已有的认知范围。在查阅资料的过程中，发现先前的学习只是触及了表面，还有大量未知领域等待探索。同时，我也认识到某些已学过的知识点并未完全掌握，无法将其灵活应用于解决现实问题中。这提示我需要更多地投入实践，持续自我学习，积极面对和解决遇到的问题，以达到更加深入理解和运用所学知识的目的。在此过程中，不仅能够加深对理论的认识，还能习得许多实用技能。

从最初的文档研究，到需求分析、概念设计、逻辑设计及物理设计等多个阶段，我全程参与了一个系统的开发流程。虽然每个步骤在教科书中都有详细的介绍，看起来似乎并不复杂，但在实际操作中却远非如此。常常是在推进项目的过程中意外发现初期设计方案存在的缺陷，不得不返回重新调整。尽管如此，经过多次迭代优化后，最终成果还算令人满意。更重要的是，我认为这次经历本身就是一次极为宝贵的学习过程，无论结果如何，能够从中获得成长就已经是非常有价值的了。