

云南大学软件学院

实验报告

课程： 数据库及安全实践 任课教师： 蔡莉 实验指导教师（签名）：

学号： 20201060448 姓名： 陈梦芸 专业： 网络空间安全

日期： 2022 年 6 月 12 日 21:05:48 成绩：

超市会员管理系统数据库设计说明书

1 引言

1.1 编写目的

本文档描述了超市会员管理系统数据库的设计，提供了数据库设计的可视性以及软件支持所需的信息，应用于超市会员管理系统开发前期，为后期的数据库设计指引方向，也可以为系统程序设计提供借鉴与参照。

预期读者：数据库设计师，数据库管理员，数据库学习者

1.2 背景

- 待开发的数据库的名称：超市会员管理系统；
- 使用此数据库的软件系统：SqlServer2019
- 任务提出者：蔡莉老师；
- 开发者：陈梦芸
- 用户：任何需要使用或学习该系统的人
- 实现该软件的计算中心：个人计算机（处理器为 AMD Ryzen 5 4600U with Radeon Graphics 2.10 GHz）
- 使用的网络：云南大学校园网

1.3 定义

- **TSQL**: T-SQL 即 Transact-SQL，是 SQL 在 Microsoft SQL Server 上的增强版，它是用来让应用程序与 SQL Server 沟通的主要语言。T-SQL 提供标准 SQL 的 DDL 和 DML 功能，加上延伸的函数、系统预存程序以及程式设计结构(例如 IF 和 WHILE)让程式设计更有弹性。
- **SQL 语言**: 结构化查询语言是一种数据库查询和程序设计语言，用于存取数据以及查询、更新和管理关系数据库系统；同时也是数据库脚本文件的扩展名。结构化查询语言是高级的非过程化编程语言。
- **SQL Server**: SQL Server 是 Microsoft 公司推出的一个高性能大型关系数据库管理系统，它为企业客户和创建商业应用程序的独立软件供应商提供了数据库管理平台，它使用方便、功能强大。使用 SQL Server 2000 可以开发不同类型的应用程序，其中包括：数据仓库、数据复制、分布式数据库应用程序、Internet 和 Intranet 应用,以及管理工具等。参考资料

- **Node:** Node.js 就是运行在服务端的 JavaScript，基于 Chrome JavaScript 运行时建立的一个平台，是一个事件驱动 I/O 服务端 JavaScript 环境，基于 Google 的 V8 引擎，V8 引擎执行 Javascript 的速度非常快，性能非常好。
- **Maven:** Maven 是基于项目对象模型(POM project object model)，可以通过一小段描述信息（配置）来管理项目的构建，报告和文档的软件项目管理工具
- **ODBC:** ODBC 即开放式数据库互连（Open Database Connectivity），是微软公司推出的一种实现应用程序和关系数据库之间通讯的接口标准。符合标准的数据库就可以通过 SQL 语言编写的命令对数据库进行操作，但只针对关系数据库。目前所有的关系数据库都符合该标准（如 SQL Server，Oracle，Access，Excel 等）。ODBC 本质上是一组数据库访问 API（应用程序编程接口），由一组函数调用组成，核心是 SQL 语句。
- **JDBC:** JDBC（Java Data Base Connectivity）是 Java Soft 公司开发的，一组 Java 语言编写的用于数据库连接和操作的类和接口，可为多种关系数据库提供统一的访问方式。通过 JDBC 完成对数据库的访问包括四个主要组件：Java 应用程序、JDBC 驱动器管理器、驱动器和数据源。

1.4 参考资料

- [1] 《数据库系统概论（第 5 版）》王珊、萨师煊著 高等教育出版社 2014 年 09 月 16 日；
- [2] 《数据库设计教程》Thomas M.Connolly、Carolyn E.Begg 著 何玉洁 / 黄婷儿译 机械工业出版社 2005 年 01 月 01 日
- [3] 《若依部署文档》RuoYi 著 来源：<http://www.twom.top/>

2 外部设计

2.1 标识符和状态

表名称	admin_id	admin_info	buy_product	charge	loss	product_type	user	vip_become	vip_card	vip_info
意义	工号表	管理员信息表	会员购买信息表	充值表	挂失表	商品类型表	角色表	会员类型表	会员卡表	会员信息表
状态	正常	正常	正常	正常	正常	正常	正常	正常	正常	正常

2.2 使用它的程序

软件	版本
docker	v20.10.14
镜像	mcr.microsoft.com/mssql/server:2022-latest
Microsoft SQL Server Management Studio	v18.11.1
RuoYi-Vue	v3.8.2

Microsoft Edge	102.0.1245.39（正式版本）64 位
JDK	v1.8.0_202
Apache Maven	v3.8.5
node	v16.15.0
超市会员管理系统	v1.0

2.3 约定

2.3.1 数据库对象命名规定：

- 表名由英文单词组合而成，每单词中间需要添加下划线
- 表和字段都需要添加中文注释
- 外键名必须以 FK_ 开头
- 视图名称使用中文，且每个字段都需要取中文别名
- 存储过程不需要下划线，具有什么样的功能、处理哪个表、使用什么字段，需要写明，例如：deladminbyid，表名该过程通过工号 id 删除已有工号
- 触发器需要写明表名，并以 tri 结尾

2.4 专门指导

向准备从事此数据库的生成、从事此数据库的测试、维护人员提供专门的指导，例如将被送入数据库的数据的格式和标准、送入数据库的操作规程和步骤，用于产生、修改、更新或使用这些数据文卷的操作指导。

可参考资料：

- [1]《数据库系统概论（第 5 版）》王珊、萨师煊著 高等教育出版社 2014 年 09 月 16 日；
- [2] 《数据库设计教程》Thomas M.Connolly、Carolyn E.Begg 著 何玉洁 / 黄婷儿译 机械工业出版社 2005 年 01 月 01 日
- [3] 《若依部署文档》RuoYi 著 来源：<http://www.twom.top/>
- [4] 云南大学 学堂在线视频课——数据库概论

2.5 支持软件

软件名称	版本	功能	技术文书
Microsoft SQL Server Management Studio	v18.11.1	管理员工具，可用于管理数据库	[1]微软官方技术文档
Navicat Premium	v16.0.13	管理员工具，可用于管理数据库	[2]navica 官方中文文档
RuoYi-Vue	v3.8.2	web 页面，便于管理员和客户使用	[3]ruoyi 官方文档 [4]csdn 博客 [5]csdn 博客

[1]<https://docs.microsoft.com/zh-cn/sql/sql-server/editions-and-components-of-sql-server-2019?view=sql-server-ver15>

[2]https://www.navicat.com.cn/manual/online_manual/cn/navicat_16/win_manual/#/main_window_0

[3]<http://doc.ruoyi.vip/ruoyi/document/kslj.html#%E9%A1%B9%E7%9B%AE%E7%AE%80%E4%BB%8B>

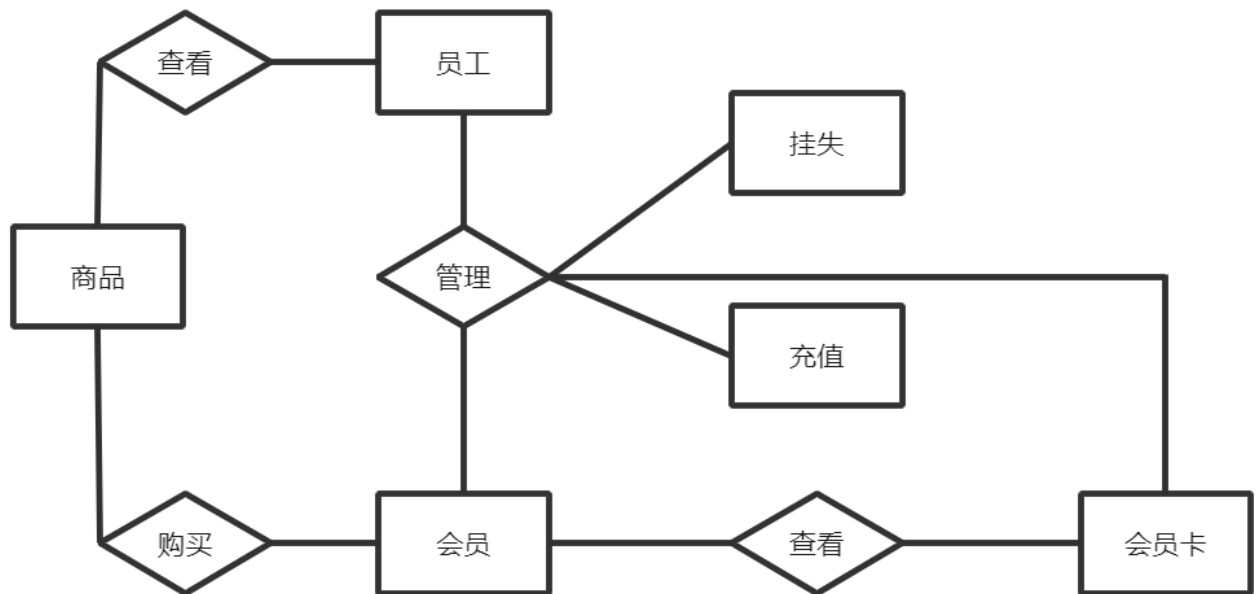
[4] [RuoYi 若依打包发布与部署](#) 作者: ITKEY_

[5] [\[精简\]RuoYi 开发实战-搭建开发环境](#) 作者: ITKEY_

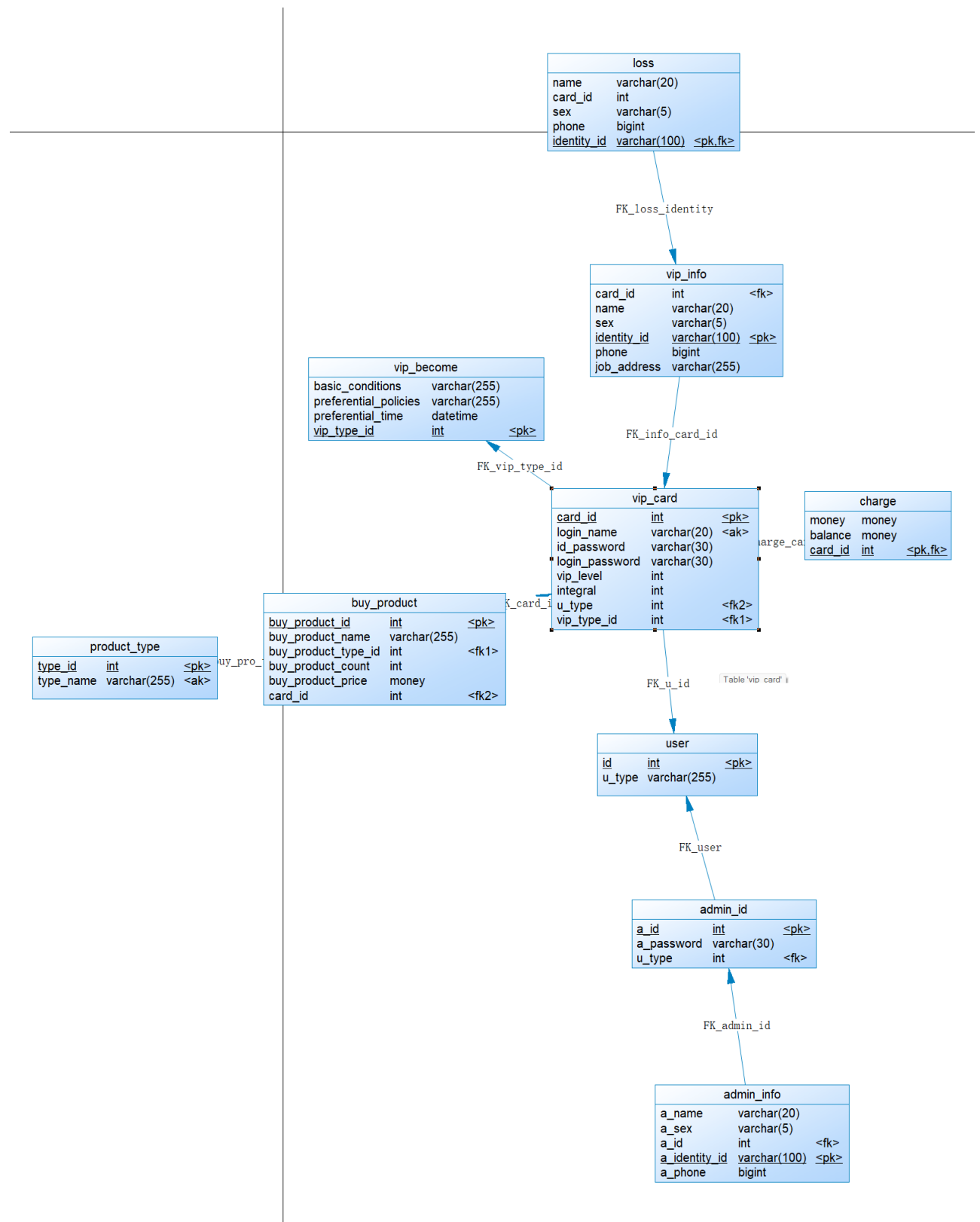
3 结构设计

3.1 概念结构设计

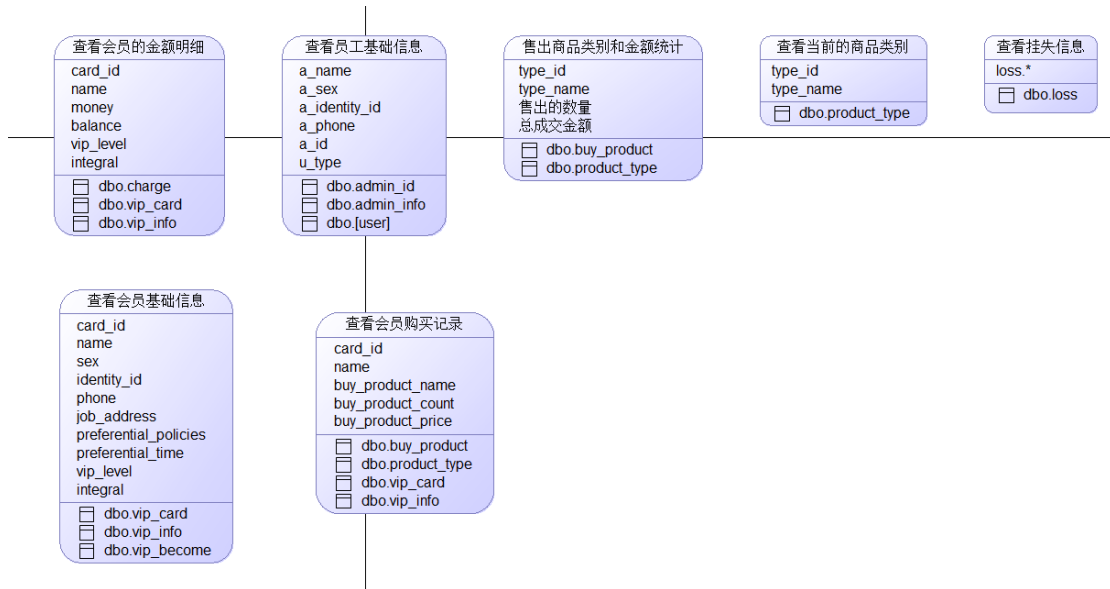
3.1.1 ER 图



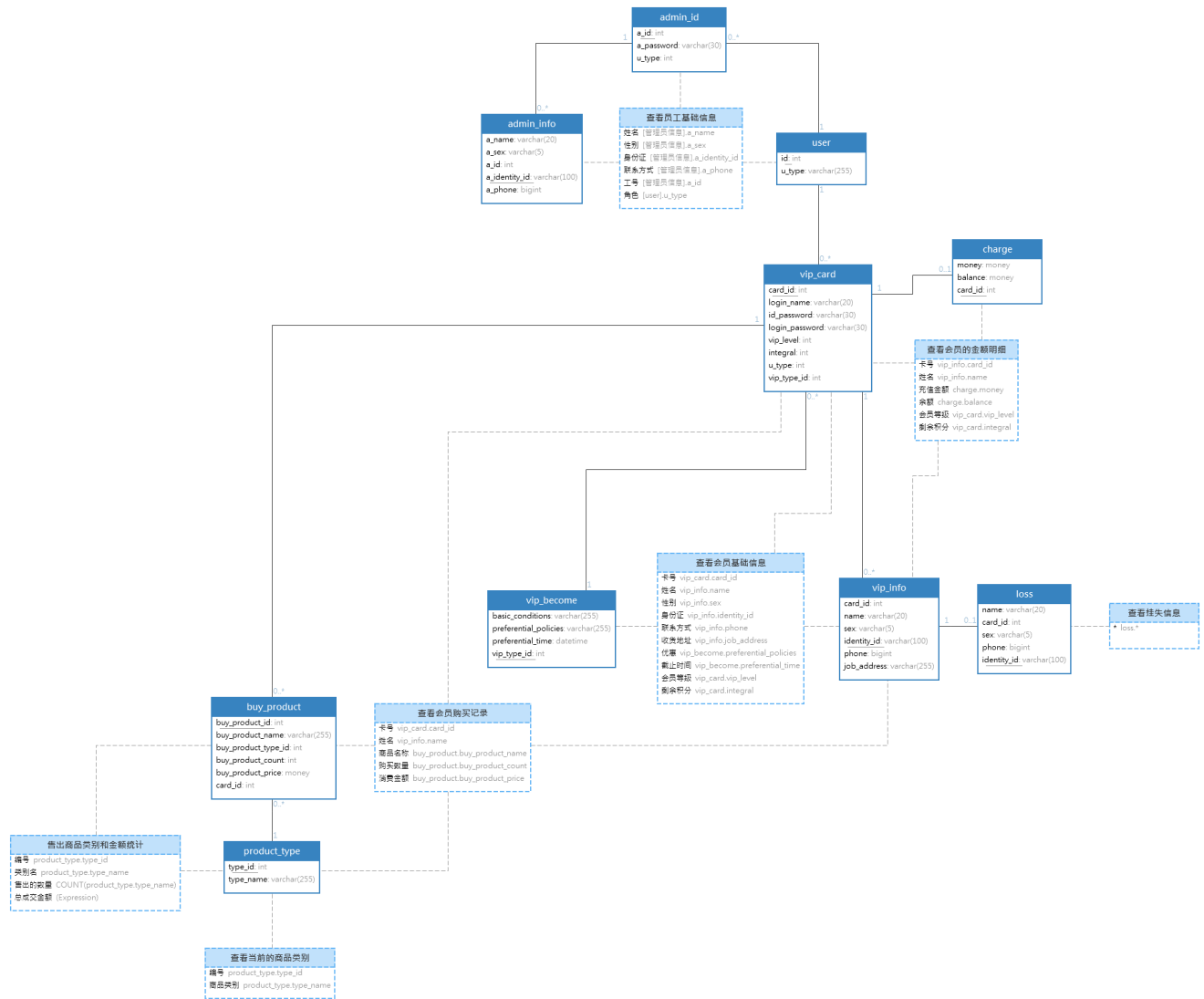
3.1.2 表模型



3.1.3 视图模型



3.1.4 整体结构



3.2 逻辑结构设计

- admin_id(a_id,a_password,u_type)
- admin_info(a_name,a_sex,a_id,a_identity_id,a_phone)
- buy_product(buy_product_id,buy_product_name, buy_product_type_id, buy_product_count, buy_product_price,card_id)
- charge(money,balance,card_id)
- loss(name,card_id,sex,phone,identity_id)
- product_type(type_id,type_name)
- user(id,u_type)
- vip_become(basic_conditions,preferential_policies,preferential_time, vip_type_id)
- vip_card(card_id,login_name,id_password,login_password,vip_level _integral,u_type,vip_type_id)
- vip_info(card_id,name,sex,identity_id,phone,job_address)

3.3 物理结构设计

3.3.1 服务器选择与配置

操作系统	Ubuntu 20.04
平台	Linux
服务器内存	0MB - 2147483647MB
处理器	12 个，自动分配
是否群集化	否
是否启用 HADR	否
是否支持 XTP	是
数据、日志、备份位置	/var/opt/mssql/data

3.3.2 访问数据的方式方法

- ODBC 数据库接口
- JDBC 数据库接口

3.3.3 存储信息

- 外存设备：硬盘
- 文件组
 - 名称：PRIMARY
 - 位置：/var/opt/mssql/data
 - 增量为 64 MB，增长无限制
- 各表主键使用聚簇索引，均存放与文件组 PRIMARY
- 未对数据进行分区
- 未对数据进行压缩

4 运用设计

4.1 数据字典设计

4.1.1 admin_id 工号表

数据项	类型	描述
a_id	int	工号
a_password	varchar(30)	登陆密码
u_type	int	身份类型

通过工号连接员工，账号为工号，登陆密码为登陆系统的个人密码，身份类型占比为 管理员:超级管理员=8:2。

4.1.2 admin_info 员工个人信息表

数据项	类型	描述
a_name	varchar(20)	管理员姓名
a_sex	varchar(5)	管理员性别
a_id	int	工号
a_identity_id	varchar(100)	管理员身份证号
a_phone	bigint	管理员联系方式

通过工号连接员工，以个人身份证号为主键，及个人标识，管理员可自行登陆修改联系方式，超级管理员可以修改普通管理员的信息。

4.1.3 buy_product 会员购买商品记录表

数据项	类型	描述
buy_product_id	int	物品编号
buy_product_name	varchar(255)	物品名称
buy_product_type_id	int	所属种类
buy_product_count	int	购买数量
buy_product_price	money	总价
card_id	int	会员卡号

通过会员卡号连接会员，记录会员购买的商品信息。

通过商品种类 ID 连接商品类型表，可用于分析商品的出售信息，调整进货策略。

4.1.4 charge 充值表

数据项	类型	描述
money	money	充值金额
balance	money	余额
card_id	int	卡号

通过会员卡号连接会员，记录会员累计充值金额以及当前余额，可以通过存储过程修改 money，借此实现充值。

4.1.5 loss 挂失表

数据项	类型	描述
name	varchar(20)	姓名
card_id	int	会员卡号
sex	varchar(5)	性别
phone	bigint	联系方式
identity_id	varchar(100)	身份证

会员告知管理员自己的姓名和身份证，经查询，如若匹配，则可以插入挂失表，并显示匹配的卡号。

4.1.6 product_type 商品类型表

数据项	类型	描述
type_id	int	商品类型编号
type_name	varchar(255)	商品类型

用于统计商品信息

4.1.7 user 身份类型表

数据项	类型	描述
id	int	身份编号
u_type	varchar(255)	身份类型

身份共三种：会员、管理员、超级管理员

4.1.8 vip_become 会员类型表

数据项	类型	描述
basic_conditions	varchar(255)	成为 vip 的基本条件
preferential_policies	varchar(255)	优惠政策
preferential_time	datetime	优惠时长
vip_type_id	int	优惠类型

会员类型，记录了每个会员加入会员时不同的活动

4.1.9 vip_card 会员卡表

数据项	类型	描述
card_id	int	会员卡号
login_name	varchar(20)	登录名
id_password	varchar(30)	卡密码
login_password	varchar(30)	登陆密码
vip_level	int	会员等级
integral	int	会员积分
u_type	int	身份类型
vip_type_id	int	优惠类型

会员卡表，主键 card_id，登陆名和卡号均可以登陆系统，设置登录名是为了账户好记，会员等级和积分会根据购买的商品和充值金额升级，身份类型均为会员，优惠类型记录了会员加入会员时不同的活动。

4.1.10 vip_info 会员个人信息表

数据项	类型	描述
card_id	int	会员卡号
name	varchar(20)	姓名
sex	varchar(5)	性别
identity_id	varchar(100)	身份证
phone	bigint	联系方式
job_address	varchar(255)	收货地址

会员个人信息表，主键为身份证，外键为会员卡号，联系方式和收货地址可由会员自由更改

4.2 安全保密设计

4.2.1 登录名

设置了三级登陆名：vip、vipadmin、super_admin，分别对应会员、普通管理员和超级管理员。

4.2.2 角色

新建了三个数据库角色：vip、vipadmin、super_admin，分别赋予不同的权限

角色\模块	员工管理	会员卡管理	商品信息管理	会员信息管理
vip	×	×	×	√
vipadmin	×	√	√	√
super_admin	√	√	√	√

4.2.3 用户

新建了三个用户：vip01、vipadmin、super_admin，属于对应的角色，赋予对应的权限

5 具体应用

5.1 查询

5.1.1 admin_id

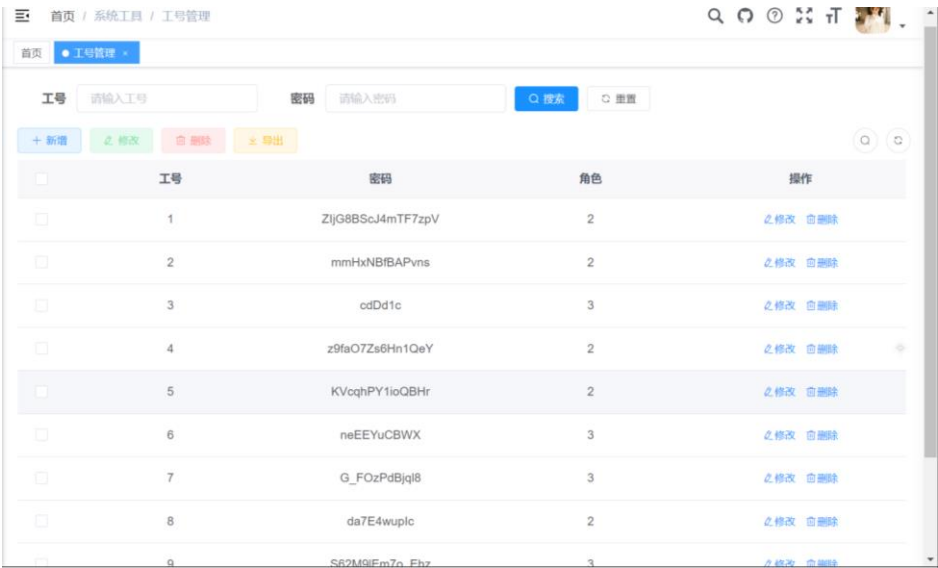
5.1.1.1简单查询语句

SELECT * FROM admin_id;

5.1.1.2查询结果

a_id	a_password	u_type
1	ZljG8BScJ4mTF7zpV	2
2	mmHxNBfBAPvns	2
3	cdDd1c	3
4	z9faO7Zs6Hn1QeY	2
5	KVcqhpY1ioQBHr	2
6	neEEYuCBWX	3
7	G_FOzPdBjqI8	3
8	da7E4wuplc	2
9	S62M9IEm7o_Ehz	3
10	GUau9gaLM	2

5.1.1.3前端界面



5.1.2 admin_info

5.1.2.1简单查询语句

SELECT * FROM admin_info;

5.1.2.2查询结果

a_name	a_sex	a_id	a_identity_id	a_phone
石宇宁	M	1	125514105979656	7698107232
石安琪	F	4	165217031572010378	7694571853
张岚	F	2	167210620787012568	16960584804
严嘉伦	M	8	271035348632674278	2841425206
杜云熙	M	5	282343686889712194	13525976957
尹宇宁	M	7	28870424128079097X	13275984367
杜秀英	F	9	501577821873902114	17597984853
孔子韬	M	3	587465575697666	15488416873
袁云熙	M	6	638701127119678	18524580046
方杰宏	M	10	777379742810635262	1010883805

5.1.2.3前端界面

首页 / 系统工具 / 员工信息管理

首页 代码生成 工号管理 员工信息管理

姓名 请输入姓名

工号 请输入工号

身份证 请输入身份证

联系方式 请输入联系方式

搜索 重置

+ 新增

修改

删除

导出

	姓名	性别	工号	身份证	联系方式	操作
<input type="checkbox"/>	石宇宁	M	1	125514105979656	7698107232	修改 删除
<input type="checkbox"/>	石安琪	F	4	165217031572010378	7694571853	修改 删除
<input type="checkbox"/>	张岚	F	2	167210620787012568	16960584804	修改 删除
<input type="checkbox"/>	严嘉伦	M	8	271035348632674278	2841425206	修改 删除
<input type="checkbox"/>	杜云熙	M	5	282343686889712194	13525976957	修改 删除
<input type="checkbox"/>	尹宇宁	M	7	28870424128079097X	13275984367	修改 删除
<input type="checkbox"/>	杜秀英	F	9	501577821873902114	17597984853	修改 删除

5.1.3 buy_product

5.1.3.1简单查询语句

SELECT * FROM buy_product;

5.1.3.2查询结果

buy_product_id	buy_product_name	buy_product_type_id	buy_product_count	buy_product_price	card_id
1	trape	1	192	963.7500	854
2	Apgle se	1	116	296.4600	711
3	Cherry core	2	988	556.4800	883
4	Kiji	6	76	454.5100	871
5	Kiwi air	3	683	258.5100	881
6	Cherky	8	66	26.8800	424
7	Raspberry pro	8	120	387.5900	299
8	Raspberry	7	74	512.1700	465
9	tiwi pi	10	111	514.3400	362
10	omni-wango	10	620	527.3500	628

5.1.3.3前端界面

充值金额
余额
卡号

<input type="checkbox"/>	充值金额	余额	卡号	操作
<input type="checkbox"/>	614.4712	333.9954	2	✎ 修改 🗑 删除
<input type="checkbox"/>	691.496	793.0461	3	✎ 修改 🗑 删除
<input type="checkbox"/>	470.3827	282.303	4	✎ 修改 🗑 删除
<input type="checkbox"/>	209.0319	314.5909	5	✎ 修改 🗑 删除
<input type="checkbox"/>	950.3222	200.3381	6	✎ 修改 🗑 删除
<input type="checkbox"/>	82.575	90.8423	7	✎ 修改 🗑 删除
<input type="checkbox"/>	667.5654	381.2397	8	✎ 修改 🗑 删除

5.1.5 loss

5.1.5.1简单查询语句

SELECT * FROM loss;

5.1.5.2查询结果

name	card_id	sex	phone	identity_id
林子韬	373	M	208092245	001095013309177130
程杰宏	364	M	17699895991	002144596457607638
江詩涵	969	F	204061635	003861179583969059
李云熙	157	M	109400694	00435679271173761x
傅子韬	140	M	17613341870	006502179677606087
阎宇宁	387	M	14298554210	007070734667064
郝詩涵	947	F	13594300899	007973135760929980
譚致远	442	M	17263933316	008207289624035033
陶震南	492	M	14212498759	008411010313969545
常睿	410	M	2126051867	008414321388808

5.1.5.3前端界面

姓名

卡号

联系方式

身份证

搜索

重置

+ 新增

修改

删除

导出

Q

Q

<input type="checkbox"/>	姓名	卡号	性别	联系方式	身份证	操作
<input type="checkbox"/>	林子韬	373	M	208092245	001095013309177130	修改 删除
<input type="checkbox"/>	程杰宏	364	M	17699895991	002144596457607638	修改 删除
<input type="checkbox"/>	江詩涵	969	F	204061635	003861179583969059	修改 删除
<input type="checkbox"/>	李云熙	157	M	109400694	00435679271173761x	修改 删除
<input type="checkbox"/>	傅子韬	140	M	17613341870	006502179677606087	修改 删除
<input type="checkbox"/>	阎宇宁	387	M	14298554210	007070734667064	修改 删除
<input type="checkbox"/>	郝詩涵	947	F	13594300899	007973135760929980	修改 删除

5.1.6 product_type

5.1.6.1简单查询语句

```
SELECT      * FROM product_type;
```

5.1.6.2查询结果

type_id	type_name
3	电子游戏
1	服装，鞋子和珠宝
6	工具与家居装饰
8	花园与户外
2	家居及厨房用具
7	乐器用品
4	其他
9	书本
5	玩具与游戏
10	优质美容

5.1.6.3前端界面

编号

请输入编号

名称

请输入名称

Q 搜索

↺ 重置

+ 新增

✎ 修改

🗑 删除

📄 导出

Q

↺

<input type="checkbox"/>	编号	名称	操作
<input type="checkbox"/>	3	电子游戏	✎ 修改 🗑 删除
<input type="checkbox"/>	1	服装, 鞋子和珠宝	✎ 修改 🗑 删除
<input type="checkbox"/>	6	工具与家居装饰	✎ 修改 🗑 删除
<input type="checkbox"/>	8	花园与户外	✎ 修改 🗑 删除
<input type="checkbox"/>	2	家居及厨房用具	✎ 修改 🗑 删除
<input type="checkbox"/>	7	乐器用品	✎ 修改 🗑 删除
<input type="checkbox"/>	4	其他	✎ 修改 🗑 删除
<input type="checkbox"/>	9	书本	✎ 修改 🗑 删除

5.1.7 user

5.1.7.1 简单查询语句

```
SELECT * FROM [user];
```

5.1.7.2 查询结果

id	u_type
1	会员
2	管理员
3	超级管理员

5.1.7.3 前端界面

系统工具 / 身份管理

身份管理

搜索

重置

新增

修改

删除

导出

<input type="checkbox"/>	编号	身份	操作
<input type="checkbox"/>	1	会员	修改 删除
<input type="checkbox"/>	2	管理员	修改 删除
<input type="checkbox"/>	3	超级管理员	修改 删除

5.1.8 vip_become

5.1.8.1 简单查询语句

```
SELECT * FROM vip_become;
```

5.1.8.2 查询结果

basic_conditions	preferential_policies	preferential_time	vip_type_id
▶ 购买200元VIP卡	充2000赠1000，购物九五折	2006-10-29 00:00:00.000	1
购买200元VIP卡	购物九折，赠送购物袋	2021-04-26 00:00:00.000	2
购买200元VIP卡	充2000赠1000，购物九五折	2008-07-03 00:00:00.000	3
购买200元VIP卡	满200元赠送小礼品	2022-01-23 00:00:00.000	4
充值2000元	购物八五折	2020-12-28 00:00:00.000	5
累计消费5000元	购物八五折	2011-03-02 00:00:00.000	6
购买200元VIP卡	购物九折，赠送购物袋	2007-05-14 00:00:00.000	7
购买200元VIP卡	充2000赠1000，购物九五折	2011-08-29 00:00:00.000	8
购买200元VIP卡	充2000赠1000，购物九五折	2012-05-16 00:00:00.000	9
累计消费5000元	满200元赠送小礼品	2008-01-16 00:00:00.000	10
购买200元VIP卡	满200元赠送小礼品	2016-02-15 00:00:00.000	11

5.1.8.3前端界面

首页

菜单管理 ×

身份管理 ×

● 会员活动管理 ×

条件

请输入条件

优惠

请输入优惠

时间

请输入时间

编号

请输入编号

搜索

重置

+ 新增

修改

删除

导出

Q

Q

<input type="checkbox"/>	条件	优惠	时间	编号	操作
<input type="checkbox"/>	购买200元VIP卡	充2000赠1000，购物九五折	29/10/2006 00:00:00	1	修改 删除
<input type="checkbox"/>	购买200元VIP卡	购物九折，赠送购物袋	26/4/2021 00:00:00	2	修改 删除
<input type="checkbox"/>	购买200元VIP卡	充2000赠1000，购物九五折	3/7/2008 00:00:00	3	修改 删除
<input type="checkbox"/>	购买200元VIP卡	满200元赠送小礼品	23/1/2022 00:00:00	4	修改 删除
<input type="checkbox"/>	充值2000元	购物八五折	28/12/2020 00:00:00	5	修改 删除
<input type="checkbox"/>	累计消费5000元	购物八五折	2/3/2011 00:00:00	6	修改 删除
<input type="checkbox"/>	购买200元VIP卡	购物九折，赠送购物袋	14/5/2007 00:00:00	7	修改 删除

5.1.9 vip_card

5.1.9.1简单查询语句

SELECT * FROM vip_card;

5.1.9.2查询结果

card_id	login_name	id_password	login_password	vip_level	integral	u_type	vip_type_id
▶ 2	kukaryan	lIttMZHVtktxfQE	kn5mf3wUYutms0oTz	806	694	3	16
3	kotw7	Qa5UqO21M72	OtuakaA4rAif0WDw	935	543	3	18
4	evajuan2	dugsuBp_yjOHP1z	OE1jiV	62	917	2	11
5	fujiehong	KCa4Dum	ubDPHxzcBb6BaUX3	762	774	2	19
6	chhughes5	AxXyqV3Cb_	cnoGoBF42riLYkchIY	760	590	2	17
7	albertparke107	sBUK5F6lmxj	ITtTBBDvLoj8Jkl2	421	55	1	20
8	miuogawa	VJ8pfAN7H9Vy1GkT	PLnO0_jVg8LFeq1	563	928	1	1
9	stonejamie1002	Y6WjYz6betdieEv	yEAsSWdWUrFEh	776	263	1	1
10	maruyamaikki117	E7qR1OAK	eqRoX7p_y4vA	237	139	3	2
11	mainakan6	LZdFZkRXWnT	zjF58yPTFPTh	785	736	2	8
12	yha3	qt1iPO41jtelu6b3Qw	hZM5bjH0p2	644	90	3	10

5.1.9.3前端界面

登录密码

请输入登录密码

等级

请输入等级

积分

请输入积分

请输入\${comment}

搜索

重置

+ 新增

修改

删除

导出

Q

<input type="checkbox"/>	卡号	登录名	卡密码	登录密码	等级	积分	\${comment}	\${comment}	操作
<input type="checkbox"/>	2	kukaryan	lttMZHvTktfQE	kn5mf3wUYutms0oTz	806	694	3	16	修改 删除
<input type="checkbox"/>	3	kotw7	Qa5UqO21M72	OtuakaA4rAif0WDw	935	543	3	18	修改 删除
<input type="checkbox"/>	4	evajuan2	dugsuBp_yjOHP1z	OE1jIV	62	917	2	11	修改 删除
<input type="checkbox"/>	5	fujiehong	KCa4Dum	ubDPhxzcBb6BaUX3	762	774	2	19	修改 删除
<input type="checkbox"/>	6	chhughes5	AxXyqV3Cb_	cnoGoBF42riLYkchIY	760	590	2	17	修改 删除
<input type="checkbox"/>	7	albertparke107	sBUK5F6lmxj	ITtTBBDvLoj8Jkl2	421	55	1	20	修改 删除
<input type="checkbox"/>	8	miuogawa	VJ8pfAN7H9Vv1GkT	PLnO0_jVg8lFae1	563	928	1	1	修改 删除

5.1.10 vip_info

5.1.10.1 简单查询语句

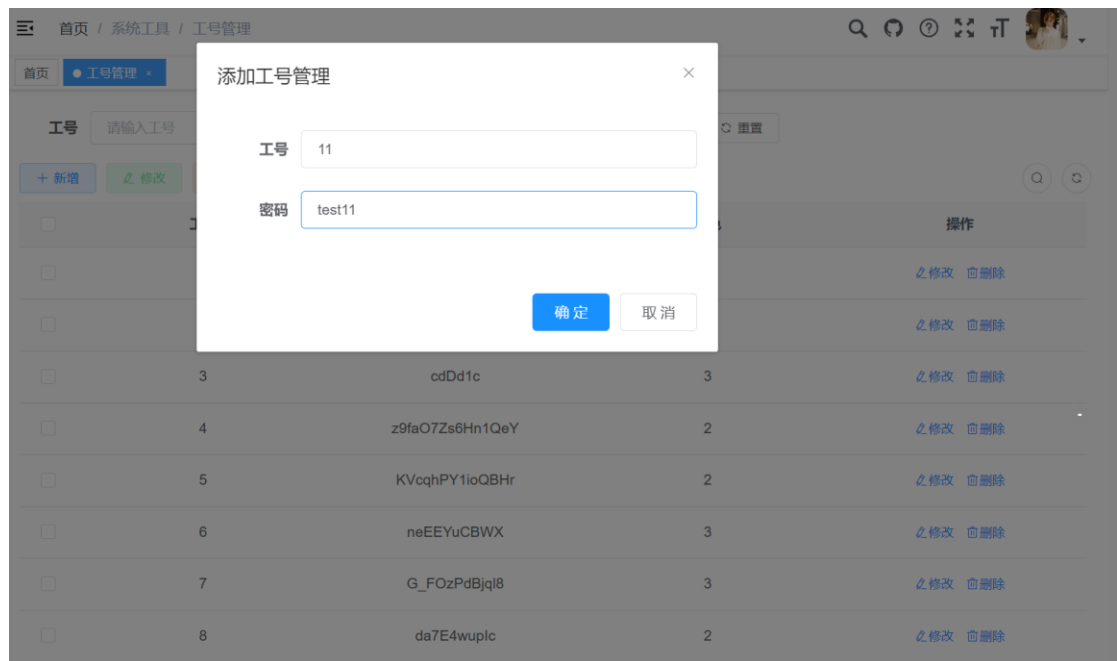
SELECT * FROM vip_info;

5.1.10.2 查询结果

card_id	name	sex	identity_id	phone	job_address
373	林子韬	M	001095013309177130	208092245	越秀区中山二路192号
364	程杰宏	M	002144596457607638	17699895991	坑美十五巷490号
969	江詩涵	F	003861179583969059	204061635	天河区天河路469号
157	李云熙	M	00435679271173761x	109400694	东城区东单王府井东街467号
140	傅子韬	M	006502179677606087	17613341870	罗湖区蔡屋围深南东路550号
387	阎宇宁	M	007070734667064	14298554210	珊瑚路702号
947	郝詩涵	F	007973135760929980	13594300899	越秀区中山二路338号
442	谭致远	M	008207289624035033	17263933316	福田区景田东一街128号
492	陶震南	M	008411010313969545	14212498759	坑美十五巷484号
410	常睿	M	008414321388808	2126051867	闵行区宾川路202号
932	范杰宏	M	008693627402421	19530232238	天河区大信商圈大信南路36号

5.1.10.3 前端界面

5.2.1.3前端界面



5.2.2 admin_info

5.2.2.1SQL 语句

```
SET IDENTITY_INSERT [admin_info] ON;  
INSERT admin_info ( admin_info.a_name, admin_info.a_sex,  
admin_info.a_id, admin_info.a_identity_id, admin_info.a_phone )  
VALUES  
( 'test', 'F', 11, 'test11', 123 );
```

5.2.2.2结果

信息	摘要	结果 1 (2)			
a_name	a_sex	a_id	a_identity_id	a_phone	
▶ 石宇宁	M		1 125514105979656	7698107232	
石安琪	F		4 165217031572010378	7694571853	
张岚	F		2 167210620787012568	16960584804	
严嘉伦	M		8 271035348632674278	2841425206	
杜云熙	M		5 282343686889712194	13525976957	
尹宇宁	M		7 28870424128079097X	13275984367	
杜秀英	F		9 501577821873902114	17597984853	
孔子韬	M		3 587465575697666	15488416873	
袁云熙	M		6 638701127119678	18524580046	
方杰宏	M		10 777379742810635262	1010883805	
test	F		11 test11	123	

5.2.2.3前端界面

首页 / 系统工具 / 员工信息管理

姓名 请输入姓名

联系方式 请输入联系方式

+ 新增

修改

姓名

石宇宁

石安琪

张岚

严嘉伦

杜云熙

尹宇宁

杜秀英

性别

M

F

F

M

M

M

F

工号

4

2

8

5

7

9

身份证

165217031572010378

167210620787012568

271035348632674278

282343686889712194

28870424128079097X

501577821873902114

联系方式

7698107232

7694571853

16960584804

2841425206

13525976957

13275984367

17597984853

操作

修改 删除

修改 删除

修改 删除

修改 删除

修改 删除

修改 删除

修改 删除

添加员工信息管理

姓名 test

工号 12

身份证 test12

联系方式 123

确定

取消

首页 / 系统工具 / 员工信息管理

姓名 请输入姓名

工号 请输入工号

身份证 请输入身份证

联系方式 请输入联系方式

搜索

重置

+ 新增

修改

删除

导出

姓名

test

test

性别

F

F

工号

11

12

身份证

test11

test12

联系方式

123

123

操作

修改 删除

修改 删除

共 12 条

10条/页

<

1

2

>

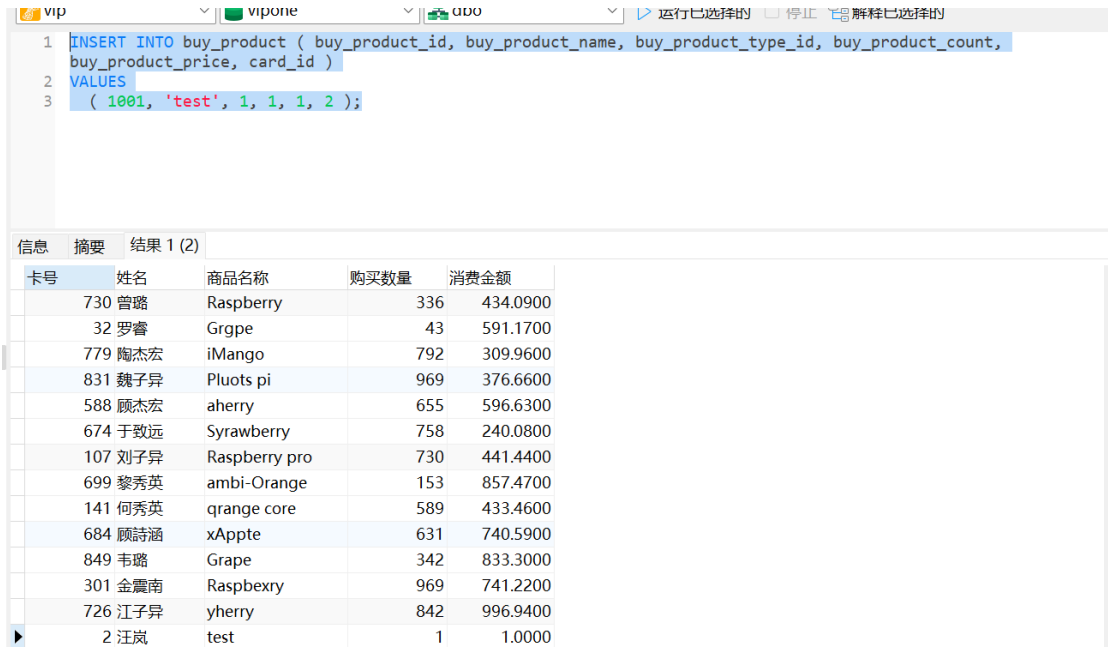
前往 2 页

5.2.3 buy_product

5.2.3.1SQL 语句

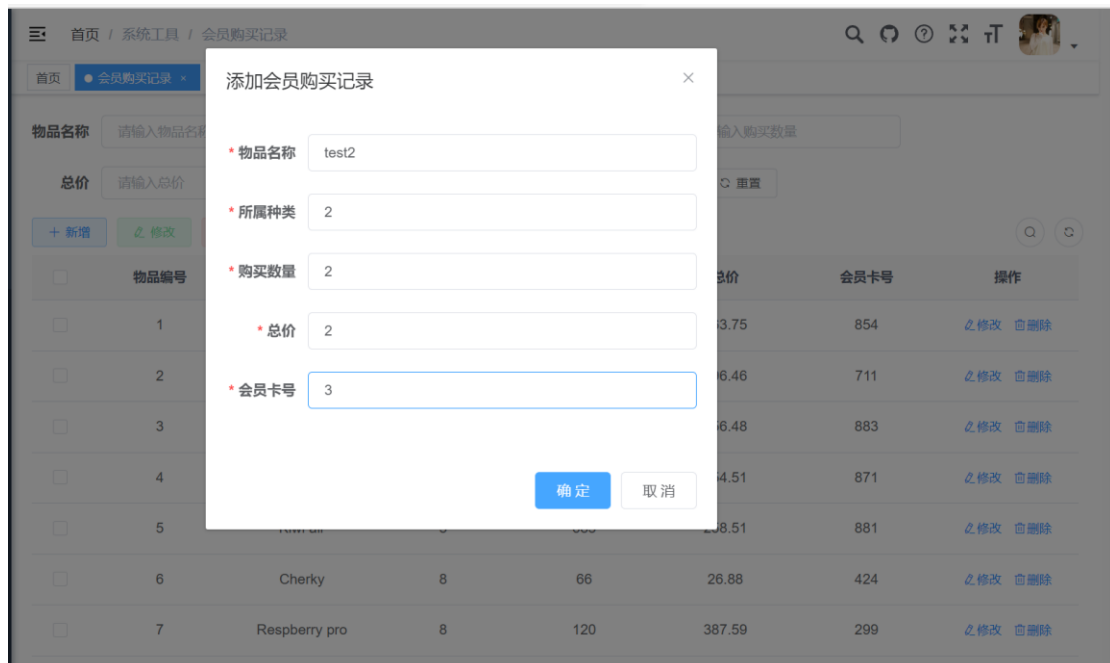
```
INSERT INTO buy_product ( buy_product_id, buy_product_name,  
buy_product_type_id, buy_product_count, buy_product_price, card_id )  
VALUES  
( 1001, 'test', 1, 1, 1, 2 );
```

5.2.3.2结果



卡号	姓名	商品名称	购买数量	消费金额
730	曾璐	Raspberry	336	434.0900
32	罗睿	Grgpe	43	591.1700
779	陶杰宏	iMango	792	309.9600
831	魏子异	Pluots pi	969	376.6600
588	顾杰宏	aherry	655	596.6300
674	于致远	Syrawberry	758	240.0800
107	刘子异	Raspberry pro	730	441.4400
699	黎秀英	ambi-Orange	153	857.4700
141	何秀英	qrange core	589	433.4600
684	顾诗涵	xAppte	631	740.5900
849	韦璐	Grape	342	833.3000
301	金震南	Raspbexry	969	741.2200
726	江子异	yherry	842	996.9400
2	汪岚	test	1	1.0000

5.2.3.3前端界面



添加会员购买记录

* 物品名称: test2

* 所属种类: 2

* 购买数量: 2

* 总价: 2

* 会员卡号: 3

确定 取消

物品名称	物品编号	购买数量	消费金额	会员卡号	操作
	1		3.75	854	修改 删除
	2		6.46	711	修改 删除
	3		6.48	883	修改 删除
	4		4.51	871	修改 删除
	5		8.51	881	修改 删除
Cherky	6	8	26.88	424	修改 删除
Raspberry pro	7	8	387.59	299	修改 删除

5.2.4 charge

5.2.4.1SQL 语句

```
INSERT INTO charge ( charge.money, charge.balance,charge.card_id )
```

VALUES

(1001,1001,1000);

报错：Violation of PRIMARY KEY constraint 'charge_pk'. Cannot insert duplicate key in object 'dbo.charge'. The duplicate key value is (1000).

原因：因为设置了会员卡表的外键，且 charge 表的 id 主键不得重复，所以需要先在会员卡表中插入新的卡号再插入该表。

正确语句：

```
INSERT INTO vip_card ( card_id, login_name, id_password, login_password, vip_level, integral, vip_type_id )
```

VALUES

(1001, 'test', 'test', 'test',1001,1001,1);

```
INSERT INTO charge ( charge.money, charge.balance, charge.card_id )
```

VALUES

(1001, 1001, 1001);

```
3 ( 1001, 'test', 1, 1, 1, 2 );
4 INSERT INTO vip_card ( card_id, login_name, id_password, login_password, vip_level, integral, vip_type_id )
5 VALUES
6 ( 1001, 'test', 'test', 'test',1001,1001,1);
7 INSERT INTO charge ( charge.money, charge.balance, charge.card_id )
8 VALUES
9 ( 1001, 1001, 1001 );
```

卡号	姓名	性别	身份证	联系方式	收货地址	优惠	截止时间	会员等级	剩余积分
(N/A)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)	(N/A)

这时会员信息还没插入，而触发器内的 select 是使用会员卡和会员信息两表 inner join 得到的，而此时还没插入 vip_info，因此显示的是 null

5.2.4.2结果

card_id	login_name	id_password	login_password	vip_level	integral	u_type	vip_type_id
1001	test	test	test	1001	1001	1	1
1000	jacksonhowar3	imnVNe3yfqvFr	jkoV9P0DEjapu	658	375	1	11
000	liang	CSsziMwE	MukhliEa4V	270	001	2	2

buy_product_id	buy_product_name	buy_product_type_id	buy_product_count	buy_product_price	card_id
1001	test	1	1	1.0000	2
1000	uberm	0	842	006 0400	726

5.2.4.3前端界面

+ 新增 修改 删除 导出

充值金额	余额	卡号	操作
1001	1001	1001	修改 删除

5.2.5 loss

5.2.5.1 SQL 语句

插入时应检测卡号和身份证是否对应的上:

- 错误插入

```
INSERT INTO loss ( loss.name, loss.card_id, loss.sex, loss.phone,
loss.identity_id )
VALUES
( 'test', 2, 'F', 123, 'test' );
```

```
> [23000] [Microsoft][ODBC Driver 17 for SQL Server][SQL Server]The INSERT
statement conflicted with the FOREIGN KEY constraint "FK_loss_identity".
The conflict occurred in database "vipone", table "dbo.vip_info", column
'identity_id'. (547)
[01000] [Microsoft][ODBC Driver 17 for SQL Server][SQL Server]The statement
has been terminated. (3621)
```

- 正确插入

```
INSERT INTO loss ( loss.name, loss.card_id, loss.sex, loss.phone,
loss.identity_id )
VALUES
( 'test', 2, 'F', 123, '777331534358445' );
```

> 挂失成功

> Affected rows: 0

Affected rows: 1

> 查询时间: 0.018s

- 使用存储过程插入 (推荐: 检测人名和身份证号, 因为一般人都记不住卡号)

- EXEC insertloss 'test', '777331534358445';

```
> Msg 515, Level 16, State 2, Server sql1, Procedure insertloss, Line
11
```

```
Cannot insert the value NULL into column 'name', table
'vipone.dbo.loss'; column does not allow nulls. INSERT fails.
```

```
> The statement has been terminated.
```

```
> [23000] [Microsoft][ODBC Driver 17 for SQL Server][SQL
Server]Cannot insert the value NULL into column 'name', table
'vipone.dbo.loss'; column does not allow nulls. INSERT fails. (515)
```

```
[01000] [Microsoft][ODBC Driver 17 for SQL Server][SQL Server]The
statement has been terminated. (3621)
```

- EXEC insertloss '张云熙', '204202113497239';

```
> 挂失成功
```

```
> OK
```

```
> 查询时间: 0.013s
```

5.2.5.2结果

开始事务 文本 筛选 排序 导入 导出 数据生成 创建图表					
name	card_id	sex	phone	identity_id	
test		2 F		123 777331534358	
会员涵	16 E		75562645707	016201467677	

5.2.5.3前端界面

首页 丢失

姓名 请输入姓名

会员卡号 请输入会员卡号

联系方式 请输入联系方式

搜索

重置

+ 新增

修改

删除

导出

姓名

会员卡号

性别

联系方式

身份证

操作

林子韬

373

M

208092245

001095013309177130

修改

删除

程杰宏

364

M

17699895991

002144596457607638

修改

删除

江詩涵

969

F

204061635

003861179583969059

修改

删除

5.2.6 product_type

5.2.6.1SQL 语句

```
INSERT INTO product_type(product_type.type_id,type_name) VALUES (11,'test');
```

5.2.6.2结果

信息	摘要	结果 1 (2)
编号	商品类别	
▶	11	test
	3	电子游戏
	1	服装, 鞋子和珠宝
	6	工具与家居装饰
	8	花园与户外
	2	家居及厨房用具
	7	乐器用品
	4	其他
	9	书本
	5	玩具与游戏
	10	优质美容

5.2.6.3前端界面

商品类型

请输入商品类型

搜索 重置

+ 新增 修改 删除 导出

商品类型编号	商品类型	操作
11	test	修改 删除

5.2.7 myuser

5.2.7.1 SQL 语句

```
INSERT INTO myuser (myuser.id,myuser.u_type) VALUES (4,'test');
```

5.2.7.2 结果

> Affected rows: 1

> 查询时间: 0.007s

表	视图	函数	用户
对象	product_type @...		
开始事务	文本	筛选	
id	u_type		
1	会员		
2	管理员		
3	超级管理员		
4	test		

5.2.7.3 前端界面

首页 商品类型 x 登录身份: 会员、管理员、超级管理员 x

搜索 重置

+ 新增 修改 删除 导出

身份编号	身份类型	操作
1	会员	修改 删除
2	管理员	修改 删除
3	超级管理员	修改 删除
4	test	修改 删除

共 4 条 10条/页 < 1 > 前往 1 页

5.2.8 vip_become

5.2.8.1 SQL 语句

```
INSERT INTO
vip_become(vip_become.basic_conditions,vip_become.preferential_policies
,vip_become.preferential_time,vip_become.vip_type_id)
VALUES('test','test',2022-01-22,21);
```

5.2.8.2 结果

> 会员类别更新成功

> Affected rows: 0

Affected rows: 1

> 查询时间: 0.011s

购买200元VIP卡	满200元赠送小礼品	2007-05-08 00:00:00.000	17
充值2000元	购物九折, 赠送购物袋	2005-09-14 00:00:00.000	18
累计消费5000元	满200元赠送小礼品	2020-07-22 00:00:00.000	19
充值2000元	购物九折, 赠送购物袋	2020-11-16 00:00:00.000	20
> test	test	1905-06-23 00:00:00.000	21

5.2.8.3前端界面

成为vip

请输入成为vip的基本条件

优惠政策

请输入优惠政策

优惠时长

请选择优惠时长

搜索

重置

的基本条件

件

+ 新增

修改

删除

导出

☐

成为vip的基本条件

优惠政策

优惠时长

优惠类型

操作

☐

test

test

1905-06-23

21

[修改](#) [删除](#)

共 21 条

10条/页

<

1

2

3

>

前往

3

页

5.2.9 vip_card

5.2.9.1 SQL 语句

```
INSERT INTO vip_card ( card_id, login_name, id_password, login_password, vip_level, integral, vip_type_id )
```

1. VALUES

```
( 1001, 'test', 'test','test',1001,1001,1);
```

5.2.9.2结果

card_id	login_name	id_password	login_password	vip_level	integral	u_type	vip_type_id
1001	test	test	test	1001	1001	1	1
1000	jacksonhowar3	imnVNe3yfqvFr	jkoV9P0DEjapu	658	375	1	11

5.2.9.3前端界面

+ 新增

修改

删除

导出

☐

会员卡号

登录名

卡密码

登陆密码

会员等级

会员积分

身份类型

优惠类型

操作

☐

1001

test

test

test

1001

1001

1

1

[修改](#) [删除](#)

共 1 条

10条/页

<

1

>

前往

1

页

5.2.10 vip_info

5.2.10.1 SQL 语句

```
INSERT INTO
```

```
vip_info(vip_info.card_id,vip_info.name,vip_info.sex,vip_info.identity_  
id,vip_info.phone,vip_info.job_address)
```

```
VALUES(1001,'test','F','test',123,'test');
```

5.2.10.2 结果

> 会员信息更新成功

> Affected rows: 0

Affected rows: 0

Affected rows: 1

> 查询时间: 0.016s

5.2.10.3 前端界面

会员卡号

1001

姓名

请输入姓名

联系方式

请输入联系方式

收货地址

请输入收货地址

搜索

重置

+ 新增

修改

删除

导出

	会员卡号	姓名	性别	身份证	联系方式	收货地址	操作
<input type="checkbox"/>	1001	test	F	test	123	test	修改 删除

共 1 条 10条/页 < 1 > 前往 1 页

5.3 修改

5.3.1 admin_id

5.3.1.1 SQL 语句

```
UPDATE [dbo].[admin_id] SET [a_id] = 13, [a_password] = 'test13' WHERE [a_id] = 12;
```

5.3.1.2 结果

a_id	a_password	u_type
1	ZljG8BScJ4mTF7	2
2	mmHxNBfBAPvr	2
3	cdDd1c	3
4	z9faO7Zs6Hn1Q	2
5	KVcqhpY1ioQBH	2
6	neEEYuCBWX	3
7	G_FOzPdBJql8	3
8	da7E4wuplc	2
9	S62M9IEm7o_Eh	3
10	GUau9gaLM	2
11	test11	2
13	test13	3

5.3.1.3 前端界面

首页	充值	会员卡信息	vip基本信息	vip的基本信息	管理员工号信息
----	----	-------	---------	----------	---------

登录密码

<input type="checkbox"/>	管理员工号	登录密码	身份类型	操作
<input type="checkbox"/>	11	test11	2	修改 删除
<input type="checkbox"/>	13	test13	3	修改 删除

共 12 条

 前往 页

5.3.2 admin_info

5.3.2.1SQL 语句

```
UPDATE [dbo].[admin_info] SET [a_name] = 'test1' WHERE [a_identity_id] = 'test11';
```

5.3.2.2结果

对象	admin_id @vipone.dbo (vip)...	admin_id @ry-vue (ry)
----	-------------------------------	-----------------------

开始事务
 文本
 筛选
 排序
 导入
 导出
 数据生

a_name	a_sex	a_id	a_identity_id	a_phone
石宇宁	M		1	125514105979657698107232
石安琪	F		4	165217031572017694571853
张岚	F		2	167210620787016960584804
严嘉伦	M		8	271035348632672841425206
杜云熙	M		5	282343686889713525976957
尹宇宁	M		7	288704241280793275984367
杜秀英	F		9	501577821873907597984853
孔子韬	M		3	587465575697665488416873
袁云熙	M		6	638701127119678524580046
方杰宏	M		10	777379742810631010883805
test1	F		11	test11
				123

5.3.2.3前端界面

管理员姓名	<input type="text" value="请输入管理员姓名"/>	工号	<input type="text" value="请输入工号"/>	管理员联系方式	<input type="text" value="请输入管理员联系方式"/>	<input type="button" value="搜索"/>	<input type="button" value="重置"/>
-------	---------------------------------------	----	------------------------------------	---------	---	-----------------------------------	-----------------------------------

<input type="checkbox"/>	管理员姓名	管理员性别	工号	管理员身份证号	管理员联系方式	操作
<input type="checkbox"/>	test1	F	11	test11	123	修改 删除

共 11 条

 前往 页

5.3.3 buy_product

5.3.3.1SQL 语句

```
UPDATE [dbo].[buy_product] SET [buy_product_id] = 1002 WHERE [buy_product_id] = 1001;
```

5.3.3.2结果

表						
对象	admin_id @vipone.d...	admin_id @ry-vue (ry...	admin_info @ry-vue (...	admin_info @vipone...	buy_product @vipon...	
开始事务	文本	筛选	排序	导入	导出	数据生成
buy_product_id	buy_product_name	buy_product_type_id	buy_product_count	buy_product_price	card_id	
1002	test	1	1	1.0000	2	
1000	yherry	9	842	996.9400	726	

5.3.3.3前端界面

物品名称

所属种类

购买数量

总价

会员卡号

搜索

重置

+ 新增

修改

删除

导出

	物品编号	物品名称	所属种类	购买数量	总价	会员卡号	操作
<input type="checkbox"/>	516	miwi pi	9	191	77.62	2	修改 删除
<input type="checkbox"/>	1002	test	1	1	1	2	修改 删除

共 2 条

10条/页

1

前往 1 页

5.3.4 charge

5.3.4.1SQL 语句

```
UPDATE [dbo].[charge] SET [money] = 614.0001, [balance] = 333.0009 WHERE [card_id] = 2;
```

5.3.4.2结果

> 充值成功

> Affected rows: 0

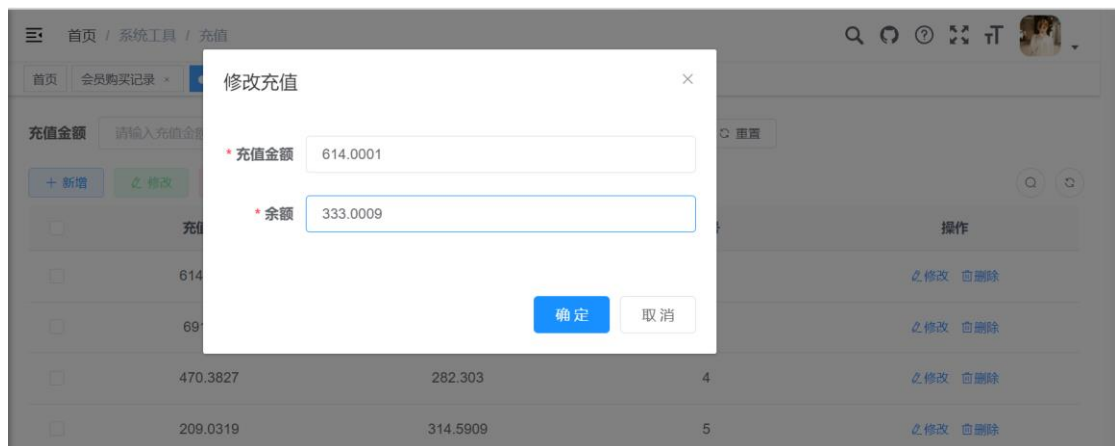
Affected rows: 0

Affected rows: 1

> 查询时间: 0.006s

表					
视图	函数	用户	其它	SQL Server 备份	
对象	charge @vipone.dbo (vip) - 表				
开始事务	文本	筛选	排序	导入	导出
money	balance	card_id			
614.0001	333.0009	2			
691.4960	793.0461	3			
470.2827	282.2827	4			

5.3.4.3前端界面



5.3.5 loss

5.3.5.1SQL 语句

```
UPDATE [dbo].[loss] SET [name] = 'test02' WHERE [identity_id] = '777331534358445';
```

5.3.5.2结果

> Affected rows: 1

> 查询时间: 0.007s

王震南	320 M	212036832 025830123221
张云熙	3 M	18264501437 204202113497
> test02	2 F	123 777331534358
严宇宁	688 M	1030082165 871715081967

5.3.5.3前端界面



5.3.6 product_type

5.3.6.1SQL 语句

```
UPDATE [dbo].[product_type] SET [type_name] = 'test01' WHERE [type_id] = 11;
```

5.3.6.2结果

> 商品类型更新成功
> Affected rows: 0
Affected rows: 0
Affected rows: 1
> 查询时间: 0.007s

信息	摘要	结果 1 (2)
编号	商品类别	
11	test01	
3	电子游戏	
1	服装, 鞋子和珠宝	

5.3.6.3前端界面



5.3.7 myuser

5.3.7.1SQL 语句

```
UPDATE [dbo].[myuser] SET [u_type] = 'test01' WHERE [id] = 4;
```

5.3.7.2结果

> Affected rows: 1
> 查询时间: 0.008s

id	u_type
1	会员
2	管理员
3	超级管理员
4	test01

5.3.7.3前端界面

修改会员卡信息

×

* 登录名

test

* 卡密码

test01

* 登陆密码

test01|

* 会员等级

1001

* 会员积分

1001

* 优惠类型

1

确定

取消

三 首页 / 系统工具 / 会员卡信息

修改成功

充值 × 挂失 × 商品类型 × vip购买记录 × vip基本信息 × vip的基本信息 × 会员卡信息 ×

登录名 test 卡密码 请输入卡密码 登陆密码 请输入登陆密码

会员等级 请输入会员等级 会员积分 请输入会员积分 优惠类型 请输入优惠类型

Q 搜索 重置

+ 新增 修改 删除 导出

Q 刷新

	会员卡号	登录名	卡密码	登陆密码	会员等级	会员积分	身份类型	优惠类型	操作
<input type="checkbox"/>	1001	test	test01	test01	1001	1001	1	1	修改 删除

5.3.10 vip_info

5.3.10.1 SQL 语句

UPDATE [dbo].[vip_info] SET [name] = 'test01', [identity_id] = 'test01',
[job_address] = 'test01' WHERE [identity_id] = 'test'

5.3.10.2 结果

对象	* 演示查询 @vipone.dbo (vip) - 查询	vip_info @vipone.dbo (vip) - 表
开始事务	文本 筛选 排序	导入 导出 数据生成 创建图表
card_id	name	sex
1001	test01	F
1000	许子韬	M
identity_id	phone	job_address
test01	76958694566	123 test01
631118765385	环区南街二巷10号	

5.3.10.3 前端界面

<input type="checkbox"/>	会员卡号	姓名	性别	身份证	联系方式	收货地址	操作
<input type="checkbox"/>	1001	test01	F	test01	123	test01	修改 删除

共 1 条10条/页<1>前往1页

5.4 删除

5.4.1 admin_id

5.4.1.1SQL 语句

```
DELETE FROM [dbo].[admin_id] WHERE [a_id] = 13;
```

5.4.1.2结果

开始事务 文本 筛选 排序

a_id	a_password	u_type
1	ZljG8BScJ4mTF7	2
2	mmHxNBfBAPvr	2
3	cdDd1c	3
4	z9faO7Zs6Hn1Q	2
5	KVcqhpY1ioQBH	2
6	neEEYuCBWX	3
7	G_FOzPdBjql8	3
8	da7E4wuplc	2
9	S62M9IEm7o_Eh	3
10	GUau9gaLM	2
11	test11	2

5.4.1.3前端界面

删除成功

充值 挂失 商品类型 会员购买记录 vip基本信息 vip的基本信息 会员卡信息

登陆密码 请输入登陆密码 搜索 重置

+ 新增 修改 删除 导出

<input type="checkbox"/>	管理员工号	登陆密码	身份类型	操作
<input type="checkbox"/>	11	test11	2	修改 删除

共 11 条10条/页<12>前往2页

5.4.2 admin_info

5.4.2.1SQL 语句

```
DELETE FROM [dbo].[admin_info] WHERE [a_identity_id] = 'test11';
```

5.4.2.2结果

> 修改成功

5.4.2.3前端界面



5.4.3 buy_product

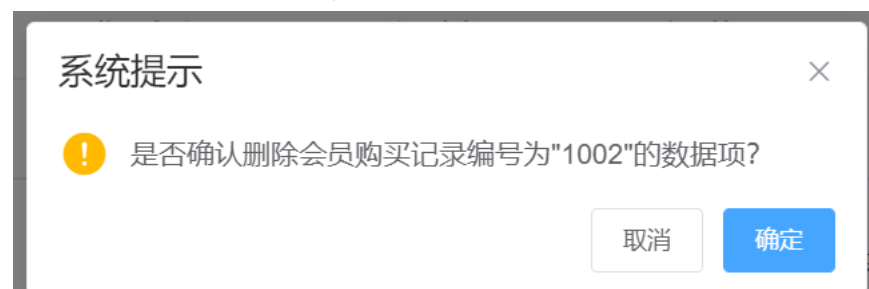
5.4.3.1SQL 语句

```
DELETE FROM [dbo].[buy_product] WHERE [buy_product_id] = 1002;
```

5.4.3.2结果

> 会员购买商品信息修改成功

5.4.3.3前端界面



5.4.4 charge

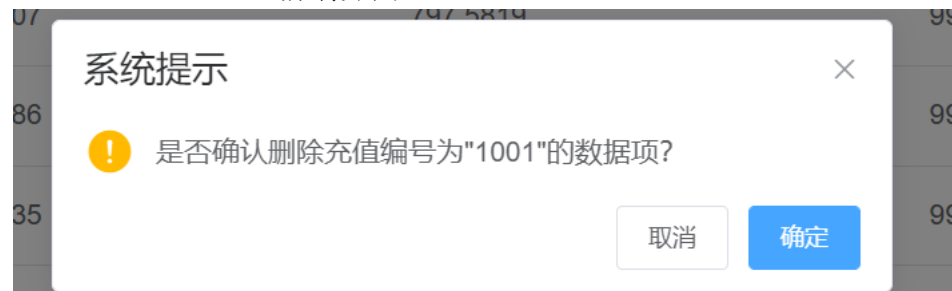
5.4.4.1SQL 语句

```
DELETE FROM [dbo].[charge] WHERE [card_id] = 1001;
```

5.4.4.2结果

> 删除成功

5.4.4.3前端界面



5.4.5 loss

5.4.5.1SQL 语句

```
DELETE FROM [dbo].[loss] WHERE [identity_id] = '777331534358445';
```

5.4.5.2结果

> 卡已找到

5.4.5.3前端界面

系统提示



是否确认删除挂失编号为"777331534358445"的数据项?

取消

确定

5.4.6 product_type

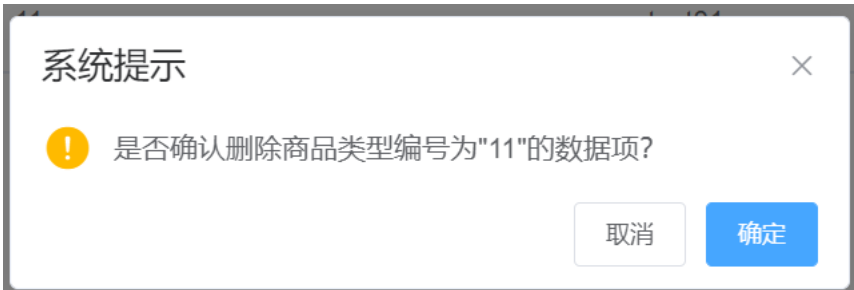
5.4.6.1SQL 语句

DELETE FROM [dbo].[product_type] WHERE [type_id] = 11;

5.4.6.2结果

> 商品类型更新成功

5.4.6.3前端界面



5.4.7 myuser

5.4.7.1SQL 语句

DELETE FROM [dbo].[myuser] WHERE [id] = 4;

5.4.7.2结果

开始事务	文本	筛选	排序
id	u_type		
	1 会员		
	2 管理员		
	3 超级管理员		

5.4.7.3前端界面

系统提示



是否确认删除登陆身份：会员、管理员、超级管理员编号为"4"的数据项?

取消

确定

5.4.8 vip_become

5.4.8.1SQL 语句

DELETE FROM [dbo].[vip_become] WHERE [vip_type_id] = 21;

5.4.8.2结果

> 会员类别更新成功

5.4.8.3前端界面



5.4.9 vip_card

5.4.9.1SQL 语句

```
DELETE FROM [dbo].[vip_card] WHERE [card_id] = 1001;
```

5.4.9.2结果

> 会员卡更新成功

5.4.9.3前端界面



5.4.10 vip_info

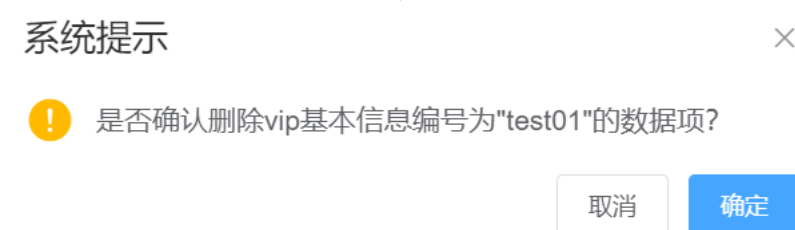
5.4.10.1 SQL 语句

```
DELETE FROM [dbo].[vip_info] WHERE [identity_id] = 'test01';
```

5.4.10.2 结果

> 会员信息更新成功

5.4.10.3 前端界面



5.5 视图

5.5.1 查看当前商品类别

5.5.1.1设计想法

给员工看商品类别

5.5.1.2结果

编号	商品类别
3	电子游戏
1	服装, 鞋子和珠宝
6	工具与家居装饰
8	花园与户外
2	家居及厨房用具
7	乐器用品
4	其他
9	书本
5	玩具与游戏
10	优质美容

5.5.2 查看挂失信息

5.5.2.1 设计想法

供员工核对信息，避免假冒认领。

5.5.2.2 结果

name	card_id	sex	phone	identity_id
林子韬	373	M	208092245	001095013309177130
程杰宏	364	M	17699895991	002144596457607638
江诗涵	969	F	204061635	003861179583969059
李云熙	157	M	109400694	00435679271173761x
傅子韬	140	M	17613341870	006502179677606087

5.5.3 查看会员金额明细

5.5.3.1 设计想法

过滤信息，让会员看自己的金额明细

5.5.3.2 结果

卡号	姓名	充值金额	余额	会员等级	剩余积分
2	汪岚	614.0001	333.0009	806	694
3	张云熙	691.4960	793.0461	935	543
4	钟睿	470.3827	282.3030	62	917
5	潘安班	209.0319	314.5909	762	774

5.5.4 查看会员购买记录

5.5.4.1 设计想法

主要是给会员看自己的消费记录。

5.5.4.2 结果

卡号	姓名	商品名称	购买数量	消费金额
401	许致远	Raspberry	761	52.3600
212	魏子韬	Knwi	384	314.4600
873	陈晓明	Appee	127	970.6700
455	邹晓明	Kiwi	694	233.9700

5.5.5 查看会员基础信息

5.5.5.1 设计想法

让会员和员工能清楚直观地看到自己需要的信息

5.5.5.2结果

	卡号	姓名	性别	身份证	联系方式	收货地址	优惠	截止时间	会员等级	剩余积分
▶	373	林子韬	M	0010950...	2080922...	越秀区中...	充2000...	2008-07...	844	100
	364	程杰宏	M	0021445...	1769989...	坑美十五...	满200元...	2008-01...	973	971

5.5.6 查看员工基础信息

5.5.6.1设计想法

过滤掉密码等不方便公开的信息，把角色编号连接起来，显示中文，方便查阅。

5.5.6.2结果

姓名	性别	身份证	联系方式	工号	角色
▶石宇宁	M	125514105979656	7698107232		1 管理员
石安琪	F	165217031572010378	7694571853		4 管理员
张岚	F	167210620787012568	16960584804		2 管理员

5.5.7 售出商品类别和金额统计

5.5.7.1设计想法

根据会员的销售情况，根据商品类别进行简单的分组，同种类别的金额累加，显示出销售情况最好的商品类别，根据其调整进货策略

5.5.7.2结果

编号	类别名	售出的数量	总成交金额
▶9	书本	123	65617.6400
2	家居及厨房用具	129	61156.7200
4	其他	120	58897.2900

5.6 触发器

5.6.1 admin_id

未设计触发器

5.6.2 admin_info

admininfotri: INSERT, UPDATE, DELETE 之后会显示修改成功，并且查询整个表。

5.6.3 buy_product

buyproducttri: INSERT, UPDATE, DELETE 之后会显示会员购买商品信息修改成功，并显示视图[查看会员购买记录]。

5.6.4 charge

chargeetri: INSERT, UPDATE 后会显示充值成功，并显示视图[查看会员的金额明细]。

delchargetri: DELETE 后会显示删除成功。

5.6.5 loss

lossdeltri: DELETE 后会显示卡已找到。

lossinserttri: INSERT 后会显示挂失成功。

5.6.6 product_type

protypetri: INSERT, UPDATE, DELETE 后会显示商品类型更新成功，并查询视图[查看当前的商品类别]。

5.6.7 myuser

未设计触发器。

5.6.8 vip_become

vipbecome: INSERT, UPDATE, DELETE 后会显示会员类别更新成功。

5.6.9 vip_card

vipcardidtri: INSERT, UPDATE, DELETE 后会显示会员卡更新成功，并显示更新的那条信息。

5.6.10 vip_info

vipinfotri: INSERT, UPDATE, DELETE 后会显示会员信息更新成功，并显示更新的那条信息。

6 存储过程

6.1 整体展示

名	
deladminbyid	insertproducttype
delvipbyid	selectadminbyid
insertcharge	selectvipbyid
insertloss	updataadminbyid
insertproductbyid	updatavipbyid

6.2 过程说明

- del 开头功能为删除，insert 开头为插入，updata 开头为更新信息，select 则是查询信息
- 以 byid 结尾的说明以 id 为查询条件，即提供 id，对特定记录进行增删查改。
- insertloss 实现了挂失时，身份验证的功能，只需要提供姓名和身份证，验证匹配即可插入表，进入挂失流程。
- insertcharge: 提供卡号和金额即可自动更改会员余额信息。

7 授权

7.1 登录名

7.1.1 super_admin: 超级管理员

7.1.2 vipadmin: 普通员工

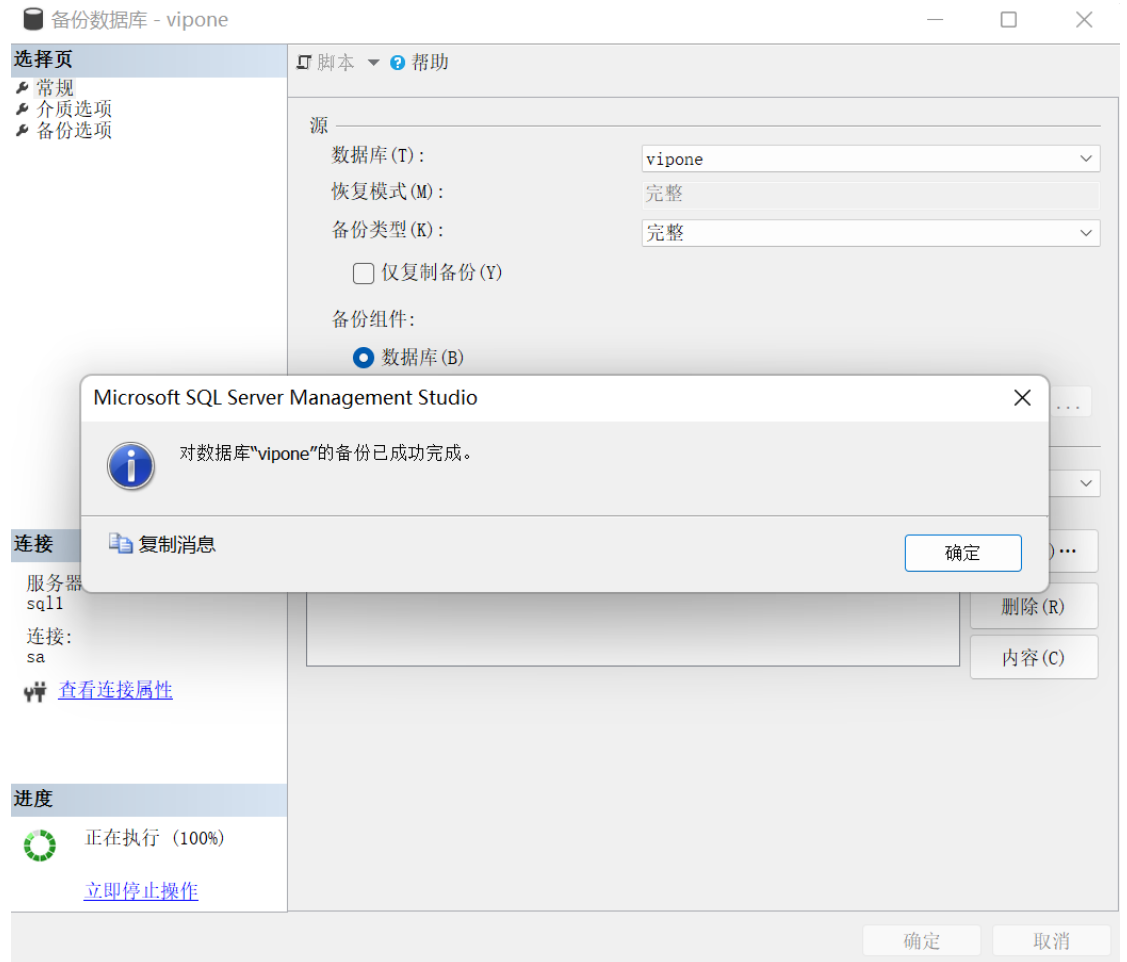
7.1.3 vip: 会员

7.2 角色

7.2.1 super_admin: 超级管理员

模式	对象	ALTER	CONTROL	DELETE	EXECUTE	INSERT	REFERENCES	SELECT	TAKE OWNERSHIP	UPDATE
dbo	admin_id	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G	<input type="checkbox"/>	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/> G
dbo	admin_info	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
dbo	myuser	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
dbo	查看员工基础信息	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> G	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

7.2.2 vipadmin: 普通员工



9 附录

--建库语句

```
CREATE DATABASE [超市会员管理系统]
```

```
ON
```

```
PRIMARY
```

```
(
```

```
    NAME = N'超市会员管理系统',
```

```
    FILENAME = N'/var/opt/mssql/data\超市会员管理系统.ndf',
```

```
    SIZE = 8MB,
```

```
    MAXSIZE = UNLIMITED,
```

```
    FILEGROWTH = 64MB
```

```
)
```

```
LOG ON
```

```
(
```

```
    NAME = N'超市会员管理系统_log',
```

```
    FILENAME = N'/var/opt/mssql/data\超市会员管理系统_log.ldf',
```

```
    SIZE = 8MB,
```

```
    MAXSIZE = UNLIMITED,
```

```
    FILEGROWTH = 64MB
```

```
)
```

```

GO
create table product_type
(
    type_id    int not null
              primary key,
    type_name  varchar(255)
              constraint pro_type
                  unique
)
go

exec sp_addextendedproperty 'MS_Description', '商品类型', 'SCHEMA', 'dbo',
'TABLE', 'product_type'
go

exec sp_addextendedproperty 'MS_Description', '商品类型编号', 'SCHEMA', 'dbo',
'TABLE', 'product_type', 'COLUMN', 'type_id'
go

exec sp_addextendedproperty 'MS_Description', '商品类型', 'SCHEMA', 'dbo',
'TABLE', 'product_type', 'COLUMN', 'type_name'
go

CREATE TRIGGER [dbo].[protypetri]
ON [dbo].[product_type]
WITH EXECUTE AS CALLER
AFTER INSERT, UPDATE, DELETE
AS
BEGIN
    print '商品类型更新成功';
    SELECT * FROM [查看当前的商品类别]
END
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on product_type to vipadmin with grant option
go

create table [user]
(
    id          int                                not null
              constraint user_pk
                  primary key,
    u_type      varchar(255) default '会员' not null

```

```

)
go

exec sp_addextendedproperty 'MS_Description', '登陆身份: 会员、管理员、超级管理员', 'SCHEMA', 'dbo', 'TABLE', 'user'
go

exec sp_addextendedproperty 'MS_Description', '身份编号', 'SCHEMA', 'dbo', 'TABLE', 'user', 'COLUMN', 'id'
go

exec sp_addextendedproperty 'MS_Description', '身份类型', 'SCHEMA', 'dbo', 'TABLE', 'user', 'COLUMN', 'u_type'
go

create table admin_id
(
    a_id          int          not null
        primary key,
    a_password varchar(30)    not null,
    u_type        int default 2 not null
        constraint FK_user
            references [user]
)
go

exec sp_addextendedproperty 'MS_Description', '管理员工号信息', 'SCHEMA', 'dbo', 'TABLE', 'admin_id'
go

exec sp_addextendedproperty 'MS_Description', '管理员工号', 'SCHEMA', 'dbo', 'TABLE', 'admin_id', 'COLUMN', 'a_id'
go

exec sp_addextendedproperty 'MS_Description', '登陆密码', 'SCHEMA', 'dbo', 'TABLE', 'admin_id', 'COLUMN', 'a_password'
go

exec sp_addextendedproperty 'MS_Description', '身份类型', 'SCHEMA', 'dbo', 'TABLE', 'admin_id', 'COLUMN', 'u_type'
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on admin_id to super_admim with grant option

```

```
go

grant update (a_password) on admin_id to vipadmin
go

create table admin_info
(
    a_name          varchar(20) not null,
    a_sex           varchar(5)  not null,
    a_id            int identity
        constraint FK_admin_id
            references admin_id,
    a_identity_id   varchar(100) not null
        constraint admin_info_pk
            primary key,
    a_phone         bigint      not null
)
go

exec sp_addextendedproperty 'MS_Description', '管理员信息', 'SCHEMA', 'dbo',
'TABLE', 'admin_info'
go

exec sp_addextendedproperty 'MS_Description', '管理员姓名', 'SCHEMA', 'dbo',
'TABLE', 'admin_info', 'COLUMN', 'a_name'
go

exec sp_addextendedproperty 'MS_Description', '管理员性别', 'SCHEMA', 'dbo',
'TABLE', 'admin_info', 'COLUMN', 'a_sex'
go

exec sp_addextendedproperty 'MS_Description', '工号', 'SCHEMA', 'dbo', 'TABLE',
'admin_info', 'COLUMN', 'a_id'
go

exec sp_addextendedproperty 'MS_Description', '管理员身份证号', 'SCHEMA', 'dbo',
'TABLE', 'admin_info', 'COLUMN',
    'a_identity_id'
go

exec sp_addextendedproperty 'MS_Description', '管理员联系方式', 'SCHEMA', 'dbo',
'TABLE', 'admin_info', 'COLUMN', 'a_phone'
go
```

```

CREATE TRIGGER [dbo].[admininfotri]
ON [dbo].[admin_info]
FOR INSERT, UPDATE, DELETE
AS
BEGIN
    -- Type the SQL Here.
    PRINT '修改成功';
    SELECT * FROM admin_info;
END
go

exec sp_addextendedproperty 'MS_Description', '管理员信息触发器', 'SCHEMA',
'dbo', 'TABLE', 'admin_info', 'TRIGGER',
    'admininfotri'
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on admin_info to super_admim
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on admin_info to vipadmin
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on [user] to super_admim
go

create table vip_become
(
    basic_conditions      varchar(255) not null,
    preferential_policies varchar(255) not null,
    preferential_time     datetime     not null,
    vip_type_id           int          not null
        primary key
)
go

exec sp_addextendedproperty 'MS_Description', 'vip 的基本信息', 'SCHEMA', 'dbo',
'TABLE', 'vip_become'
go

exec sp_addextendedproperty 'MS_Description', '成为 vip 的基本条件', 'SCHEMA',
'dbo', 'TABLE', 'vip_become', 'COLUMN',

```

```

        'basic_conditions'
go

exec sp_addextendedproperty 'MS_Description', '优惠政策', 'SCHEMA', 'dbo',
'TABLE', 'vip_become', 'COLUMN',
    'preferential_policies'
go

exec sp_addextendedproperty 'MS_Description', '优惠时长', 'SCHEMA', 'dbo',
'TABLE', 'vip_become', 'COLUMN',
    'preferential_time'
go

exec sp_addextendedproperty 'MS_Description', '优惠类型', 'SCHEMA', 'dbo',
'TABLE', 'vip_become', 'COLUMN', 'vip_type_id'
go

CREATE TRIGGER [dbo].[vipbecome]
ON [dbo].[vip_become]
FOR INSERT, UPDATE, DELETE
AS
BEGIN
    PRINT '会员类别更新成功';
END
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on vip_become to vipadmin with grant option
go

create table vip_card
(
    card_id          int          not null
        primary key,
    login_name       varchar(20) not null
        constraint vip_login_name
            unique,
    id_password      varchar(30) not null,
    login_password   varchar(30) not null,
    vip_level        int          not null,
    integral         int          not null,
    u_type           int          not null
        constraint FK_u_id
            references [user]

```

```

        on update cascade on delete cascade,
vip_type_id    int            not null
        constraint FK_vip_type_id
        references vip_become
        on update cascade on delete cascade
)
go

exec sp_addextendedproperty 'MS_Description', '会员卡信息', 'SCHEMA', 'dbo',
'TABLE', 'vip_card'
go

exec sp_addextendedproperty 'MS_Description', '会员卡号', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'card_id'
go

exec sp_addextendedproperty 'MS_Description', '登录名', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'login_name'
go

exec sp_addextendedproperty 'MS_Description', '卡密码', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'id_password'
go

exec sp_addextendedproperty 'MS_Description', '登陆密码', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'login_password'
go

exec sp_addextendedproperty 'MS_Description', '会员等级', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'vip_level'
go

exec sp_addextendedproperty 'MS_Description', '会员积分', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'integral'
go

exec sp_addextendedproperty 'MS_Description', '身份类型', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'u_type'
go

exec sp_addextendedproperty 'MS_Description', '优惠类型', 'SCHEMA', 'dbo',
'TABLE', 'vip_card', 'COLUMN', 'vip_type_id'
go

```



```

create table buy_product
(
    buy_product_id      int          not null
        primary key,
    buy_product_name    varchar(255) not null,
    buy_product_type_id int          not null
        constraint FK_buy_pro_type
            references product_type
            on update cascade on delete cascade,
    buy_product_count   int          not null,
    buy_product_price   money        not null,
    card_id             int          not null
        constraint FK_card_id
            references vip_card
            on update cascade on delete cascade
)
go

exec sp_addextendedproperty 'MS_Description', '会员购买记录', 'SCHEMA', 'dbo',
'TABLE', 'buy_product'
go

exec sp_addextendedproperty 'MS_Description', '物品编号', 'SCHEMA', 'dbo',
'TABLE', 'buy_product', 'COLUMN',
    'buy_product_id'
go

exec sp_addextendedproperty 'MS_Description', '物品名称', 'SCHEMA', 'dbo',
'TABLE', 'buy_product', 'COLUMN',
    'buy_product_name'
go

exec sp_addextendedproperty 'MS_Description', '所属种类', 'SCHEMA', 'dbo',
'TABLE', 'buy_product', 'COLUMN',
    'buy_product_type_id'
go

exec sp_addextendedproperty 'MS_Description', '购买数量', 'SCHEMA', 'dbo',
'TABLE', 'buy_product', 'COLUMN',
    'buy_product_count'
go

exec sp_addextendedproperty 'MS_Description', '总价', 'SCHEMA', 'dbo', 'TABLE',
'buy_product', 'COLUMN',

```

```

        'buy_product_price'
go

exec sp_addextendedproperty 'MS_Description', '会员卡号', 'SCHEMA', 'dbo',
'TABLE', 'buy_product', 'COLUMN', 'card_id'
go

CREATE TRIGGER [dbo].[buyproducttri]
ON [dbo].[buy_product]
FOR INSERT, UPDATE, DELETE
AS
BEGIN
    -- Type the SQL Here.
    PRINT '会员购买商品信息修改成功';
    SELECT * FROM [查看会员购买记录]
END
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on buy_product to vipadmin
go

create table charge
(
    money    money not null,
    balance money not null,
    card_id int    not null
        constraint charge_pk
            primary key
        constraint FK_charge_card_id
            references vip_card
            on update cascade on delete cascade
)
go

exec sp_addextendedproperty 'MS_Description', '充值', 'SCHEMA', 'dbo', 'TABLE',
'charge'
go

exec sp_addextendedproperty 'MS_Description', '充值金额', 'SCHEMA', 'dbo',
'TABLE', 'charge', 'COLUMN', 'money'
go

```

```

exec sp_addextendedproperty 'MS_Description', '余额', 'SCHEMA', 'dbo', 'TABLE',
'charge', 'COLUMN', 'balance'
go

exec sp_addextendedproperty 'MS_Description', '卡号', 'SCHEMA', 'dbo', 'TABLE',
'charge', 'COLUMN', 'card_id'
go

CREATE TRIGGER [dbo].[chargetri]
ON [dbo].[charge]
FOR INSERT, UPDATE
AS
BEGIN
    -- Type the SQL Here.
    PRINT '充值成功';
    SELECT * FROM [查看会员的金额明细]
END
go

CREATE TRIGGER [dbo].[delchargetri]
ON [dbo].[charge]
FOR DELETE
AS
BEGIN
    PRINT '删除成功';
END
go

CREATE TRIGGER [dbo].[vipcardidtri]
ON [dbo].[vip_card]
FOR INSERT, UPDATE, DELETE
AS
BEGIN
    PRINT '会员卡更新成功';
    DECLARE @id int;
    SELECT @id=card_id FROM DELETED;
    EXEC selectvipbyid @id;
END
go

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on vip_card to vipadmin
go

```

```

create table vip_info
(
    card_id      int          not null
        constraint FK_info_card_id
        references vip_card
        on update cascade on delete cascade,
    name         varchar(20)  not null,
    sex          varchar(5)   not null,
    identity_id  varchar(100) not null
        primary key,
    phone        bigint       not null,
    job_address  varchar(255) not null
)
go

exec sp_addextendedproperty 'MS_Description', 'vip 基本信息', 'SCHEMA', 'dbo',
'TABLE', 'vip_info'
go

exec sp_addextendedproperty 'MS_Description', '会员卡号', 'SCHEMA', 'dbo',
'TABLE', 'vip_info', 'COLUMN', 'card_id'
go

exec sp_addextendedproperty 'MS_Description', '姓名', 'SCHEMA', 'dbo', 'TABLE',
'vip_info', 'COLUMN', 'name'
go

exec sp_addextendedproperty 'MS_Description', '性别', 'SCHEMA', 'dbo', 'TABLE',
'vip_info', 'COLUMN', 'sex'
go

exec sp_addextendedproperty 'MS_Description', '身份证', 'SCHEMA', 'dbo',
'TABLE', 'vip_info', 'COLUMN', 'identity_id'
go

exec sp_addextendedproperty 'MS_Description', '联系方式', 'SCHEMA', 'dbo',
'TABLE', 'vip_info', 'COLUMN', 'phone'
go

exec sp_addextendedproperty 'MS_Description', '收货地址', 'SCHEMA', 'dbo',
'TABLE', 'vip_info', 'COLUMN', 'job_address'
go

create table loss

```

```

(
    name          varchar(20)  not null,
    card_id       int          not null,
    sex           varchar(5)   not null,
    phone         bigint       not null,
    identity_id   varchar(100) not null
    primary key
    constraint FK_loss_identity
        references vip_info
        on update cascade on delete cascade
)
go

exec sp_addextendedproperty 'MS_Description', '挂失', 'SCHEMA', 'dbo', 'TABLE',
'loss'
go

exec sp_addextendedproperty 'MS_Description', '姓名', 'SCHEMA', 'dbo', 'TABLE',
'loss', 'COLUMN', 'name'
go

exec sp_addextendedproperty 'MS_Description', '会员卡号', 'SCHEMA', 'dbo',
'TABLE', 'loss', 'COLUMN', 'card_id'
go

exec sp_addextendedproperty 'MS_Description', '性别', 'SCHEMA', 'dbo', 'TABLE',
'loss', 'COLUMN', 'sex'
go

exec sp_addextendedproperty 'MS_Description', '联系方式', 'SCHEMA', 'dbo',
'TABLE', 'loss', 'COLUMN', 'phone'
go

exec sp_addextendedproperty 'MS_Description', '身份证', 'SCHEMA', 'dbo',
'TABLE', 'loss', 'COLUMN', 'identity_id'
go

CREATE TRIGGER [dbo].[lossdeltri]
ON [dbo].[loss]
FOR DELETE
AS
BEGIN
    PRINT '卡已找到';
END

```

```
go

CREATE TRIGGER [dbo].[lossinsertri]
ON [dbo].[loss]
FOR INSERT
AS
BEGIN
    PRINT '挂失成功';
END
go

grant insert, select, update on loss to vip
go

grant insert, select, update on loss to vipadmin
go

CREATE TRIGGER [dbo].[vipinfotri]
ON [dbo].[vip_info]
FOR INSERT, UPDATE, DELETE
AS
BEGIN
    PRINT '会员信息更新成功';
    DECLARE @id int;
    SELECT @id=card_id FROM DELETED;
    EXEC selectvipbyid @id;
END
go

grant alter, control, delete, insert, references, take ownership, view
definition on vip_info to vip
go

grant select, update on vip_info to vip with grant option
go

grant alter, control, delete, insert, references, take ownership, view
definition on vip_info to vipadmin
go

grant select, update on vip_info to vipadmin with grant option
go

CREATE VIEW [dbo].[售出商品类别计数] AS SELECT
```

```

        product_type.type_id AS [编号],
        product_type.type_name AS [类别名],
        COUNT(product_type.type_name) AS [售出的数量],
        SUM(buy_product.buy_product_price) AS [总成交金额]
FROM
    dbo.buy_product
    INNER JOIN
    dbo.product_type
    ON
        buy_product.buy_product_type_id = product_type.type_id
GROUP BY
    product_type.type_name,
    product_type.type_id
ORDER BY
    总成交金额 DESC
OFFSET 0 ROWS
FETCH NEXT 10 ROWS ONLY
go

```

```

grant alter, control, delete, insert, references, select, take ownership, update,
view definition on 售出商品类别和金额统计 to vipadmin with grant option
go

```

```

CREATE VIEW [dbo].[查看会员基础信息] AS SELECT
    vip_card.card_id AS [卡号],
    vip_info.name AS [姓名],
    vip_info.sex AS [性别],
    vip_info.identity_id AS [身份证],
    vip_info.phone AS [联系方式],
    vip_info.job_address AS [收货地址],
    vip_become.preferential_policies AS [优惠],
    vip_become.preferential_time AS [截止时间],
    vip_card.vip_level AS [会员等级],
    vip_card.integral AS [剩余积分]
FROM
    dbo.vip_card
    INNER JOIN
    dbo.vip_info
    ON
        vip_card.card_id = vip_info.card_id
    INNER JOIN
    dbo.vip_become
    ON
        vip_card.vip_type_id = vip_become.vip_type_id

```

```

go

grant select on 查看会员基础信息 to vip
go

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看会员基础信息 to vipadmin
go

grant select on 查看会员基础信息 to vipadmin with grant option
go

CREATE VIEW [dbo].[查看会员的金额明细] AS SELECT
    vip_info.card_id AS [卡号],
    vip_info.name AS [姓名],
    charge.money AS [充值金额],
    charge.balance AS [余额],
    vip_card.vip_level AS [会员等级],
    vip_card.integral AS [剩余积分]
FROM
    dbo.charge
    INNER JOIN
    dbo.vip_card
    ON
        charge.card_id = vip_card.card_id
    INNER JOIN
    dbo.vip_info
    ON
        vip_card.card_id = vip_info.card_id
go

grant select on 查看会员的金额明细 to vip
go

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看会员的金额明细 to vipadmin
go

grant select on 查看会员的金额明细 to vipadmin with grant option
go

CREATE VIEW [dbo].[查看会员购买记录] AS SELECT
    vip_card.card_id AS [卡号],
    vip_info.name AS [姓名],

```



```

        buy_product.buy_product_name AS [商品名称],
        buy_product.buy_product_count AS [购买数量],
        buy_product.buy_product_price AS [消费金额]
FROM
    dbo.buy_product
    INNER JOIN
    dbo.product_type
    ON
        buy_product.buy_product_type_id = product_type.type_id
    INNER JOIN
    dbo.vip_card
    ON
        buy_product.card_id = vip_card.card_id
    INNER JOIN
    dbo.vip_info
    ON
        vip_card.card_id = vip_info.card_id
go

grant select on 查看会员购买记录 to vip
go

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看会员购买记录 to vipadmin
go

grant select on 查看会员购买记录 to vipadmin with grant option
go

CREATE VIEW [dbo].[查看员工信息] AS SELECT
    [管理员信息].a_name AS [姓名],
    [管理员信息].a_sex AS [性别],
    [管理员信息].a_identity_id AS [身份证],
    [管理员信息].a_phone AS [联系方式],
    [管理员信息].a_id AS [工号],
    [user].u_type AS [角色]
FROM
    dbo.admin_id
    INNER JOIN
    dbo.admin_info AS [管理员信息]
    ON
        admin_id.a_id = [管理员信息].a_id
    INNER JOIN
    dbo.[user]

```

```

        ON
            admin_id.u_type = [user].id
go

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看员工基础信息 to super_admim
go

grant select on 查看员工基础信息 to super_admim with grant option
go

grant select on 查看员工基础信息 to vipadmin
go

CREATE VIEW [dbo].[查看当前的商品类别] AS SELECT
    product_type.type_id AS [编号],
    product_type.type_name AS [商品类别]
FROM
    dbo.product_type
go

grant alter, control, references, take ownership, view definition on 查看当前
的商品类别 to vipadmin
go

grant delete, insert, update on 查看当前的商品类别 to vipadmin with grant option
go

deny select on 查看当前的商品类别 to vipadmin
go

CREATE VIEW [dbo].[查看挂失信息] AS SELECT
    loss.*
FROM
    dbo.loss
go

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看挂失信息 to vip
go

grant select on 查看挂失信息 to vip with grant option
go

```

```

grant alter, control, delete, insert, references, take ownership, update, view
definition on 查看挂失信息 to vipadmin
go

grant select on 查看挂失信息 to vipadmin with grant option
go

CREATE PROCEDURE deladminbyid(@id INT) AS
DELETE
FROM admin_id
WHERE admin_id.a_id = @id
go

create or alter procedure delvipbyid(@id int) as
-- missing source code
go

CREATE PROCEDURE insertcharge(@chargrmoney money, @id int) AS
SELECT *
FROM [查看会员的金额明细]
WHERE [查看会员的金额明细].[卡号]=@id;
UPDATE charge SET charge.money=charge.money+@chargrmoney
WHERE charge.card_id=@id;
UPDATE charge SET charge.balance=charge.balance+@chargrmoney
WHERE charge.card_id=@id;
go

CREATE PROCEDURE insertloss(@lossname VARCHAR(20), @lossidentityid
VARCHAR(100)) AS
DECLARE @name VARCHAR(20), @identityid VARCHAR(100), @cardid int, @sex
VARCHAR(5), @phone bigint;
SELECT @name = vip_info.name,
       @identityid = vip_info.identity_id,
       @cardid = vip_info.card_id,
       @sex = vip_info.sex,
       @phone = vip_info.phone
FROM vip_info
WHERE vip_info.name = @lossname
  and vip_info.identity_id = @lossidentityid;
INSERT loss(name, card_id, sex, phone, identity_id)
VALUES (@name, @cardid, @sex, @phone, @identityid);
go

```

```

CREATE PROCEDURE insertproductbyid(@pid INT, @name VARCHAR(225), @typeid INT,
@productcount INT,
                                @price money, @cardid INT) AS
INSERT buy_product (buy_product_id, buy_product_name, buy_product_type_id,
buy_product_count, buy_product_price,
                    card_id)
VALUES
    (@pid, @name, @typeid, @productcount, @price, @cardid)
go

CREATE PROCEDURE insertproducttype ( @id INT,@name VARCHAR ( 255 ) ) AS SELECT
*
FROM
    [查看当前的商品类别];
INSERT product_type (type_id, type_name )
VALUES
    ( @id,@name )
go

create or alter procedure selectadminbyid(@id int) as
-- missing source code
go

create or alter procedure selectvipbyid(@id int) as
-- missing source code
go

CREATE PROCEDURE updataadminbyid ( @id INT,@name VARCHAR ( 20 ),@identityid
VARCHAR ( 100 ),@phone INT ) AS UPDATE admin_info
SET admin_info.a_name=@name,
    admin_info.a_identity_id=@identityid,
admin_info.a_phone=@phone

WHERE
    admin_info.a_id = @id
go

CREATE PROCEDURE updatavipbyid ( @id INT,@name VARCHAR ( 20 ),@identityid
VARCHAR ( 100 ),@phone INT,@address VARCHAR(255) ) AS UPDATE
vip_info
SET vip_info.name=@name,
vip_info.identity_id=@identityid,
vip_info.phone=@phone ,
vip_info.job_address=@address

```

```
WHERE
    vip_info.card_id = @id
go
```