



# 广东工业大学

## 数据库课程设计

课程名称\_\_\_\_\_数据库课程设计\_\_\_\_\_

题目名称\_\_\_\_\_超市管理系统\_\_\_\_\_

学生学院\_\_\_\_\_管理学院\_\_\_\_\_

专业班级\_\_\_\_\_信息管理与信息系统创新班（1）班\_\_\_\_\_

学    号\_\_\_\_\_3120003369\_\_\_\_\_

学生姓名\_\_\_\_\_洪晓彬\_\_\_\_\_

指导教师\_\_\_\_\_吴学雁\_\_\_\_\_

2022 年 5 月

# 目 录

<b>1. 需求分析</b>	<b>1</b>
<b>1.1 用户需求分析</b>	<b>1</b>
1.1.1 行业背景	1
1.1.2 系统现状	1
1.1.3 解决的主要问题	1
1.1.4 达到的具体指标	2
<b>1.2 业务流程分析</b>	<b>2</b>
1.2.1 基本业务流程图	2
1.2.2 顶层数据流图	3
1.2.3 中层数据流图	3
1.2.4 底层数据流图（核心模块）	4
<b>1.3 信息需求分析</b>	<b>6</b>
1.3.1 资料收集	6
1.3.2 事项分析	6
<b>1.4 功能需求分析</b>	<b>12</b>
1.4.1 管理员功能模块	12
1.4.2 收银员功能模块	14
<b>2. 数据库概念模型设计</b>	<b>16</b>
<b>2.1 分 ER 图</b>	<b>16</b>
2.1.1 每个实体的 ER 图	16
2.1.2 主实体间联系	19
<b>2.2 总 ER 图</b>	<b>21</b>
<b>3. 数据库逻辑模型设计</b>	<b>21</b>
<b>3.1 一般逻辑模型转化</b>	<b>21</b>
3.1.1 ER 图转化基本表	21
3.1.2 基本表优化	25
3.1.3 最终基本表	25

3.2 具体逻辑模型转化.....	26
3.3 视图设计 .....	30
4. 数据库物理设计和数据库保护性设计 .....	31
4.1 索引设计 .....	31
4.2 表间关联设计 .....	34
4.3 完整性设计 .....	34
5. 处理功能设计 .....	37
5.1 主控模块设计 .....	37
5.2 子模块设计 .....	39
5.3 查询设计 .....	43
6. 数据库应用系统的实现.....	46
6.1 数据库及其表结构的建立 .....	46
6.1.1 表的建立 .....	46
6.1.2 视图的建立.....	50
6.1.3 建立索引 <sup>①</sup> .....	51
6.2 数据输入 .....	53
6.3 模块实现 .....	56
7. 数据库应用系统运行及感想 .....	65
7.1 系统运行说明 <sup>1</sup> .....	65
7.2 系统使用说明与展示.....	65
7.3 系统评价与体会 .....	77
7.3.1 系统评价 .....	77
7.3.2 体会 .....	77
参考文献 .....	78

# 1.需求分析

## 1.1 用户需求分析

### 1.1.1 行业背景

随着现代科学技术的高速发展，计算机技术已经渗透到各个领域，成为各行业必不可少的工具。网络与信息化的普及已经成为各行业发展不可缺少的一部分。而各种管理信息系统的开发为行业竞争提供有效的管理信息平台，也是行业信息化发展的关键。

现如今超市经营管理变得越来越复杂，早起的售货员手算结账，手动记录账本的形式已经不能满足现有的销售业的发展，因此需要引入一个综合性的超市管理系统来提高销售效率。作为国内的中小型超市，特别是农村城镇的一些小型超市，至今在信息化的步伐也落后于大型超市，其商品入库，结账结算等手段大都依靠人工记录数据，效率低下。

因此，通过综合性的超市管理系统将超市的各项信息进行有效的归类以及业务的处理，有利于降低经营成本、提高经营效率，从而实现超市的最大效益化，因此，合适的超市管理系统的建立对于农村城镇的小型超市迫切需求。

### 1.1.2 系统现状

- 1.农村城镇地区缺少超市管理系统支持
- 2.大部分系统只支持收银结账功能
- 3.现有系统缺少超市综合性管理

### 1.1.3 解决的主要问题

- 1.解决农村城镇小型超市管理效率低下问题
- 2.解决现有系统功能单一问题

### 1.1.4 达到的具体指标

- 1.提高农村城镇地区小型超市管理效率
- 2.除了传统的收银结账功能增加商品入库，补货管理
- 3.增加超市会员管理，增加超市销售样式多样化
- 4.增加超市销售管理，提高超市店主等管理员对员工的业绩管理，可调动薪资等
- 5.增加商品类别、员工类别、商品规格等管理，实现超市相关信息的统一标准

## 1.2 业务流程分析

### 1.2.1 基本业务流程图

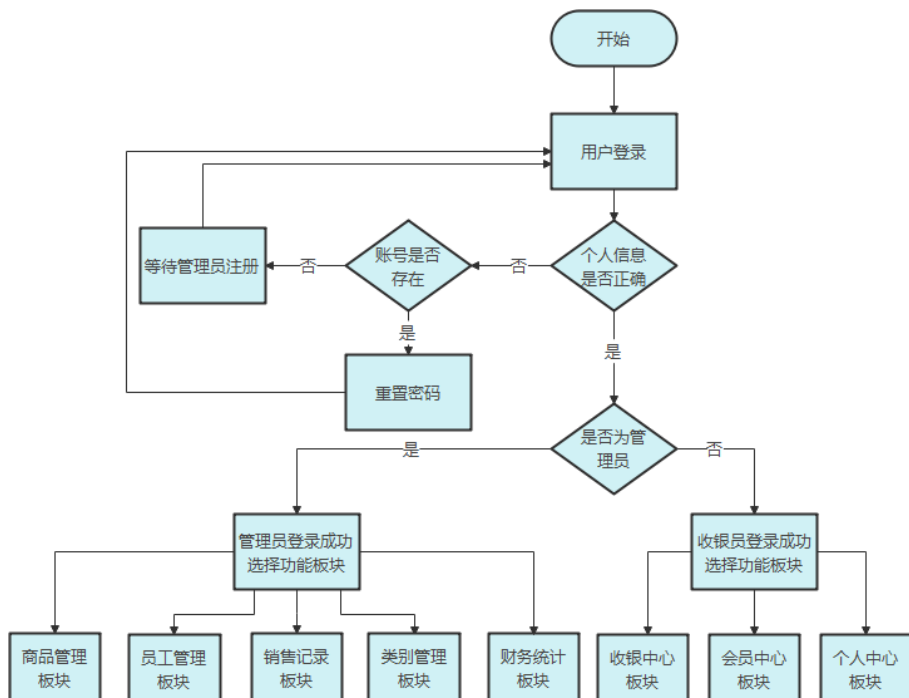


图 1 基本业务流程图

根据需求，超市管理系统的使用者包括超市管理员（简称管理员）和收银员，用户登录成功后，管理员可以对商品管理，包括商品的添加、删除、修改、查询等操作，可以对员工进行管理，包括员工的添加，删除，修改，查询等操作，可以进行销售管理，主要是查询操作，还可以对类别管理和财务统计，方便对员工的业绩考核等业务。

收银员登录成功后，核心的业务是进行结账收银，其次是对会员进行管理，可以注册会员，删除会员，会员续费等业务，除此之外还可以对收银员的个人信息和个人销售记录进行管理，包括修改个人信息、修改密码等和个人销售记录查询，统计等业务。

1.2.2 顶层数据流图

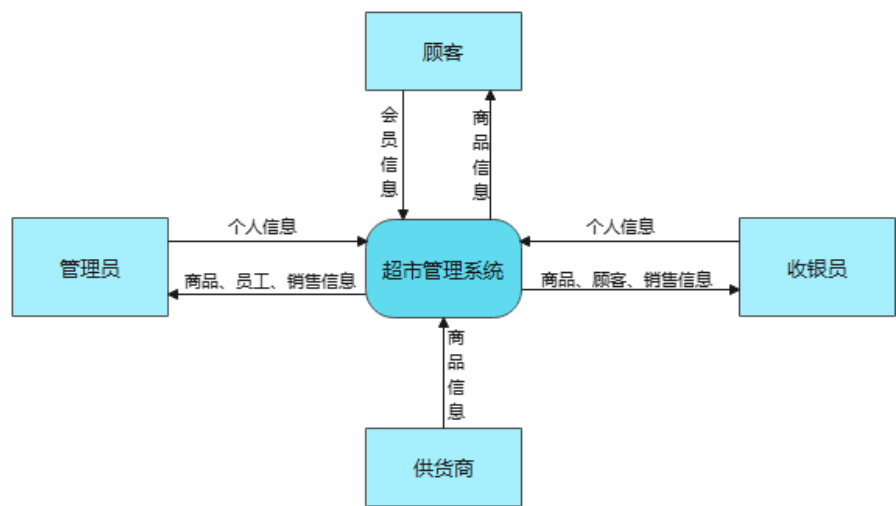


图 2 顶层数据流图

1.2.3 中层数据流图

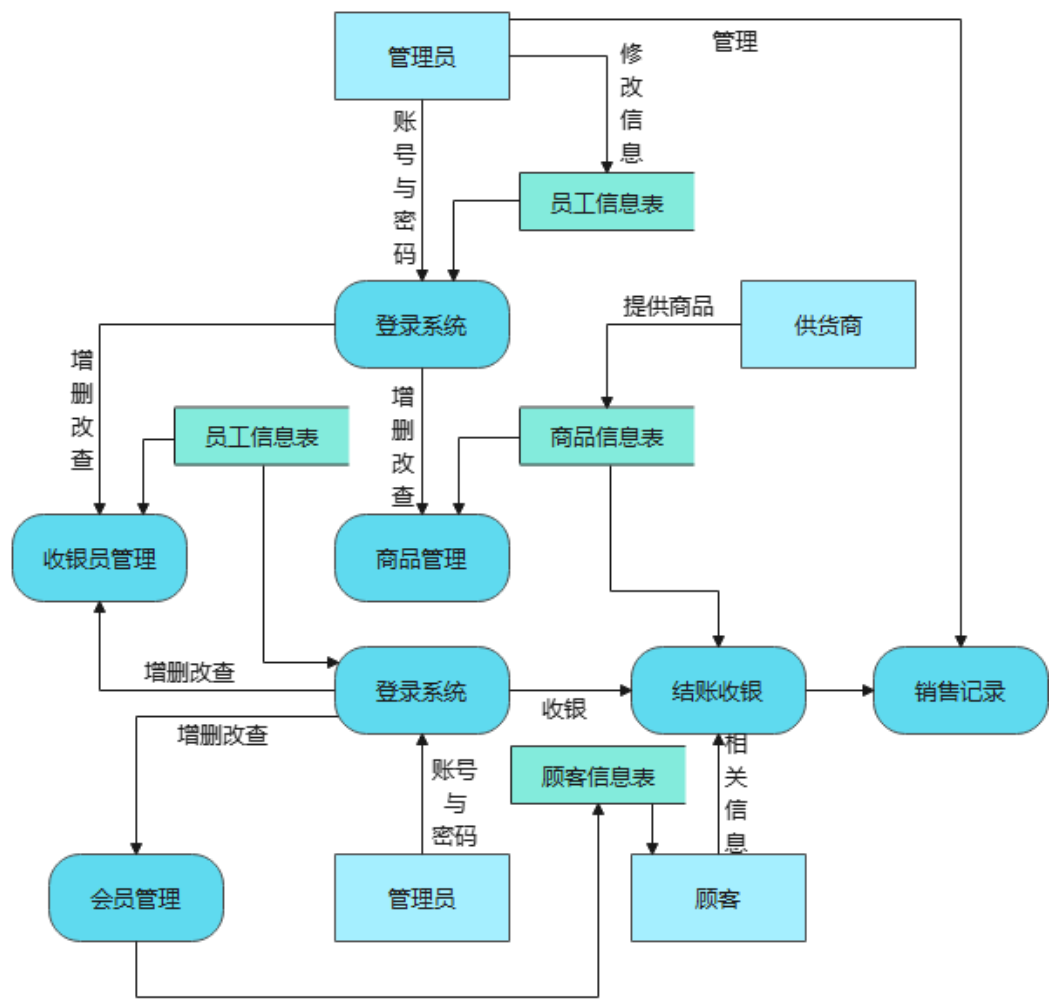


图 3 中层数据流图

1.2.4 底层数据流图（核心模块）

收银员收银结账功能：

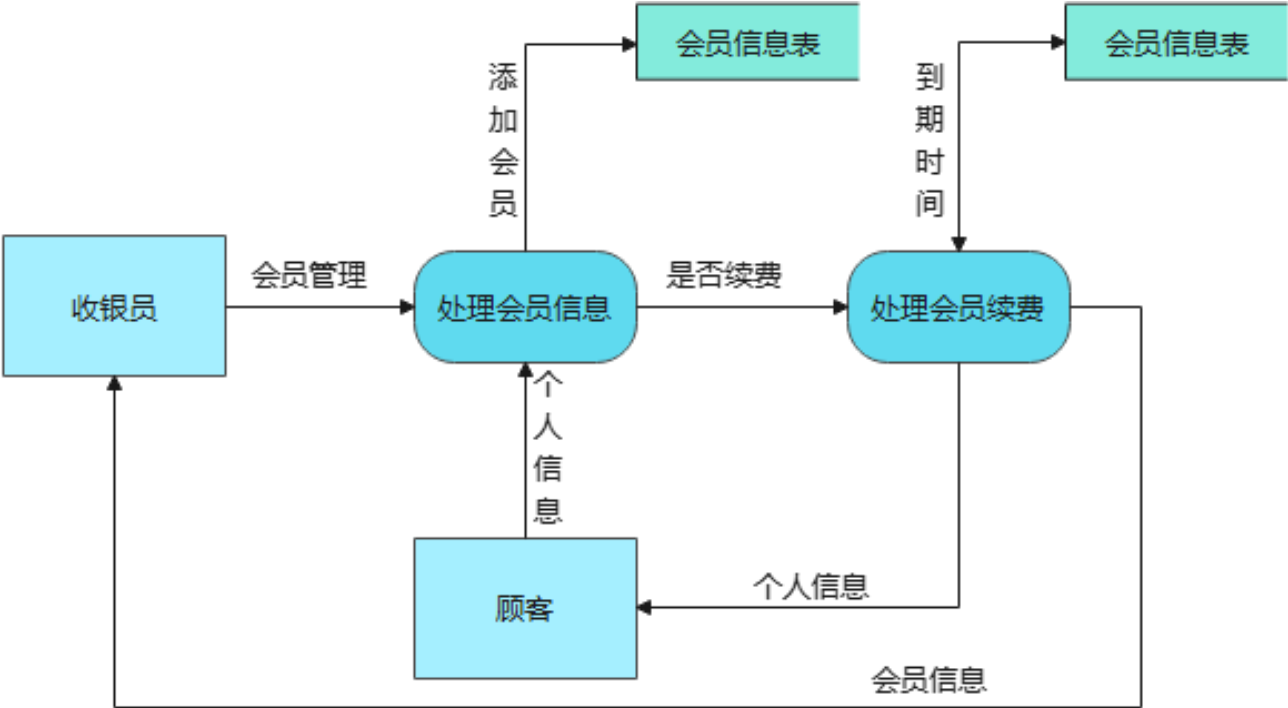


图 4 收银员结账数据流图

顾客结账形成销售记录的数据流动：

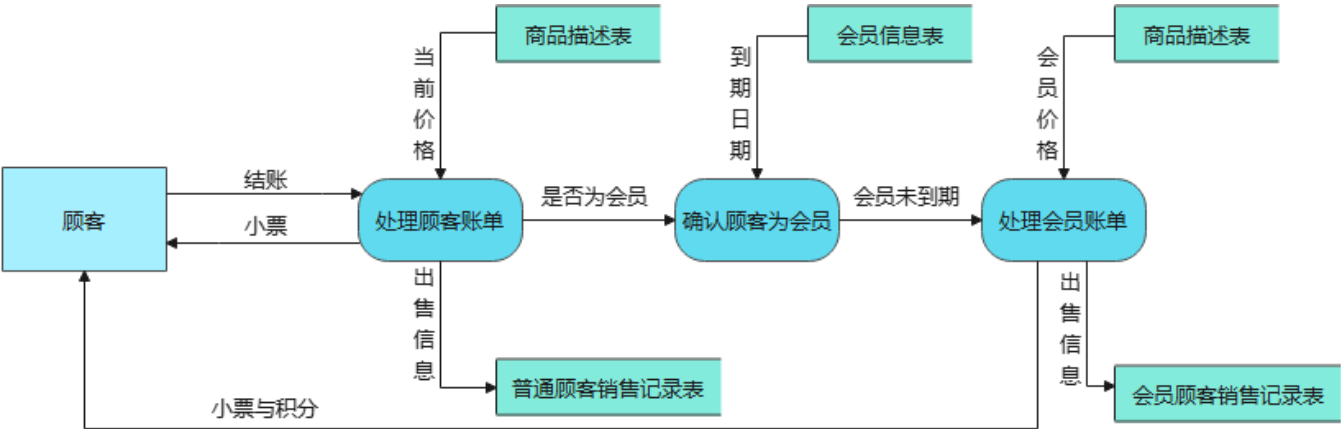


图 5 顾客结账数据流图

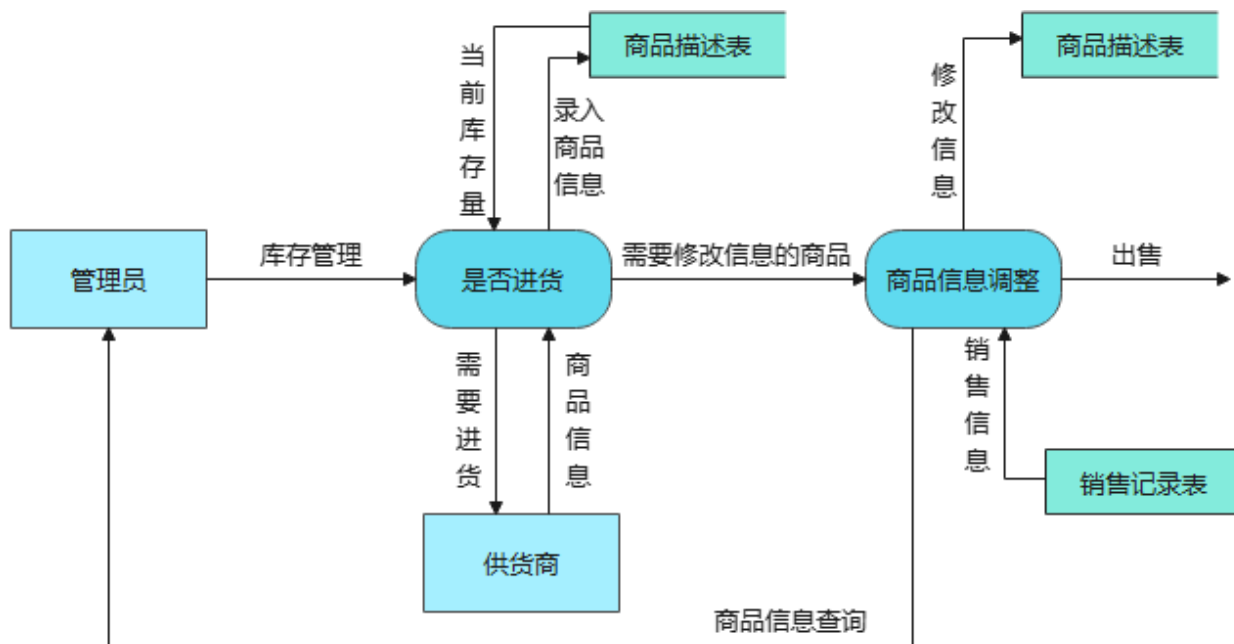
管理员对商品信息管理的功能：

图 6 管理员对商品信息管理数据流图

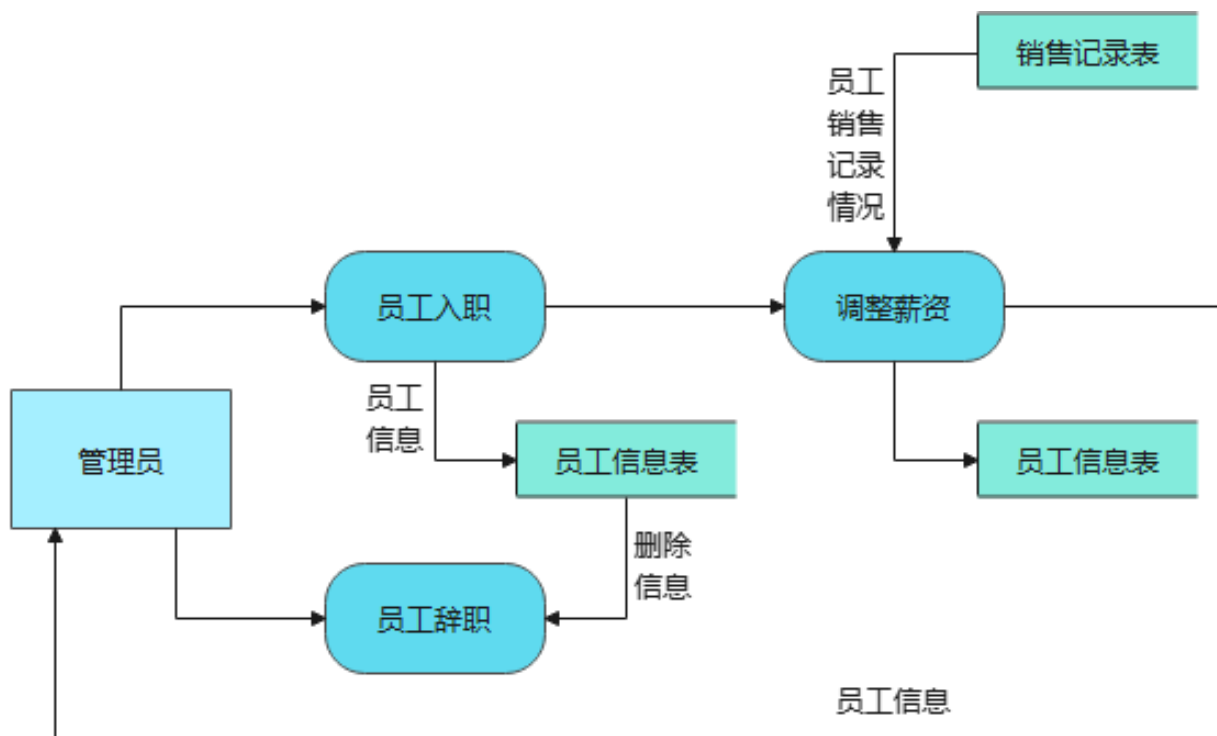
管理员对员工信息管理的功能：

图 7 管理员对员工信息管理数据流图



## 1.3 信息需求分析

### 1.3.1 资料收集

- ①国家商品分类标准
- ②常用超市小票信息
- ③商品编号统一标识标准
- ④采购商信息参照 商品终端网信息

### 1.3.2 事项分析

#### 数据字典

##### (1)数据项

<p>1. 数据项名称：<u>商品编号</u></p> <p>数据项别名：商品号</p> <p>说明： 作为超市描述商品的唯一标识码</p> <p>类型：字符型</p> <p>长度：16</p> <p>与其他数据项的逻辑关系：可以标识其他数据项（商品名称，普通售价，会员折扣，会员售价，供应商名称，进货数量，进货价格，进货日期，规格名称，类别名称）</p>	<p>3. 数据项名称：<u>商品名称</u></p> <p>数据项别名：商品名</p> <p>说明： 作为超市中商品的名称</p> <p>类型：字符型</p> <p>长度：32</p> <p>取值含义：特定标识商品的文字</p>
<p>2. 数据项名称：<u>商品类别</u></p> <p>数据项别名：类别</p> <p>说明： 区分不同的商品，如电子产品、玩具等</p> <p>类型：字符型</p> <p>长度：16</p> <p>取值范围：符合标准的商品分门别类</p> <p>取值含义：特定标识商品类别的文字</p>	<p>4. 数据项名称：<u>生产地址</u></p> <p>数据项别名：商品生产地</p> <p>说明： 记录产品的生产地址</p> <p>类型：字符型</p> <p>长度：32</p> <p>取值范围：只包含省份和城市的地址，格式为 XX 省 XX 市</p> <p>取值含义：包含生产地的省份和城市</p>

5. 数据项名称: 供应商名称  
数据项别名: 商品供货商  
说明: 记录商品的来源  
类型: 字符型  
长度: 32  
取值含义: 表示供应商名称(或简称)的文字

6. 数据项名称: 进货数量  
数据项别名: 商品进货数量  
说明: 记录商品进货到超市的数量  
类型: 双精度浮点型  
长度: 6  
取值含义: 该值需要与规格(计量单位)同时使用才有意义, 不限小数位数  
与其他数据项的逻辑关系: 与商品规格组合能够计量某一个商品的数量

7. 数据项名称: 普通售价  
数据项别名: 普通顾客售价  
说明: 销售给普通顾客(非会员)的价格  
类型: 双精度浮点型  
长度: (6, 2)  
取值范围: 0.00-9999.99  
取值含义: 最多保留两位小数, 单位为元

8. 数据项名称: 会员售价  
数据项别名: 会员顾客售价  
说明: 销售给会员顾客的价格

9. 数据项名称: 进货日期  
数据项别名: 商品进货日期  
说明: 记录商品进货到超市的日期  
类型: 日期类型  
长度: 3  
取值含义: 供应商将商品提供给超市的年月日即为进货日期

10. 数据项名称: 进价  
数据项别名: 商品进价  
说明: 商品进货到超市时的价格  
类型: 双精度浮点型  
长度: (6, 2) [说明: 最大 6 位长度, 保留 2 位小数, 下同]  
取值范围: 0.00-9999.99  
取值含义: 最大保留两位小数, 单位为元

11. 数据项名称: 商品折扣  
数据项别名: 会员销售折扣  
说明: 商品销售给会员顾客的折扣优惠  
类型: 双精度浮点型  
长度: (3, 2)  
取值范围: 0.00-1.00  
取值含义: 最多保留两位小数, 相当于在普通售价基础上打几折

12. 数据项名称: 商品规格  
数据项别名: 规格  
说明: 记录商品计数单位如 KG、件等

类型：双精度浮点型  
长度：(6,2)  
取值范围：0.00-9999.99  
取值含义：最多保留两位小数，单位为元  
与其他数据项的逻辑关系：由数据项普通售价与商品折扣的乘积获得

13. 数据项名称：员工编号  
数据项别名：员工编号  
说明：作为员工的唯一标识  
类型：字符型  
长度：10  
取值范围：0000000000-9999999999  
取值含义：按照年份后两位+月份+性别标识（01 为男，02 为女）+四位随机码，一共 10 位数字组成  
与其他数据项的逻辑关系：可以标识其他数据项（员工姓名，密码，住址，详细地址，联系电话，出生日期，性别，基本工资和职位名称）

14. 数据项名称：员工登录密码  
数据项别名：登录密码  
说明：作为员工登录的验证信息  
类型：字符型  
长度：20  
取值含义：员工初始密码与员工编号一致

类型：字符型  
长度：16  
取值范围：计量单位  
取值含义：为数量提供计量单位，让数量具有实在性意义，便于数量上的统计

15. 数据项名称：员工姓名  
数据项别名：姓名  
说明：记录员工的姓名  
类型：字符型  
长度：8  
取值范围：不超过四个汉字  
取值含义：若存在重名，则在姓名最后用数字标识

16. 数据项名称：员工住址  
数据项别名：住址  
说明：记录员工的住址，包括省份城市县区  
类型：字符型  
长度：64  
取值范围：只记录到省份，城市，地区如 XX 省 XX 市 XX 县（区）  
取值含义：表示员工住址的大致位置  
与其他数据项的逻辑关系：与数据项详细地址组合成为员工住址的完整地址

17. 数据项名称: 员工联系电话  
数据项别名: 联系电话  
说明: 记录员工联系电话  
类型: 字符型  
长度: 12  
取值含义: 表示员工的固定电话或者手机号码

18. 数据项名称: 员工生日  
数据项别名: 出生日期  
说明: 记录员工的生日, 方便统计年龄  
类型: 日期类型  
长度: 3

19. 数据项名称: 基本工资  
数据项别名: 员工基本工资  
说明: 记录员工基本工资  
类型: 双精度浮点型  
长度: (7, 2)  
取值范围: 0.00-99999.99  
取值含义: 最多保留两位小数, 单位为元

20. 数据项名称: 员工职位  
数据项别名: 职位  
说明: 记录员工职位, 可用于系统登录权限选择  
类型: 字符型  
长度: 16  
取值含义: 符合约定的员工职位名称命名标准

21. 数据项名称: 员工性别  
数据项别名: 性别  
说明: 记录员工的性别  
类型: 字符型  
长度: 4  
取值范围: ‘男’或‘女’

22. 数据项名称: 会员编号  
数据项别名: 会员号  
说明: 作为会员的唯一标识  
类型: 字符型  
长度: 15  
取值范围: V0000000000000-V9999999999999  
取值含义: 按照 V+年份后两位+月份+日期+六位随机码组成的 13 位标识码组成  
与其他数据项的逻辑关系: 可以标识其他数据项(会员姓名, 会员性别, 会员联系电话, 注册日期, 到期日期, 会员状态, 积分数量)

23. 数据项名称: 会员姓名  
数据项别名: 会员名  
说明: 记录会员的姓名  
类型: 字符型  
长度: 10  
取值范围: 最多包含五位汉字  
取值含义: 若出现重名, 则在会员姓名数据项后添加数字左右区分

24. 数据项名称: 会员联系电话  
数据项别名: 联系电话  
说明: 记录会员的联系电话  
类型: 字符型  
长度: 12  
取值含义: 表示会员的固定电话或者手机号码

25. 数据项名称: 会员注册日期  
数据项别名: 注册日期  
说明: 记录会员的注册日期  
类型: 日期类型  
长度: 3

26. 数据项名称: 会员状态  
数据项别名: 状态  
说明: 记录会员是否到期的状态  
类型: 字符型  
长度: 8

27. 数据项名称: 会员性别  
数据项别名: 性别  
说明: 记录会员的性别  
类型: 字符型  
长度: 4  
取值范围: ‘男’或‘女’

28. 数据项名称: 会员积分  
数据项别名: 积分  
说明: 记录会员每一次消费获得的积分  
类型: 整型  
长度: 8  
取值范围: 00000000-99999999  
取值含义: 记录每一次会员消费产生的积分, 可用于积分兑换等活动

29. 数据项名称: 会员到期日期  
数据项别名: 到期日期  
说明: 记录会员到期的日期, 判断是否可用会员  
类型: 日期类型  
长度: 3

30. 数据项名称: 付款金额  
数据项别名: 现金金额  
说明: 结账时顾客支付的金额  
类型: 双精度浮点型  
长度: (8, 2)

<p>取值范围：‘未到期’或‘已到期’</p> <p>与其他数据项的逻辑关系：根据会员到期日期数据项与今日日期对比，若超过今日日期则记录为已到期，否则为未到期</p> <p>31. 数据项名称：<u>登录时间</u></p> <p>数据项别名：员工登录系统时间</p> <p>说明：记录员工上班时间</p> <p>类型：日期时间类型</p> <p>长度：8</p> <p>32. 数据项名称：<u>购买数量</u></p> <p>数据项别名：购买数量</p> <p>说明：记录顾客一次购买的商品数量</p> <p>类型：双精度浮点型</p> <p>长度：(6, 2)</p> <p>取值范围：0.00-9999.99</p> <p>取值含义：与商品规格数据项组合即可计量某一商品</p>	<p>取值范围：0.00-999999.99</p> <p>取值含义：最多保留两位小数，单位为元</p> <p>33. 数据项名称：<u>注销时间</u></p> <p>数据项别名：员工注销系统时间</p> <p>说明：记录员工下班时间</p> <p>类型：日期时间类型</p> <p>长度：8</p> <p>34. 数据项名称：<u>小票编号</u></p> <p>数据项别名：小票编号</p> <p>说明：记录每一次交易的生成的小票的编号</p> <p>类型：字符型</p> <p>长度：12</p> <p>取值范围：000000000000-999999999999</p> <p>取值含义：由六位年月日和六位随机数字组成</p>
---	--

### 主要数据结构

数据结构名称	含义说明	数据组成
商品信息表	记录商品的相关信息	商品编号+商品名称+商品类别+进货商+生产地+进货日期+进货数量+进价+普通售价+会员折扣+商品规格
员工信息表	记录员工相关信息	员工编号+员工姓名+密码+地址+详细地址+职位+性别+基本工资
会员信息表	记录会员相关信息	会员编号+会员姓名+联系电话+积分+注册日期+到期日期+状态
小票信息表	记录顾客购买的小票信息	小票编号+商品号+购买数量

普通顾客销售表	记录普通顾客销售信息	小票编号+顾客编号+购买时间+支付金额+经手人
会员顾客销售表	记录会员顾客销售信息	小票编号+会员编号+购买时间+支付金额+经手人+获得积分

## 1.4 功能需求分析

### 1.4.1 管理员功能模块

根据需求，管理员登录成功后，可以对商品进行管理，包括对商品的添加，删除，修改商品信息和查询商品信息，便于管理员快速了解产库中现有的商品库存量，以及每一种商品的信息，为了方便管理员能够批量进行添加和删除商品，提高工作效率，系统提供批量上传和删除的功能。

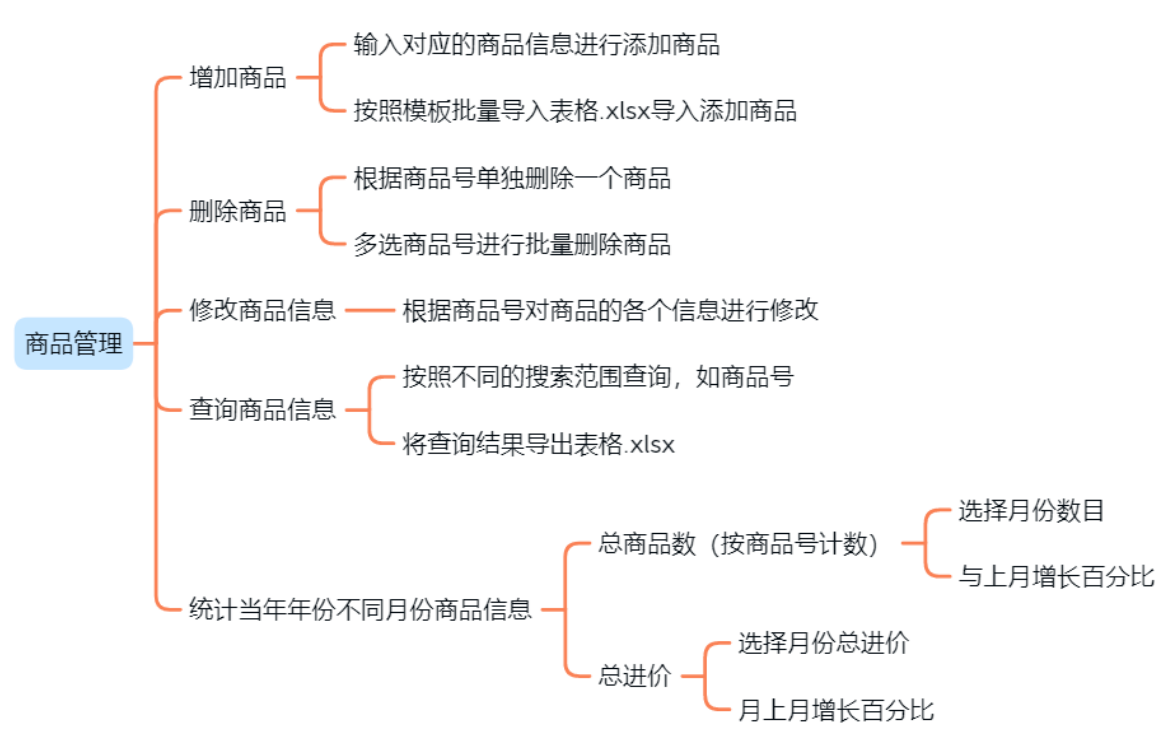


图 8 管理员功能模块

对于员工管理方面，管理员需要对超市的所有员工进行管理，包括员工入职，辞退，修改和查询员工信息等功能，同样为了提高工作效率，系统提供了批量导入和删除员工的信息。

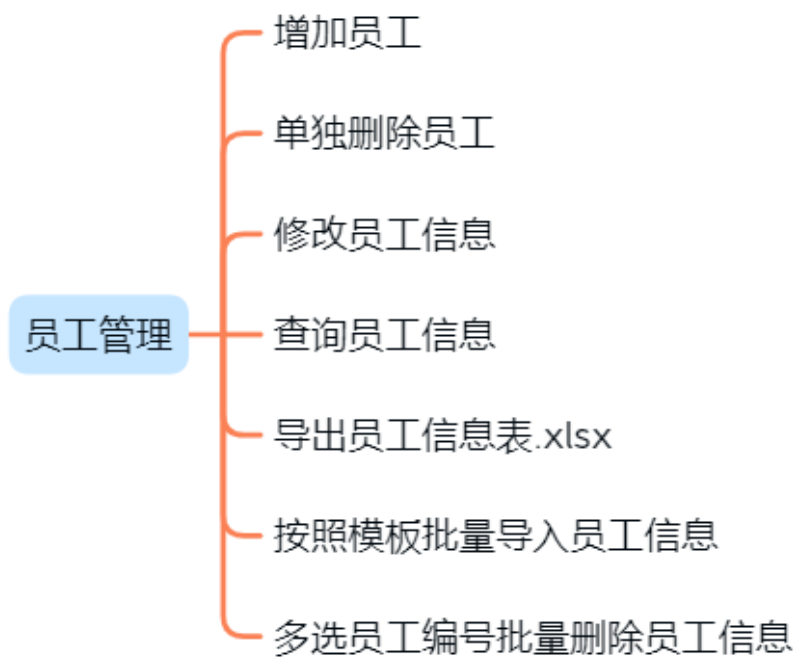


图 9 员工管理功能

管理员为了了解超市的经营情况，还需要对销售情况进行财务统计，系统支持的财务统计包括员工在某段时间内营业收入总额，某段时间内售出商品类别数量的百分比，某段时间内超市产生的利润额，某段时间内超市的营业收入，成本和利润等财务指标，除此之外还提供了查询某年份每个月份超市营业利润的变化情况。

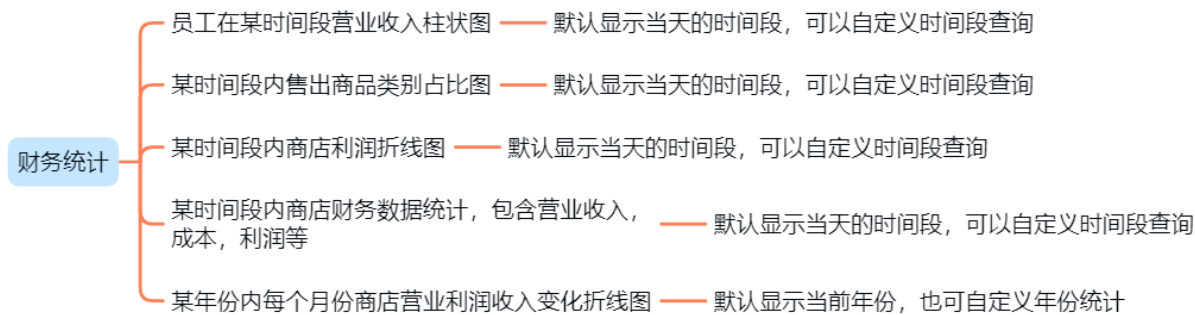


图 10 财务统计功能

为了便于超市对于商品规格，商品类别和员工职位名称的统一管理，该系统提供了管理员对商品规格，商品类别和职位名称的增删改操作，在此基础上，员工对于商品信息和员工信息的管理可以直接进行选择类别，规格和职位名称，形成超市内部统一标准。



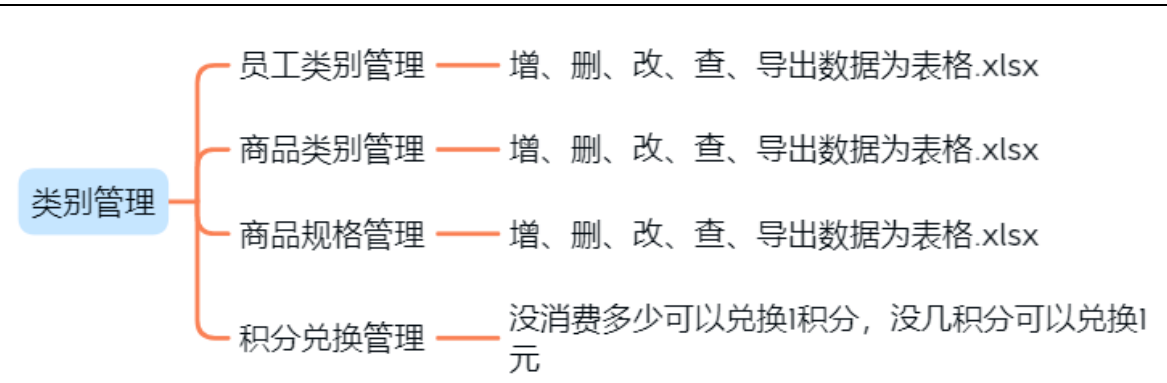


图 11 类别管理功能

管理员为了能够了解超市的销售情况已经每位员工的销售情况，系统提供了管理员进行销售记录管理的功能，其中包括根据小票编号等信息进行搜索得到小票的先关信息，包括购买时间，购买的商品名称，数量，价格等，同时支持将数据导出为表格便于保存。除此之外，管理员可以选择单独的以为员工查看对应的销售记录，便于管理员对员工的业绩考核等。

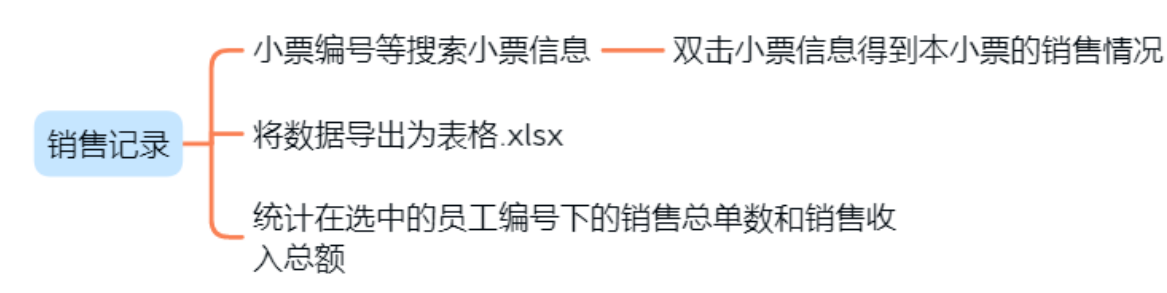


图 12 销售记录功能

1.4.2 收银员功能模块

收银员的核心业务是对顾客购买的商品进行收银并结账，收银员可以扫描商品编号后系统自动显示商品相关信息，如果商品编号不易识别，收银员可以手动输入商品编号或者查询商品编号信息双击录入。如果顾客是会员，收银员还可以查看会员的相关信息，双击表格后将会员编号显示在收银界面中，便于收银员的操作，提高效率。

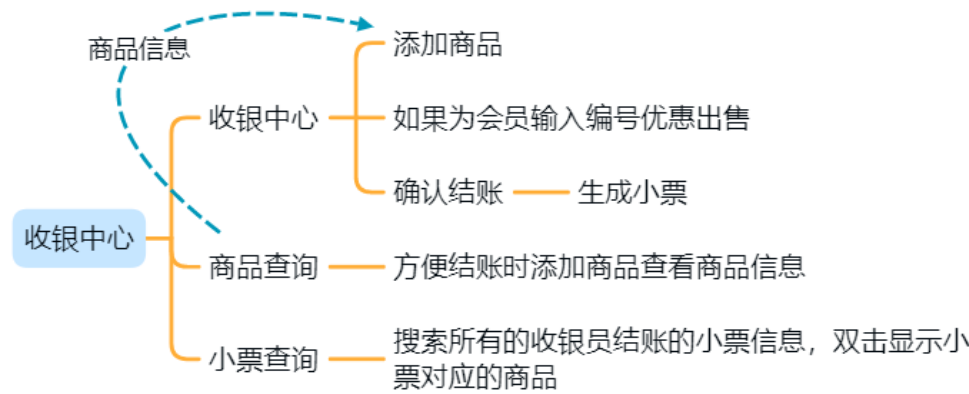


图 13 收银中心功能

会员注册和延期等业务属于收银员的范畴，收银员可以对会员进行添加，删除，修改，查询，进行积分兑换和会员续费等，同时双击会员信息表可以将会员编号显示在收益界面中，提高收银员收银结账的效率。

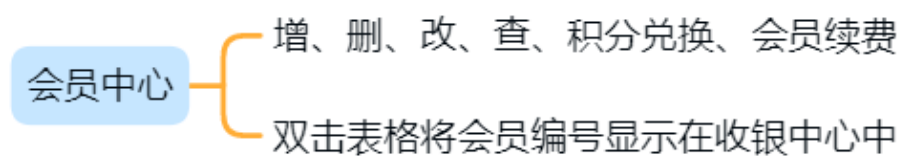


图 14 会员中心功能

收银员可以在个人中心中修改和查看自己的个人信息以及个人的销售记录。

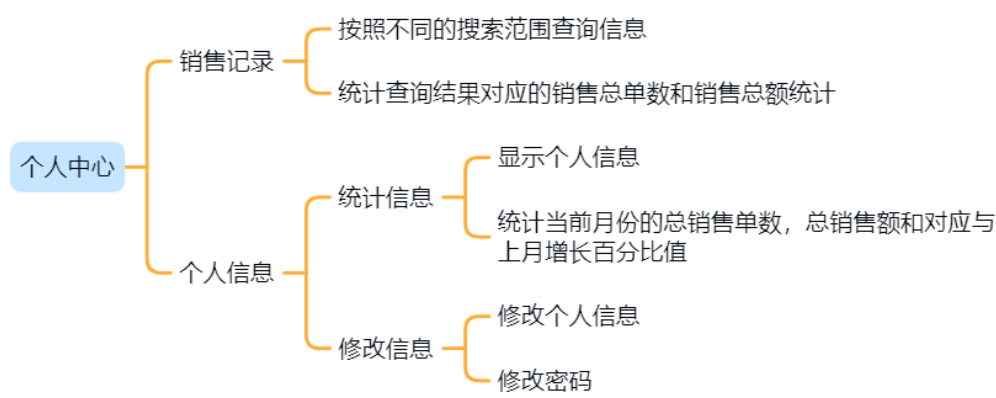


图 15 个人中心功能

## 2.数据库概念模型设计

### 2.1 分 ER 图

该系统实现的主要功能是收银员进行收银结账、对会员进行注册注销等、对员工自己的销售记录进行统计管理。超市管理员进行对货物的管理，包括进货，定价，清除库存等、添加以及删除员工、对超市总体和每一位收银员的销售记录进行管理。

附加的功能包括会员积分兑换、地址的关联选择即选择某省份则城市中只显示该省份的城市、记录员工登录注销时间等。

根据以上分析，该系统的主实体可以包括商品，员工，会员顾客，普通顾客，供应商，规格，类别，职位和小票。

#### 2.1.1 每个实体的 ER 图

##### ①商品

商品的属性包括商品编号、商品名称、普通售价、会员售价、会员折扣。

商品的分 ER 图如下：



图 16 商品实体

②员工商品的属性包括员工编号、员工姓名、登录密码、员工住址、详细地址、联系电话、出生日期、员工性别、基本工资。

员工的分 ER 图如下：



图 17 员工实体

③会员顾客的属性包括会员编号、会员姓名、会员性别、联系电话、注册日期、到期日期、现有积分、会员状态。

会员顾客的分 ER 图如下：



图 18 会员顾客实体

④普通顾客的属性包括顾客编号。

普通顾客的分 ER 图如下：



图 19 普通顾客实体

⑤供应商的属性包括供应商名称、生产地址。

供应商的分 ER 图如下：

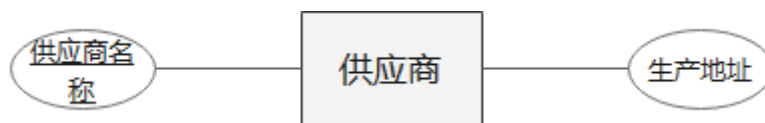


图 20 供应商实体

⑥规格的属性包括规格名称。

规格的分 ER 图如下：



图 21 规格实体

⑦类别的属性包括类别名称。

类别的分 ER 图如下：



图 22 类别实体

⑧职位的属性包括职位名称。

职位的分 ER 图如下：



图 23 职位实体

⑨小票的属性包括小票编号。

小票的分 ER 图如下：

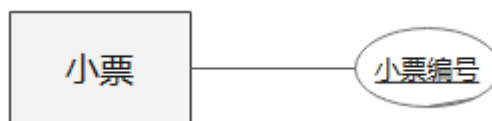


图 24 小票实体

2.1.2 主实体间联系

①商品-会员顾客-员工（收银员）之间的联系

会员顾客在购买完商品时，由一位收银员进行计算结账操作，并且结账时产生小票，记录积分，购买时间等信息。并且员工在上下班即登录和退出登录时记录时间。一张小票上记录着多种商品，并且一种商品可以在多张小票上记录。一个员工可以对多个顾客结账，生成多张小票。得到的联系图如下：

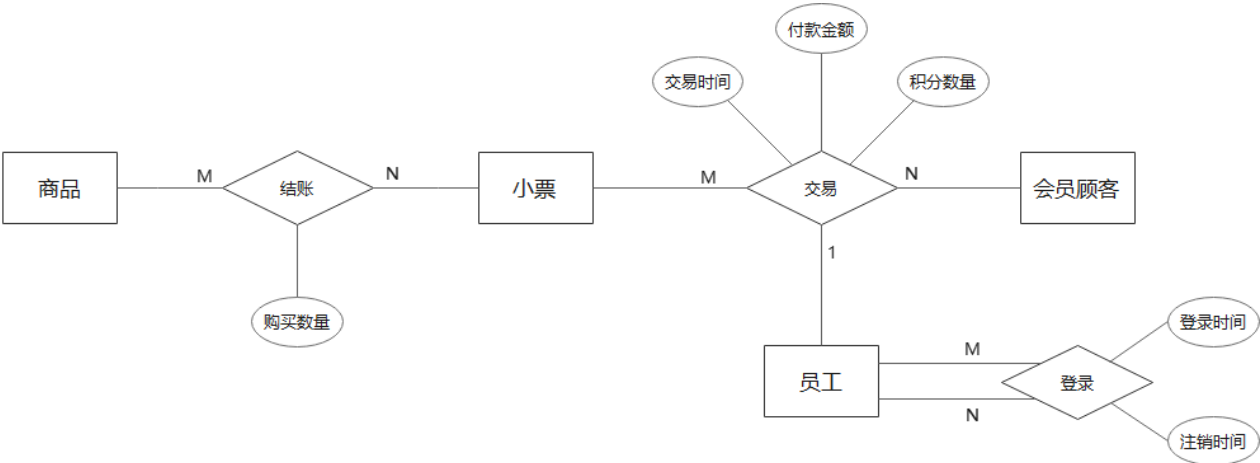


图 25 商品-会员顾客-员工（收银员）之间的联系

②商品-普通顾客-员工（收银员）之间的联系

普通顾客在购买完商品时，由一位收银员进行计算结账操作，并且结账时产生小票，记录购买时间等信息。并且员工在上下班即登录和退出登录时记录时间。一张小票上记录着多种商品，并且一种商品可以在多张小票上记录。一个员工可以对多个顾客结账，生成多张小票。得到的联系图如下：

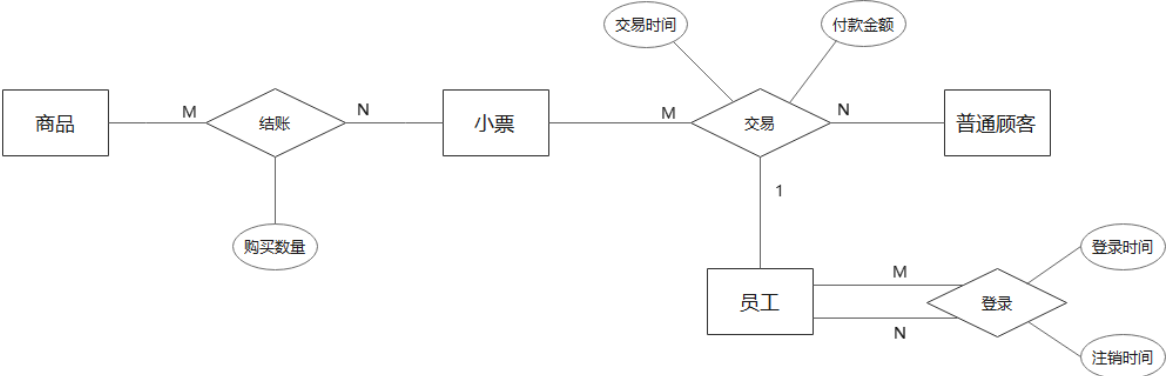


图 26 商品-普通顾客-员工（收银员）之间的联系

### ③商品-供应商之间的联系

超市向供应商购买商品，即进货时，供应商提供商品，并且之间产生进货时间，进货数量，进货价格等属性。一个供应商可以提供多种商品，并且每一种商品只能由一个供应商提供。得到的联系图如下：



图 27 商品-供应商之间的联系

### ④类别-商品-规格之间的联系

商品具有商品类别和商品规格（计量单位）的属性，所以一个商品对应一种规格和类别，但每一种类别或规格被多种商品使用。得到的联系图如下：

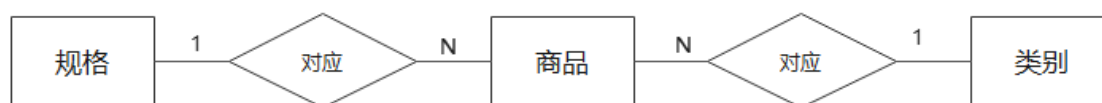


图 28 类别-商品-规格之间的联系

### ⑤职位-员工之间的联系

员工具有职位的属性，每一位员工对应一种职位，每一种职位被多个员工使用。得到的联系图如下：



图 29 职位-员工之间的联系





②员工表

员工编号	员工姓名	密码	住址	详细地址	联系电话	出生日期	性别	基本工资
主码								

③会员顾客表

会员编号	会员姓名	会员性别	联系电话	注册日期	到期日期	会员状态	积分数量
主码							

④普通顾客表

顾客编号
主码

⑦商品类别表

商品类别名称
主码

⑤供应商表

供应商名称	生产地址
主码	

⑧员工职位表

员工职位名称
主码

⑥商品规格表

商品规格名称
主码

⑨小票表

小票编号
主码

(2) 将 1:N 的关系中把 1 的一端的主码放置在 N 的一端中, 作为 N 的一端的外码, 并且把关系的属性放置在 N 的一端中。更新后得到的关系表如下:

①商品表

商品编号	商品名称	普通售价	会员折扣	会员售价	供应商名称	进货数量	进货价格	进货日期	规格名称	类别名称
主码										
					外码				外码	外码

②员工表

员工编号	员工姓名	密码	住址	详细地址	联系电话	出生日期	性别	基本工资	职位名称
主码									
									外码

③会员顾客表

会员编号	会员姓名	会员性别	联系电话	注册日期	到期日期	会员状态	积分数量
主码							

④普通顾客表

顾客编号
主码

⑦商品类别表

商品类别名称
主码

⑤供应商表

供应商名称	生产地址
主码	

⑧员工职位表

员工职位名称
主码

⑥商品规格表

商品规格名称
主码

⑨小票表

小票编号
主码

(3) 对于 M:N 的关系中，需要将两端的实体的主码和关系的属性作为一个关系，并且两端的主码作为关系的复合主码和外码。

更新后等到的基本表如下：

1.商品信息表

商品编号	商品名称	普通售价	会员折扣	会员售价	供应商名称	进货数量	进货价格	进货日期	规格名称	类别名称
主码										
					外码				外码	外码

2.员工信息表

员工编号	员工姓名	密码	住址	详细地址	联系电话	出生日期	性别	基本工资	职位名称
主码									
									外码

3.会员顾客信息表

会员编号	会员姓名	会员性别	联系电话	注册日期	到期日期	会员状态	积分数量
主码							

4.会员顾客销售记录表

小票编号	会员编号	员工编号	交易时间	付款金额	积分数量
主码（符复合主码）					
外码	外码	外码			

5.普通顾客销售记录表

小票编号	顾客编号	员工编号	交易时间	付款金额
主码（符复合主码）				
外码	外码	外码		

6.小票信息表

小票编号	商品编号	购买数量
主码（复合主码）		
外码	外码	

7.员工登录信息表

员工编号	登录时间	注销时间
主码（复合主码）		
外码		

8.供应商表

供应商名称	生产地址
主码	

10.商品类别表

商品类别名称
主码

9.商品规格表

商品规格名称
主码

11.员工职位表

员工职位名称
主码

12.小票表

小票编号
主码

13.普通顾客表

顾客编号
主码

### 3.1.2 基本表优化

经检查，所有的基本表均不存在非主属性对主属性的传递函数依赖和部分函数依赖，因此均满足 3NF 的要求。

由于普通顾客编号是结账完成后根据一定的规则由系统随机产生的唯一编号，且不需对其进行删除和修改的操作，因此普通顾客表（顾客编号）可以剔除，并且更新普通顾客销售信息表。

### 3.1.3 最终基本表

根据以上的转化以及优化之后得到的基本表如下所示：

1.商品信息表

商品编号	商品名称	普通售价	会员折扣	会员售价	供应商名称	进货数量	进货价格	进货日期	规格名称	类别名称
主码										
					外码				外码	外码

2.员工信息表

员工编号	员工姓名	密码	住址	详细地址	联系电话	出生日期	性别	基本工资	职位名称
主码									
									外码

3.会员顾客信息表

会员编号	会员姓名	会员性别	联系电话	注册日期	到期日期	会员状态	积分数量
主码							

4.会员顾客销售记录表

小票编号	会员编号	员工编号	交易时间	付款金额	积分数量
主码（符复合主码）					
外码	外码	外码			

5.普通顾客销售记录表

小票编号	员工编号	顾客编号	交易时间	付款金额
主码（复合主码）				
外码	外码			

6.小票信息表

小票编号	商品编号	购买数量
主码（复合主码）		
外码	外码	

7.员工登录信息表

员工编号	登录时间	注销时间
主码（复合主码）		
外码		

8.供应商表

供应商名称	生产地址
主码	

9.商品规格表

商品规格名称
主码

11.员工职位表

员工职位名称
主码

10.商品类别表

商品类别名称
主码

12.小票表

小票编号
主码

### 3.2 具体逻辑模型转化

将转化并优化完成的基本表中表名和字段名用英文表示，并且添加数据类型，数据长度和取值范围等行。完整表述一个表结构的定义。

## 1)商品信息表（Goods）

字段名	商品编号	商品名称	普通售价	会员折扣	会员售价	供应商名称	进货数量	进货价格	进货日期	规格名称	类别名称
主码	Primary key										
外码						Foreign key				Foreign key	Foreign key
英文名	gid	gname	gpprice	gdiscount	gvprice	gsupplier	gnum	gpurchaseprice	gindete	specifications	gcategory
类型	char	char	double	double	double	char	double	double	date	char	char
长度	16	32	(6,2) <sup>①</sup>	(3,2)	(6,2)	32	6	(6,2)	3	16	16
取值范围			0.00-9999.99	0.00-1.00	0.00-9999.99			0.00-9999.99			

注：①6 为长度，2 位小数位数，下同。

## 2)员工信息表（Employee）

字段名	员工编号	员工姓名	密码	住址	详细地址	联系电话	出生日期	性别	基本工资	职位名称
主码	Primary key									
外码										Foreign key
英文名	eid	ename	epassword	eaddress	edaddress	ephone	ebirthday	esex	esalay	Erole
类型	char	char	char	char	char	char	date	char	double	char
长度	10	8	20	64	64	12	3	4	(7,2)	16
取值范围									0.00-99999.99	

## 3)会员信息表 (Vipcustomer)

字段名	会员编号	会员姓名	会员性别	联系电话	注册日期	到期日期	会员状态	积分数量
主码	Primary key							
外码								
英文名	vid	vname	vsex	vphone	vrdate	vodate	vstate	vintegral
类型	char	char	char	char	date	date	char	int
长度	15	10	4	12	3	3	8	
取值范围								00000000-99999999

## 4)会员销售记录表 (Vsales)

字段名	小票编号	会员编号	员工编号	交易时间	付款金额	积分数量
主码	Primary key					
外码	Foreign key	Foreign key	Foreign key			
英文名	tid	vid	eid	sdate	scash	sintegral
类型	char	char	char	datetime	double	int
长度	12	15	10	8	(8,2)	
取值范围					0.00-999999.99	

## 5)普通顾客销售记录表 (Psales)

字段名	小票编号	员工编号	顾客编号	交易时间	付款金额
主码	Primary key				
外码	Foreign key	Foreign key			
英文名	tid	eid	pid	pdate	pcash
类型	char	char	char	datetime	double
长度	12	10	15	8	(8,2)
取值范围					0.00-999999.99

## 6)员工登录记录表 (Insign)

字段名	员工编号	登录时间	注销时间
主码	Primary key		
外码	Foreign key		
英文名	eid	idate	ioutdate
类型	char	datetime	datetime
长度	10	8	8

7) 供应商信息表 (Origin)

字段名	供应商名称	生产地址
主码	Primary key	
外码		
英文名	gsupplier	gorigin
类型	char	char
长度	32	32

8) 小票信息表 (Ticket)

字段名	小票编号	商品编号	购买数量
主码	Primary key		
外码	Foreign key	Foreign key	
英文名	tid	gid	tnum
类型	char	char	double
长度	12	16	(6,2)
取值范围			0.00-9999.99

9) 商品规格表 (Reficition)

字段名	商品规格名称
主码	Primary key
外码	
英文名	specifications
类型	char
长度	16

11) 员工职位表 (Employeeec)

字段名	员工职位名称
主码	Primary key
外码	
英文名	erole
类型	char
长度	16

10) 商品类别表 (Goodsc)

字段名	商品类别名称
主码	Primary key
外码	
英文名	gcategory
类型	char
长度	16

12) 小票表 (Ticketc)

字段名	小票编号
主码	Primary key
外码	
英文名	tid
类型	char
长度	12



### 3.3 视图设计

#### 1.查看员工信息视图（employee\_view）

将员工信息中的员工编号，员工姓名，住址，详细地址，联系电话，性别，职位名称作为一个视图，即

员工信息视图（员工编号，员工姓名，住址，详细地址，联系电话，性别，职位名称）

该视图作用：为了方便其他用户对员工信息的查询，但是隐藏员工的登录密码和基本工资等敏感信息，保证员工信息安全。

#### 2.员工会员销售记录视图（evs\_view）

字段名	商品编号	员工编号	会员顾客编号	小票编号
来源	Goods 表的 gid	Employee 表的 eid	Vsales 表的 vid	Ticket 表的 tid

该视图作用：该视图将商品编号，员工编号和会员顾客编号连接起来，可以得到某一位员工对会员顾客销售的记录进行查询，统计，便于管理员对员工销售记录的管理。

#### 3.员工普通顾客销售记录视图（eps\_view）

字段名	商品编号	员工编号	普通顾客编号	小票编号
来源	Goods 表的 gid	Employee 表的 eid	Psales 表的 pid	Ticket 表的 tid

该视图作用：该视图将商品编号，员工编号和普通顾客编号连接起来，可以得到某一位员工对普通顾客销售的记录进行查询，统计，便于管理员对员工销售记录的管理。

## 4.数据库物理设计和数据库保护性设计

### 4.1 索引设计

#### 1)商品信息表（Goods）

字段名	索引类型	作用
gid	唯一索引	主码，唯一索引
gcategory	聚簇索引	外码
gsupplier	聚簇索引	外码
specifications	聚簇索引	外码

#### 2)员工信息表（Employee）

字段名	索引类型	作用
eid	唯一索引	主码，唯一索引
erole	聚簇索引	外码
epassword	聚簇索引	便于查询

#### 3)会员信息表（Vipcustomer）

字段名	索引类型	作用
vid	唯一索引	主码，唯一索引
vodate	聚簇索引	便于查询

## 4)会员销售记录表（Vsales）

字段名	索引类型	作用
eid	聚簇索引	外码
vid	聚簇索引	外码

## 5)普通顾客销售记录表（Psales）

字段名	索引类型	作用
eid	聚簇索引	外码

## 6)员工登录记录表（Insign）

字段名	索引类型	作用
eid	聚簇索引	外码

## 7)供应商信息表（Origin）

字段名	索引类型	作用
gsupplier	唯一索引	主码，唯一索引

## 8)小票信息表（Ticket）

字段名	索引类型	作用
gid	聚簇索引	外码

## 9)商品规格表 (Reficition)

字段名	索引类型	作用
specifications	唯一索引	主码，唯一索引

## 10)商品类别表 (Goodsc)

字段名	索引类型	作用
gcategory	唯一索引	主码，唯一索引

## 11)员工职位表 (Employeeec)

字段名	索引类型	作用
erole	唯一索引	主码，唯一索引

## 12)小票表 (Ticketc)

字段名	索引类型	作用
tid	唯一索引	主码，唯一索引

4.2 表间关联设计

每个表根据外键连接的关系如下图所示（忽略所有属性）：

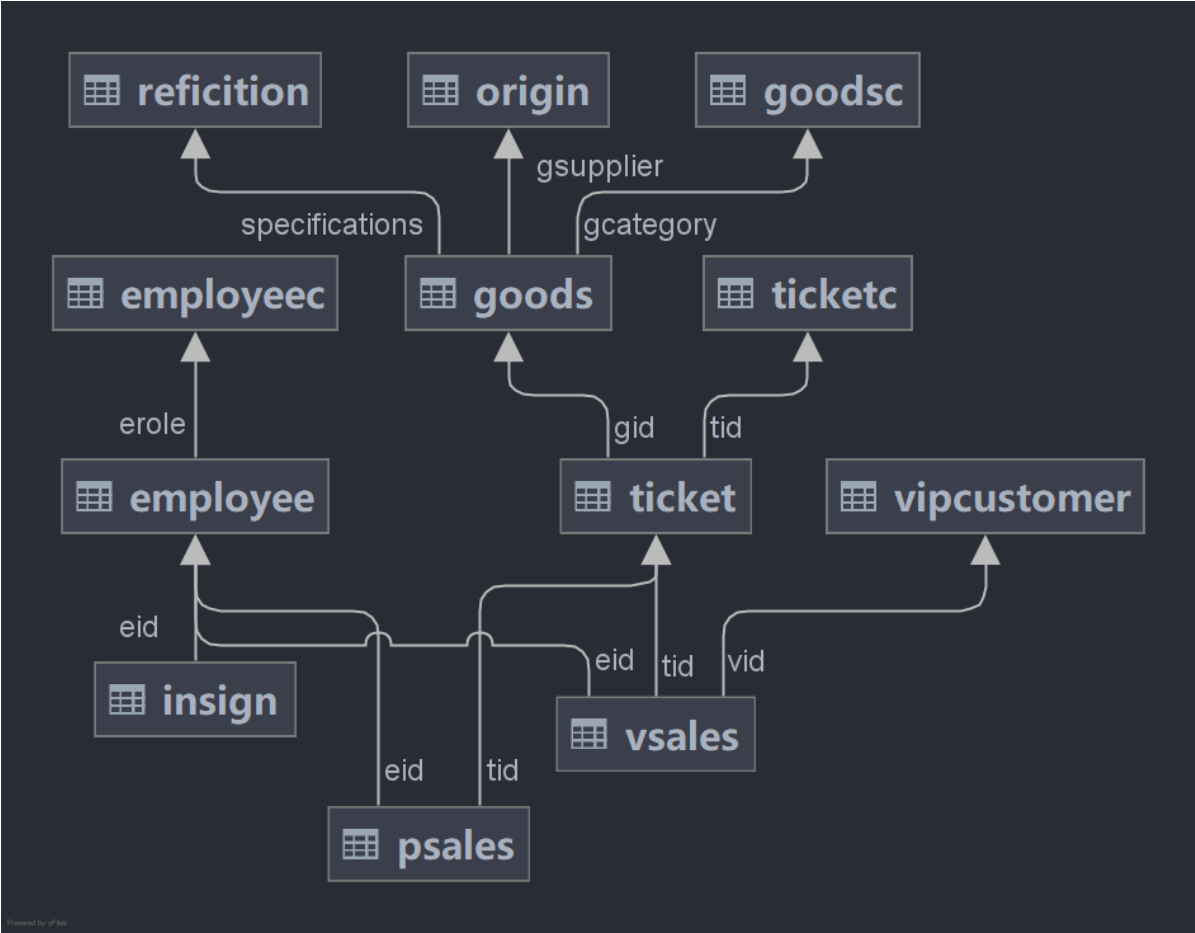


图 31 表格间关系图

4.3 完整性设计

1)商品信息表（Goods）

列名	gid (商品编号)	gname (名称)	gpprice (普通售价)	gdiscount (会员折扣)	gvprice (会员售价)	gsupplier (供应商)	gpurchase-price (进价)	specificati- ons(类别)	gcategory (规格)
实体完整性	primary key								
参照完整性						参照供应 商信息表 gsupplier		参照商品 类别表的 gcategory	参照商品 规格表的 specificati ons

用户自定义完整性		not null	最多保留两位小数，在 0.00-9999.99 间取值	最多保留两位小数，在 0.00-1.00 间取值	最多保留两位小数，在 0.00-9999.99 间取值		最多保留两位小数，在 0.00-9999.99 间取值		
----------	--	----------	-----------------------------	--------------------------	-----------------------------	--	-----------------------------	--	--

2)员工信息表 (Employee)

列名	eid(员工编号)	ename(姓名)	esex(性别)	esalay(基本工资)	erole(职位)
实体完整性	primary key				
参照完整性					参照员工职位表的 vid
用户自定义完整性		not null	只取‘男’或‘女’	最多保留两位小数，在 0.00-99999.99 间取值	

3)会员信息表 (Vipcustomer)

列名	vid(会员编号)	vname(姓名)	vsex(性别)	vstate(会员状态)
实体完整性	primary key			
参照完整性				
用户自定义完整性		not null	只取‘男’或‘女’	只取‘未到期’或‘已到期’

4)会员销售记录表 (Vsales)

列名	tid(小票编号)	eid(员工编号)	vid(会员编号)	scash(付款金额)
实体完整性	primary key			
参照完整性	参照小票表的 tid	参照员工信息表的 eid	参照会员信息表的 vid	
用户自定义完整性				最多保留两位小数，在 0.00-999999.99 间取值

## 5)普通顾客销售记录表 (Pssales)

列名	tid(小票编号)	eid(员工编号)	pcash(付款金额)
实体完整性	primary key		
参照完整性	参照小票表的 tid	参照员工信息表的 eid	
用户自定义完整性			最多保留两位小数， 在 0.00-999999.99 间 取值

## 6)员工登录记录表 (Insign)

列名	eid(员工编号)	idate(登录时间)	ioutdate(注销时间)
实体完整性	primary key		
参照完整性	参照员工信息表的 eid		
用户自定义完整性			

## 7)供应商信息表 (Origin)

列名	gsupplier(供应商名称)	gorigin(生产地址)
实体完整性	primary key	
参照完整性		
用户自定义完整性		not null

## 8)小票信息表 (Ticket)

列名	tid(小票编号)	gid(商品编号)	tnum(购买数量)
实体完整性	primary key		
参照完整性	参照小票表的 tid	参照商品信息表的 gid	
用户自定义完整性	在 000000000000- 999999999999 之间取 值		最多保留两位小数， 在 0.00-9999.99 间取 值

9)商品规格表（Reficition）		11)员工职位表（Employeeec）	
列名	specifications(商品规格名称)	列名	erole(职位名称)
实体完整性	primary key	实体完整性	primary key
参照完整性		参照完整性	
用户自定义完整性		用户自定义完整性	

10)商品类别表（Goodsc）		12)小票表（Ticketc）	
列名	gcategory(商品类别名称)	列名	tid(小票编号)
实体完整性	primary key	实体完整性	primary key
参照完整性		参照完整性	
用户自定义完整性		用户自定义完整性	在 0000000000000-999999999999 之间取值

## 5.处理功能设计

### 5.1 主控模块设计

环境初始化、用户身份验证页面



图 32 环境初始化、用户身份验证页面



管理员主页面

Window

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

查询 添加 删除 修改 统计

搜索范围 请选择 搜索内容 确认 重置 导出数据

商品号	商品名	类别	产地	供货商	进货日期	现有库存量	规格	进价 (元)	零售价 (元)	会员价 (...)	会员折扣 *
表中无内容											

图 33 管理员主页面

收银员主页面

Window

收银中心 会员中心 个人中心 重新登录 注销

收银中心 商品查询 小票查询

数量合计 总价合计 实付金额 支付金额 找零

商品编号 商品编号 数量 购买数量 添加 删 ... 清空列表

是否为会员 否 是 会员编号 会员编号 重置 确认结账

小票编号	商品编号	商品名	类别	单价	数量	总价	+
表中无内容							

图 34 收银员主页面

告别页面、系统退出页面

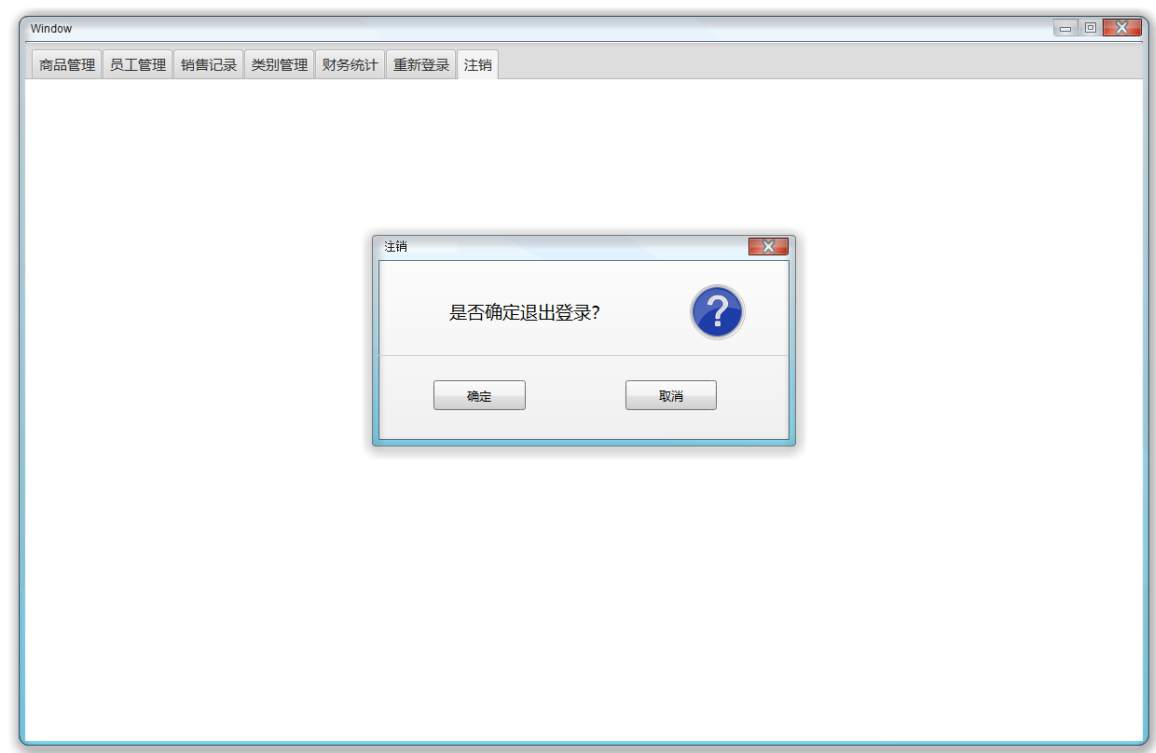


图 35 告别页面、系统退出页面

5.2 子模块设计

重置密码页面

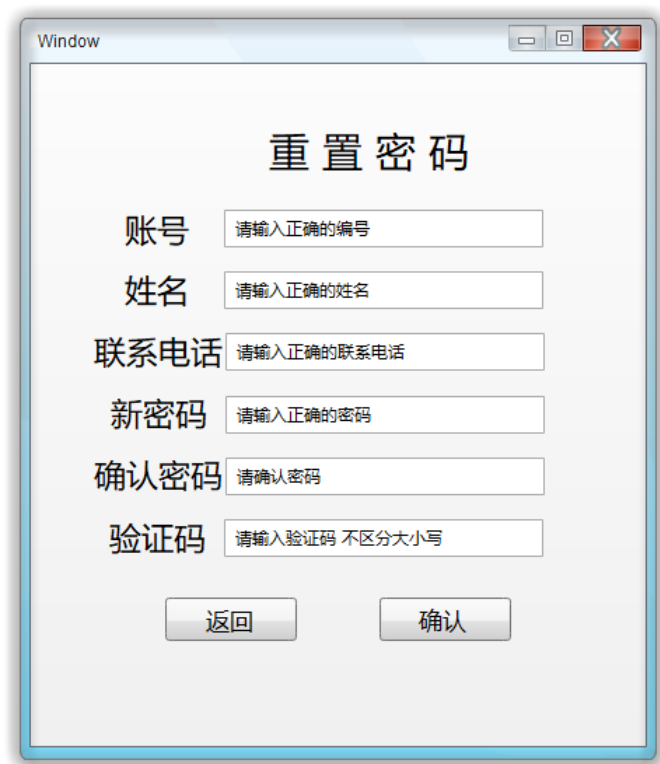


图 36 重置密码页面

员工管理页面

Window

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

操作类型 请...

员工编号 员工编号

员工姓名 员工姓名

性别 ☒ 男 ☐ 女

职位 请选...

刷新

住址 省份 城市 县区

详细地址

基本工资 基本工资

搜索范围 请选择

查询内容

确认

导出数据

批量添加

批量删除

员工编号	员工姓名	员工地址		联系电话	年龄	性别	基本工资 (元)	职位	+
		地址	详细地址						
表中无内容									

图 37 员工管理页面

销售记录页面

Window

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

搜索范围 请选择

搜索内容 搜索内容

确认

重置

导出数据

员工编号 员工编号

销售总单数

销售总额

刷新

小票编号	顾客编号	经手人	购买时间	支付现金	获得积分 +	小票编号	商品编号	商品名称	单价	数量	规格	总价 +
表中无内容						表中无内容						

图 38 销售记录页面

类别管理页面

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

员工类别

表中无内容

商品类别

表中无内容

规格

表中无内容

目前每消费 元兑换 1 积分，每 积分兑换 1 元

每消费  元兑换 1 积分，每  积分兑换 1 元

表格  操作对象

操作类型

图 39 类别管理页面

财务统计页面

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

统计时间 (注: 选择相同日期时表示统计一天) 从  到

第一页 第二页 第三页

单位: 元

110 100 90 80 70 60 50 40 30 20 10 0

日期	利润额 (元)
表中无内容	

总营业收入额  与上月同期相比增长

总销售成本额  与上月同期相比增长

总营业利润额  与上月同期相比增长

平均每天利润额  与上月同期相比增长

图 40 财务统计页面

会员中心页面

Window

收银中心会员中心个人中心重新登录注销

操作类型

请选择

会员编号

姓名

姓名

性别

☒男☐女

续费时长

请选择

月

联系电话

积分兑换

积分

重置

搜索范围

请选择

搜索内容

确认

会员编号	姓名	性别	联系电话	注册时间	到期时间	现有积分	状态	+
表中无内容								

图 41 会员中心页面

个人中心页面

Window

收银中心会员中心个人中心重新登录注销

销售记录个人信息

搜索范围

请选择

搜索内容

搜索内容

确认

重置

销售总单数

销售总额

小票编号	顾客编号	购买时间	支付现金	获得积分	+	小票编号	商品编号	商品名称	单价	数量	总价	+
表中无内容						表中无内容						

图 42 个人中心页面

42

## 5.3 查询设计

### ①商品数量查询

业务需求：统计某一件商品（使用商品编号标识）在某段时间内进货的数量

涉及表：商品信息表（Goods）

查询类型：精确查询

查询语句<sup>1</sup>：

注 1：查询语句中问号(?)为占位符，即查询时需要输入的数据，下同

```
select count(distinct gid)
from goods
where date_format(gindate, '%Y-%m-%d') >= date_format(?, '%Y-%m-%d')
      and date_format(gindate, '%Y-%m-%d') <= date_format(?, '%Y-%m-%d')
      and gid = ?
```

### ②员工信息查询

业务需求：根据员工编号查询员工的个人信息

涉及表：员工信息表（Employee）

查询类型：模糊查询

查询语句：

```
select eid,ename,erole,eaddress,edaddress,ephone,esex,
       round(timestampdiff(year, ebirthday, now())) eage,
       ebirthday
from employee
where eid like concat('%',?, '%')
```

### ③会员销售查询

业务需求：根据会员编号模糊查询会员销售的总金额

涉及表：商品信息表（Goods）小票信息表（Ticket）会员销售记录表（Vsales）

查询类型：模糊查询

查询语句:

```
select sum(goods.gvprice * ticket.tnum)
from goods,ticket,vsales
where goods.gid = ticket.gid
      and vsales.tid = ticket.tid
      and vid like concat('%', ?, '%')
```

#### ④查询售出商品类别占比

业务需求: 查询在某段时间内所有的销售商品类别数量以及占总数的比例

涉及表: 商品信息表 (Goods) 小票信息表 (Ticket) 会员销售记录表 (Vsales)

普通销售记录表 (Psales)

查询类型: 精确查询

查询语句:

```
select gcategory, concat(sum(percentage1), '%') percentage, sum(sum1) sum
from (select gcategory, format((sum(tnum) / ?) * 100, 2) percentage1, sum(tnum)
      sum1
      from goods,ticket,vsales
      where goods.gid = ticket.gid and ticket.tid = vsales.tid
      and sdate >= ? and sdate <= ?
      group by gcategory
      union all
      select gcategory, format((sum(tnum) / ?) * 100, 2) percentage1, sum(tnum)
      sum1
      from goods,ticket,psales
      where goods.gid = ticket.gid and ticket.tid = psales.tid
      and pdate >= ? and pdate <= ?
      group by gcategory) a
group by gcategory
```

### ⑤销售信息查询

业务需求：根据会员编号和普通顾客编号，查询销售记录

涉及表：会员销售记录表（Vsales）普通顾客销售记录表（Psales）

查询类型：模糊查询

查询语句：

```
select *  
from vsales  
where vid like concat('%', ?, '%')  
union all  
select tid, pid vid, pdate sdate, pcash scash, eid, 0 sintegral  
from psales  
where pid like concat('%', ?, '%')
```

### ⑥查询小票信息详情

业务需求：根据小票编号，查询小票详情，包括商品编号，商品名称，购买数量，  
价格等

涉及表：商品信息表（Goods）小票信息表（Ticket）

查询类型：精确查询

查询语句：

```
select tid, goods.gid, gname, gcategory, gpprice, gvprice, tnum, specifications  
from goods, ticket  
where goods.gid = ticket.gid and tid = ?
```



## 6.数据库应用系统的实现

### 6.1 数据库及其表结构的建立

#### 6.1.1 表的建立

##### 商品信息表 (Goods)

```
create table if not exists goods (  
    gid            char(16) primary key,  
    gname          char(32),  
    gcategory      char(16),  
    gsupplier      char(32),  
    gindate        date,  
    gnum           double          default 0,  
    gpurchaseprice double(6, 2) default 0.00,  
    gpprice        double(6, 2) default 0.00,  
    gvprice        double(6, 2) as ((`gpprice` * `gdiscount`)),  
    gdiscount      double(3, 2) default 0.00,  
    specifications char(16),  
    constraint goods_gid_uindex unique (gid),  
    constraint goods_goodsc_gcategory_fk foreign key (gcategory) references goodsc  
(gcategory) on update cascade on delete cascade,  
    constraint goods_origin_gsupplier_fk foreign key (gsupplier) references origin  
(gsupplier) on update cascade on delete cascade,  
    constraint goods_reficition_specifications_fk foreign key (specifications) references  
reficition (specifications) on update cascade on delete cascade,  
    check ((`gdiscount` >= 0.00) and (`gdiscount` <= 1.00))  
);
```

### 员工信息表 (Employee)

```
create table if not exists employee (  
    eid          char(10) primary key,  
    ename        char(8),  
    epassword    char(20),  
    eaddress    char(64)      default '-',  
    edaddress    char(64)      default '-',  
    ephone       char(12)      default '-',  
    ebirthday    date          default '1970-01-01',  
    erole        char(16),  
    esex         char(4),  
    esalary      double(7, 2) default 0.00,  
    constraint employee_ibfk_1 foreign key (erole) references employeec (erole) on  
update cascade on delete cascade  
);
```

### 会员信息表 (Vipcustomer)

```
create table if not exists vipcustomer (  
    vid          char(15) primary key,  
    vname        char(10),  
    vsex         char(4),  
    vphone       char(12),  
    vrdate       date,  
    vodate       date,  
    vstate       char(8),  
    vintegral    int,  
    constraint vipcustomer_vid_uindex unique (vid)  
);
```

### 会员销售记录表 (Vsales)

```
create table if not exists vsales (  
    tid      char(12),  
    vid      char(15),  
    sdate    datetime,  
    scash     double(8, 2),  
    eid      char(10),  
    sintegral int,  
    primary key (tid, vid, eid),  
    constraint sales_employee_eid_fk foreign key (eid) references employee (eid) on  
update cascade on delete cascade,  
    constraint vsales_ibfk_1 foreign key (tid) references ticket (tid) on update cascade  
on delete cascade,  
    constraint vsales_vipcustomer_vid_fk foreign key (vid) references vipcustomer  
(vid) on update cascade on delete cascade);
```

### 普通顾客销售记录表 (Psasles)

```
create table if not exists psales (  
    tid      char(12),  
    pid      char(15),  
    pdate    datetime,  
    pcash     double(8, 2),  
    eid      char(10),  
    primary key (tid, eid),  
    constraint psales_employee_eid_fk foreign key (eid) references employee (eid),  
    constraint psales_ticket_tid_fk foreign key (tid) references ticket (tid) on update  
cascade on delete cascade  
);
```

### 员工登录记录表 (Insign)

```
create table if not exists insign (  
    eid      char(10),  
    idate    datetime,  
    ioutdate datetime,  
    primary key (idate, eid),  
    constraint insign_employee_eid_fk foreign key (eid) references employee (eid) on  
update cascade on delete cascade  
);
```

### 供应商信息表 (Origin)

```
create table if not exists origin (  
    gsupplier char(32) primary key,  
    gorigin   char(32),  
    constraint origin_gsupplier_uindex unique (gsupplier)  
);
```

### 小票信息表 (Ticket)

```
create table if not exists ticket (  
    tid  char(12),  
    gid  char(16),  
    tnum double(6, 2),  
    primary key (tid, gid),  
    constraint ticket_ibfk_1 foreign key (gid) references goods (gid) on update cascade  
on delete cascade,  
    constraint ticket_ibfk_2 foreign key (tid) references ticketc (tid) on update cascade  
on delete cascade  
);
```

### 商品规格表 (Refiction)

```
create table if not exists refiction (  
    specifications char(16) primary key,  
    constraint refiction_specifications_uindex unique (specifications)  
);
```

### 商品类别表 (Goodsc)

```
create table if not exists goodsc (  
    gcategory char(16) primary key,  
    constraint goodsc_gcategory_uindex unique (gcategory)  
);
```

### 员工职位表 (Employeeec)

```
create table if not exists employeeec (  
    erole char(16) primary key,  
    constraint employeeec_erole_uindex unique (erole)  
);
```

### 小票表 (Ticketc)

```
create table if not exists ticketc (  
    tid char(12) primary key,  
    constraint ticketc_tid_uindex unique (tid)  
);
```

## 6.1.2 视图的建立

### 员工信息数据视图 (employee\_view)

```
create view employee_view as  
select eid, ename, eaddress, edaddress, ephone, esex, erole from employee
```

### 员工会员销售记录视图 (evs\_view)

```
create view evs_view as
select goods.gid, employee.eid, vsales.vid, ticket.tid
from goods,
     employee,
     vsales,
     ticket
where vsales.eid = employee.eid
     and goods.gid = ticket.gid
     and ticket.tid = vsales.tid;
```

### 员工普通顾客销售记录视图 (eps\_view)

```
create view eps_view as
select goods.gid, employee.eid, psales.pid, ticket.tid
from goods,
     employee,
     psales,
     ticket
where psales.eid = employee.eid
     and goods.gid = ticket.gid
     and ticket.tid = psales.tid;
```

## 6.1.3 建立索引<sup>①</sup>

注①：DBMS 自动创建的索引不再创建

### 商品信息表 (Goods)

```
create unique index goods_gid on goods(gid);
```

### 员工信息表 (Employee)

```
create unique index employee_eid on employee(eid);
create index employee_epassword on employee(epassword);
```

会员信息表 (Vipcustomer)

```
create unique index vipcustomer_vid on vipcustomer(vid);  
create index vipcustomer_vodate on vipcustomer(vodate);
```

会员销售记录表 (Vsales)

由 DBMS 完成创建

普通顾客销售记录表 (Psales)

由 DBMS 完成创建

员工登录记录表 (Insign)

由 DBMS 完成创建

供应商信息表 (Origin)

```
create unique index origin_gsupplier on origin(gsupplier);
```

小票信息表 (Ticket)

由 DBMS 完成创建

商品规格表 (Reficition)

```
create unique index reficition_specifications on reficition(specifications);
```

商品类别表 (Goodsc)

```
create unique index goodsc_gcategory on goodsc(gcategory);
```

员工职位表 (Employeeec)

```
create unique index employeeec_erole on employeeec(erole);
```

小票表 (Ticketc)

```
create unique index ticketc_tid on ticketc(tid);
```

## 6.2 数据输入

### 系统初始化数据录入

为了能够让系统正常运行，需要在系统初始化时录入管理员的账号密码保证可以正常进入系统，除此之外还需要初始化商品类别，员工类别，商品规格等数据

#### ①录入商品类别，员工类别和商品规格初始化数据

```
insert into goodsc value ('<空>');
```

```
insert into employeec value ('管理员'),('<空>');
```

```
insert into reficition value ('<空>');
```

#### ②录入管理员信息初始化数据

```
insert into employee(eid, epassword, ename, erole, esex) value ('1234567890', 'admin',  
'管理员', '管理员', '男');
```

### 测试数据录入

测试数据录入是为了调试系统过程中需要使用的数据，主要包括商品信息，员工信息以及会员信息

#### ①商品信息

```
INSERT INTO cashier.goods (gid, gname, gcategory, gsupplier, gindate, gnum,  
gpurchaseprice, gpprice, gdiscount, specifications) VALUES
```

```
('6723123332', '沐浴露', '日用百货', '义乌小市场', '2022-04-30', 2164, 34, 54, 0.9, '  
件'),
```

```
('6723212314', '阿瓜故事集', '书籍音像制品', '阿瓜出版社', '2022-04-12', 351, 12.9,  
30, 0.9, '件'),
```

```
('6723212315', '瓜碟', '书籍音像制品', '阿瓜出版社', '2022-04-13', 352, 32.9, 80, 0.9,  
'件'),
```

```
('6723212317', '新华字典', '书籍音像制品', '新华出版社', '2022-04-15', 354, 56, 170,  
0.9, '件'),
```

```
('6723212324', '一体式煎锅', '家用电器', '义乌小市场', '2022-04-22', 361, 213, 233,  
0.9, '件');
```



②员工信息

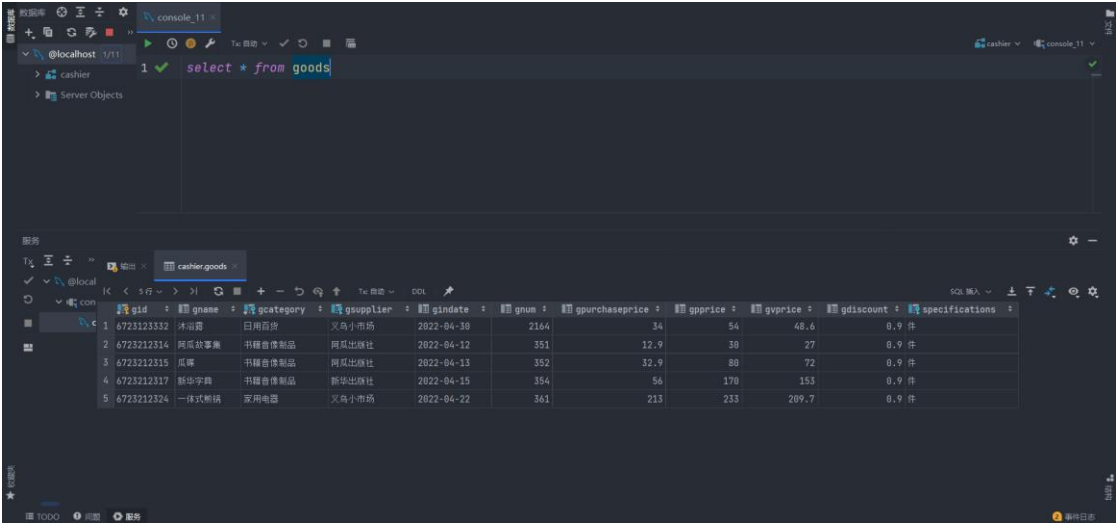
```
INSERT INTO cashier.employee (eid, ename, epassword, eaddress, erole, esex, esalary)
VALUES ('2205015198', '123', '123456', '广东省揭阳市', '收银员', '男', 3000);
```

③会员信息

```
INSERT INTO cashier.vipcustomer (vid, vname, vsex, vphone, vrdate)
VALUES ('V220510755259', 'hxb', '男', 12345678901, '2022-05-24');
```

打印表格内容

①商品信息表



gid	gname	gcategory	gsupplier	gindate	gnum	gpurchaseprice	gpprice	gvprice	gdiscount	gspecifications
1	沐浴露	日用百货	义乌小市场	2022-04-30	2164	34	54	48.6	0.9	件
2	阿瓜故事集	书籍音像制品	阿瓜出版社	2022-04-12	351	12.9	30	27	0.9	件
3	瓜藤	书籍音像制品	阿瓜出版社	2022-04-13	352	32.9	80	72	0.9	件
4	新华字典	书籍音像制品	新华出版社	2022-04-15	354	56	170	153	0.9	件
5	一体机锅	家用电器	义乌小市场	2022-04-22	361	213	233	209.7	0.9	件

图 43 打印商品信息表数据

②员工信息表

The image shows a database management interface. At the top, a SQL query editor displays the command 'select \* from employee'. Below the editor, a table of results is shown. The table has 10 columns: eid, ename, epassword, eaddress, edaddress, ephone, ebirthday, erole, esex, and esalary. There are two rows of data. The first row represents an administrator (admin) with a salary of 5000. The second row represents a cashier (收银员) with a salary of 3000. The interface includes various toolbars for navigation and editing, and a sidebar on the left showing the database structure.

图 44 打印员工信息表数据

③会员信息表

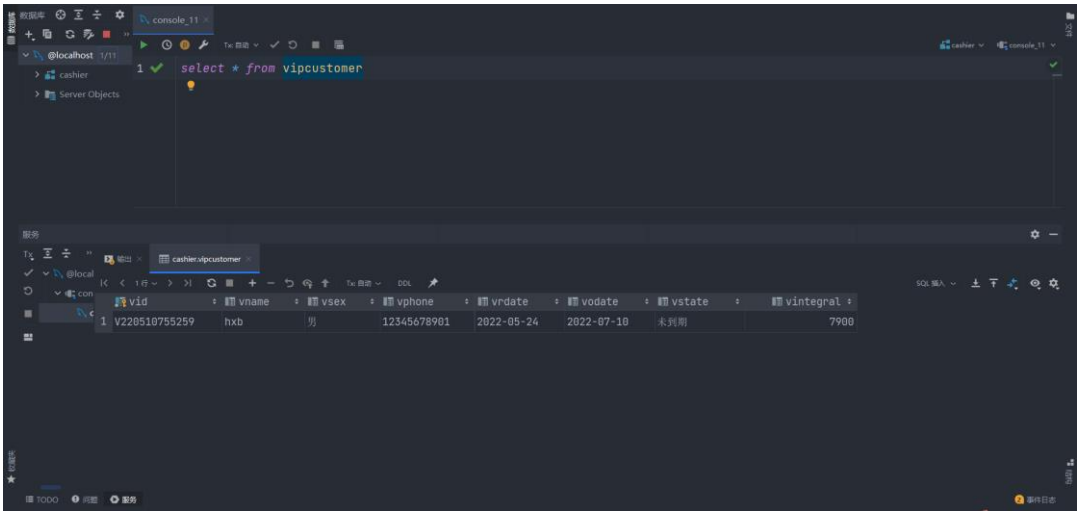


图 45 打印会员信息表数据

④商品类别表

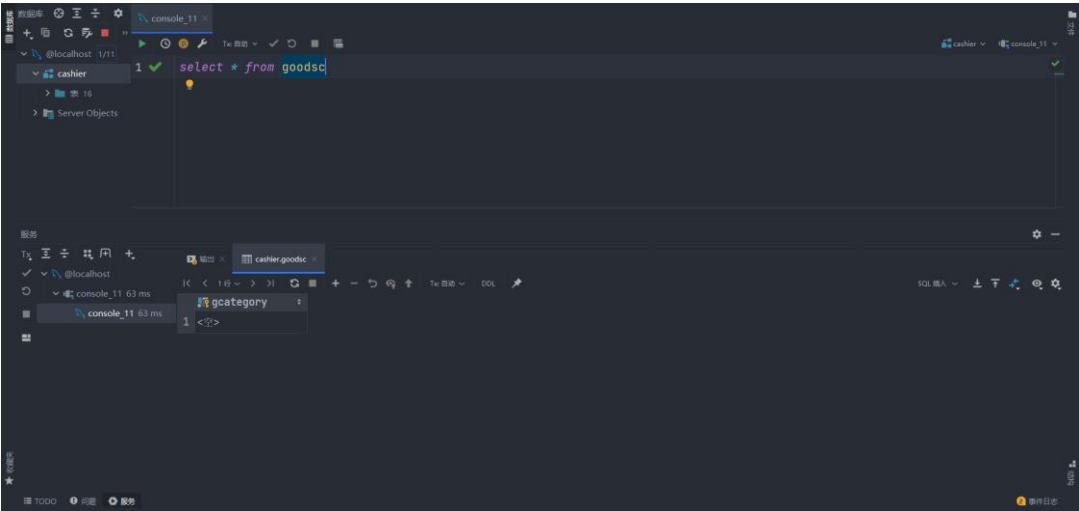


图 46 打印商品类别表数据

⑤商品规格表

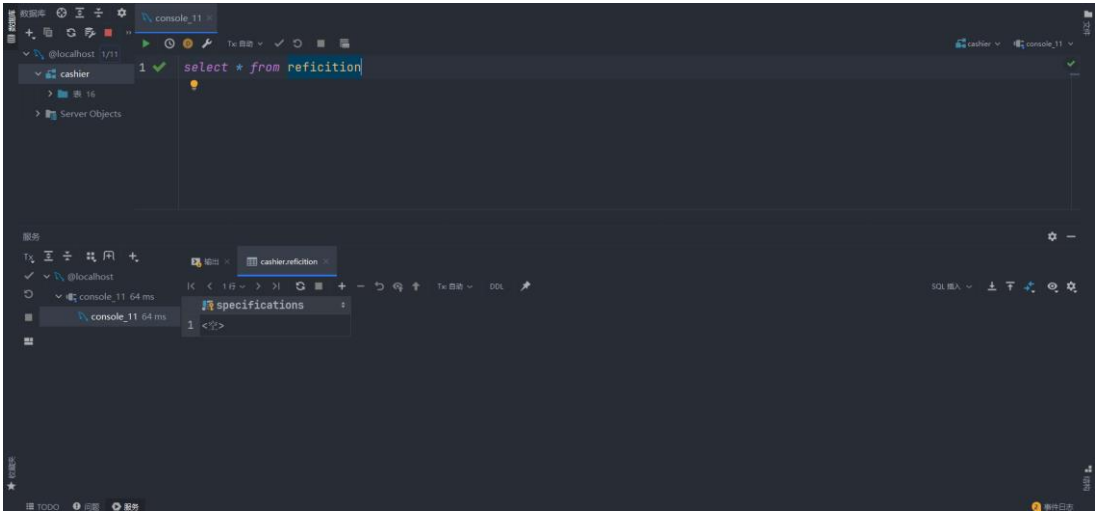


图 47 打印商品规格表数据

⑥员工类别表

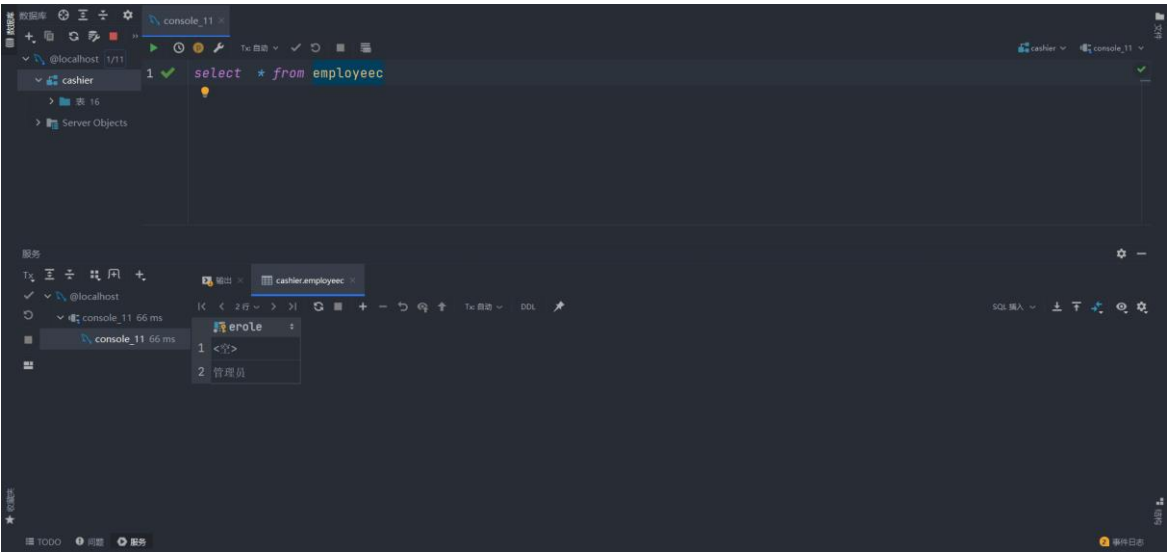


图 48 打印员工类别表数据

6.3 模块实现

数据库连接与系统启动模块

在实现此模块时，出现最大的问题是数据库连接失败，经过排查后发现配置文件的存放位置出错，导致找不到配置文件，数据库连接信息错误，故数据库连接失败，最后将配置文件存放在正确的位置方可成功连接数据库。

```
public void init() throws Exception {  
    InputStream inputStream =  
ClassLoader.getSystemClassLoader().getResourceAsStream("jdbc.properties");  
  
    Properties properties = new Properties();  
    properties.load(inputStream);  
  
    String driver = properties.getProperty("driver");  
  
    String user = properties.getProperty("user");  
  
    String password = properties.getProperty("password");  
  
    String url = properties.getProperty("url");  
  
    Class.forName(driver);  
  
    connection = DriverManager.getConnection(url, user, password);  
  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Application.launch(Loginin.class);  
    }  
}
```

### 登录模块

在实现登录模块时，出现的主要问题是数据交互问题，即登录成后用户名如何在收银员界面显示员工编号的问题，在编写中，出现较多的条件判断，原因是在用户输入错误时需要为用户提供明确的错误提示，后续可以进行优化，尽量简洁代码。

#### 登录界面启动

```
public Stage loginStage() throws Exception {  
    Stage stage = new Stage();  
    stage.close = stage;  
    FXMLLoader fxmlLoader = new  
FXMLLoader(Loginin.class.getResource("loginin.fxml"));  
    Scene scene = new Scene(fxmlLoader.load(), 380, 410);  
    stage.setScene(scene);  
    stage.setResizable(false);  
    stage.setTitle("超市管理系统");  
    return stage;  
}
```

#### 用户登录的数据库事务

```
@Override  
public List<Allentity> getEmployee(Connection connection) {  
    String sql = "select * from employee";  
    return getBeanList(connection, sql);  
}
```

@Override

```
public List<Allentity> getEmployeeEid(Connection connection, String eid) {  
    String sql = "select * from employee where eid = ?";  
    return getBeanList(connection, sql, eid);  
}
```

@Override

```
public List<Allentity> getGcategory(Connection connection) {  
    String sql = "select * from goodsc where gcategory = ?";  
    return getBeanList(connection,sql,LogininEvent.fristString);  
}
```

@Override

```
public void insertGcategory(Connection connection) {  
    String sql = "insert into goodsc(gcategory) values (?)";  
    update(connection,sql,LogininEvent.fristString);  
}
```

判断用户职位，启动对应的界面

```
Loginin loginin = new Loginin();  
if (roleadmin.isSelected() && "管理员".equals(list.get(0).getErole())) {  
    Loginin.stageclose.close();  
    gid = idtf.getText();  
    try {  
        loginin.adminStage().show();  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
} else if (roleemployee.isSelected() && "收银员".equals(list.get(0).getErole())) {  
    Loginin.stageclose.close();
```

```
Date date = new Date();

SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyy-MM-dd
HH:mm:ss");

logintime = simpleDateFormat.format(date);

eid = idtf.getText();

try {

    loginin.cashierStage().show();

} catch (Exception e) {

    e.printStackTrace();

}

} else {

    showAlert("请选择正确的角色");

}
```

### 重置密码模块

重置密码需要先判断用户输入的账号是属于管理员还是收银员，如果是管理员，姓名与联系方式的文本框将不可输入，如果是收银员，则姓名与联系方式的文本框将必须输入信息方可进行重置密码。

```
if (is) {

    if (fpasswordtf.getText().length() >= 6) {

        if (fpasswordtf.getText().equals(fpasswordtf.getText())) {

            if (fcodeable.getText().equalsIgnoreCase(fcodetf.getText())) {

                epassword = fpasswordtf.getText();

                eid = fidtf.getText();

                loginDaoImpl.updateEmployee(connection);

                Alert alert = new Alert(AlertType.INFORMATION);

                alert.setTitle("修改成功");

                alert.setHeaderText("重置密码成功！");

                Optional<ButtonType> buttonType = alert.showAndWait();
```

```
        if (buttonType.isPresent() && buttonType.get() == ButtonType.OK) {  
            Loginin.stageclose.close();  
            try {  
                loginin.loginStage().show();  
            } catch (Exception e) {  
                showAlert(e.getMessage());  
            }  
        }  
    } else {  
        showAlert("验证码错误！");  
    }  
} else {  
    showAlert("确认密码与新密码输入不同！");  
}  
} else {  
    showAlert("新密码长度小于 6 位！");  
}  
}
```

重置密码使用到的数据库事务

@Override

```
public void updateEmployee(Connection connection) {  
    String sql = "update employee set epassword = ? where eid = ?";  
    update(connection, sql, LogininEvent.epassword, LogininEvent.eid);  
}
```

## 管理员模块

实现管理员模块时，出现的主要问题是表格内容如何在切换不同的界面时不会重复加载数据，解决方案是使用一个布尔值进行标识，如果表格数据存在就不再加载数据。除此之外主要还有财务统计的数据处理以及价格输入框的数据验证，需要最多保留两位小数的格式，需要使用正则表达式进行验证。

### 管理员界面启动

```
public Stage adminStage() throws Exception {  
    Stage stage = new Stage();  
    stage.close = stage;  
    FXMLLoader fxmlLoader = new  
FXMLLoader(Loginin.class.getResource("Admin.fxml"));  
    Scene scene = new Scene(fxmlLoader.load(), 1200, 750);  
    stage.setOnCloseRequest(Event::consume);  
    stage.setScene(scene);  
    stage.setTitle("管理员");  
    return stage;  
}
```

### 管理员界面涉及主要数据库事务

```
@Override  
public List<Allentity> getGoodsgid(Connection connection) {  
    String sql = "select * from goods ,origin where goods.gsupplier = origin.gsupplier and  
gid like concat('%!',?,'%')";  
    return getBeanList(connection,sql,AdminEvent.fristString);  
}
```

```
@Override  
public List<Allentity> getGoodsgsupplier(Connection connection) {  
    String sql = "select * from goods ,origin where goods.gsupplier = origin.gsupplier and
```



```
goods.gsupplier like concat('%',?, '%')";

        return getBeanList(connection,sql,AdminEvent.fristString);
    }

    @Override
    public void deleteEmployee(Connection connection) {
        String sql = "delete from employee where eid = ?";
        update(connection,sql,AdminEvent.fristString);
    }

    @Override
    public void updateGoodsgpurchaseprice(Connection connection) {
        String sql = "update goods set gpurchaseprice = ? where gid = ?";
        update(connection,sql,AdminEvent.fristString,AdminEvent.secondString);
    }

    @Override
    public List<AllEntity> getEmployeeTable(Connection connection) {
        String sql = "select
eid,ename,erole,eaddress,edaddress,ephone,esex,round(timestampdiff(year,ebirthday,now()))eage,
ebirthday ,esalary from employee";

        return getBeanList(connection, sql);
    }
}
```

### 收银员模块

在实现收银员模块是，出现的最主要的问题是在收银界面，如果切换不同的界面会出现小票编号改变，收银界面已经收银的商品被删除，解决方案是使用一个布尔值变量进行标识，如果小票编号存在则标识为 false 那么在切换时就不能重置小票编号，收银界面商品也不会被删除。

## 收银员界面启动

```

public Stage cashierStage() throws Exception {

    Stage stage = new Stage();

    stage.close = stage;

    Date date = new Date();

    SimpleDateFormat simpleDateFormat = new SimpleDateFormat("yyyy-MM-dd
HH:mm:ss");

    logintime = simpleDateFormat.format(date);

    FXMLLoader fxmllLoader = new
FXMLLoader(Loginin.class.getResource("Cashier.fxml"));

    Scene scene = new Scene(fxmllLoader.load(), 1200, 750);

    stage.setOnCloseRequest(Event::consume);

    stage.setScene(scene);

    stage.setTitle("收银员 编号: " + LogininEvent.eid);

    return stage;

}

```

## 收银员模块涉及主要数据库事务

```

@Override

public void deleteTicketidgid(Connection connection) {

    String sql = "delete from ticket where tid = ? and gid = ?";

    update(connection, sql, Cashierevent.fristString, Cashierevent.secondString);

}

@Override

public List<Allentity> getGoodsgid(Connection connection) {

    String sql = "select * from goods ,origin where goods.gsupplier = origin.gsupplier and
gid = ?";

    return getBeanList(connection, sql, Cashierevent.fristString);

}

```

@Override

```
public Object getPsalescounttid(Connection connection) {  
    String sql = "select count(distinct tid) from psales where eid = ? and tid like  
concat('%',?, '%')";  
    return getValue(connection,sql,Cashierevent.fristString,Cashierevent.secondString);  
}
```

@Override

```
public Object gtePssql(Connection connection) {  
    String sql = "select count(distinct tid) from psales where eid = ? and pdate >= ? and  
pdate <= ?";  
    return  
getValue(connection,sql,Cashierevent.fristString,Cashierevent.secondString,Cashierevent.thirdStri  
ng);  
}
```

@Override

```
public Object getPsalescounttid(Connection connection) {  
    String sql = "select count(distinct tid) from psales where eid = ? and tid like  
concat('%',?, '%')";  
    return getValue(connection,sql,Cashierevent.fristString,Cashierevent.secondString);  
}
```

## 7. 数据库应用系统运行及感想

### 7.1 系统运行说明<sup>1</sup>

源码地址: [https://github.com/1binbin/cash\\_register\\_system](https://github.com/1binbin/cash_register_system)

启动系统步骤<sup>2</sup>

- ① 导入项目源码, 可选择以下两种方式之一
  - 1) 复制整个项目到编译器
  - 2) 使用 GitHub 克隆项目 [https://github.com/1binbin/cash\\_register\\_system.git](https://github.com/1binbin/cash_register_system.git)
- ② 运行 sql/script.sql 中的 MySQL 脚本文件, 创建数据库以及表格
- ③ 将 lib 文件夹下的 jar 包导入项目
- ④ 将 src/main/resources/jdbc.properties 配置文件中的数据库连接信息更正, 可用 src/main/java/ConnectionTest.java 运行测试数据库连接是否正常
- ⑤ 运行 src/main/java/Main.java 文件, 启动系统

管理员初始账号与密码

账号: 1234567890      密码: admin

注 1: 为了保证系统正常运行, 需要从外部导入全国省份, 城市, 县区的数据以及积分规则表, 由于这四个表不属于系统设计得到的表, 故不在报告中呈现。但已在 SQL 脚本文件编写这四个表的建表和录入数据语句。系统使用者无需录入数据。

注 2: 由于使用 jdk11+javaFx 编写系统, 暂时不可以导出 exe 可执行文件, 需要导入源码在编译器运行。

### 7.2 系统使用说明与展示

#### 登录与重置密码模块

① 登录界面如图 49 登录界面所示, 当系统启动时, 管理员或收银员需要输入正确的个人信息方可进入系统, 否则将提示用户名或密码错误, 用户需要重置信息重新填写。



图 49 登录界面

②如果用户忘记密码时，可以在登录界面点击重置密码进入重置密码界面，系统检测用户输入的账号，如果是管理员账号，姓名与联系电话将不可输入；如果是收银员账号，姓名与联系电话必须输入才可重置密码，如图 50 重置密码界面所示。

找回密码

重置密码

账号1234567890

姓名该账号为管理员，不需要输入

联系电话该账号为管理员，不需要输入

新密码\*\*\*\*\*

确认密码\*\*\*\*\*

验证码lgmujLGNU

返回确定

找回密码

重置密码

账号8888888888

姓名hxb

联系电话12345678901

新密码\*\*\*\*\*

确认密码\*\*\*\*\*

验证码lgnuLGNU

返回确定

图 50 重置密码界面

管理员模块

1) 类别管理

为了方便管理员对超市的员工类别，商品类别以及规格进行统一管理，管理员可以在此界面先选择要操作的表格，然后选择操作类型，包括添加、删除以及修改，如果选择的是修改或删除，还需在操作对象中选择要修改或删除的对象。管理员还可以点击导出数据，将三个表以 Excel 的文件格式保存在本地电脑。

除此之外，超市设定了会员积分可队员现金的规则，管理员可以修改决定积分的兑换规则。如图 51 类别管理界面所示

管理员

商品管理 员工管理 销售记录 类别管理 财务统计 重新登录 注销

员工类别

商品类别

规格

目前每消费 1 元兑换 1 积分，每 1 积分兑换 1 元

每消费 请输入 元兑换 1 积分，每 请输入 积分兑换 1 元 确认

表格 员工类别 操作对象 请选择

操作类型 请选择

添加 删除 修改

重置 数据导出 确认

图 51 类别管理界面

2) 商品管理

管理员在商品管理界面可以对超市的商品进行管理，包括对商品的添加，删除，修改，查询以及统计等业务。在添加商品界面，管理员依次输入商品信息，如图 52 添加商品界面所示。除此之外，系统提供导入 Excel 表格批量添加商品。

管理员

商品管理

员工管理

销售记录

类别管理

财务统计

重新登录

注销

查询

添加

删除

修改

统计

商品号

978030603135

商品名

计量经济学

进 价

49

会员折扣

0

15

30

45

60

75

90

100

95%

95

类 别

书籍

进 货 量

1000

售 价

59

供 货 商

科学出版社

规 格

本

产 地

北京市

市 镇 区

朝阳区

确 认

刷 新

批 量 添 加

商品号	商品名	类别	产地	供货商	进货日期	现有库存量	规格	进价 (元)	零售价 (元)	会员价 (元)	会员折扣
表中无内容											

图 52 添加商品界面

管理员输入信息后，点击确认即可以正常将商品添加到系统中，如果出现价格格式不正确，售价低于进价等错误将弹窗提示管理员相关错误。如图 53 添加成功界面所示。

管理员

商品管理

员工管理

销售记录

类别管理

财务统计

重新登录

注销

查询

添加

删除

修改

统计

商品号

商品号

商品名

商品名

进 价

进价

会员折扣

0

15

30

45

60

75

90

100

会员折扣

95

类 别

书籍

进 货 量

进货量

售 价

普通售价

供 货 商

供货商

规 格

本

产 地

北京市

市 镇 区

朝阳区

确 认

刷 新

批 量 添 加

商品号	商品名	类别	产地	供货商	进货日期	现有库存量	规格	进价 (元)	零售价 (元)	会员价 (元)	会员折扣
978030603135	计量经济学	书籍	北京市朝阳区	科学出版社	2022-05-24	1000.0	本	49.0	59.0	56.05	0.95

提示!

添加成功!

确定

图 53 添加成功界面

除此之外，管理员可以双击表格中一行，商品编号将显示在删除框内，管理员点击删除即可将商品删除，管理员也可以选择多行点击批量删除进行一次性删除多条数据。如图 54 删除商品界面所示。

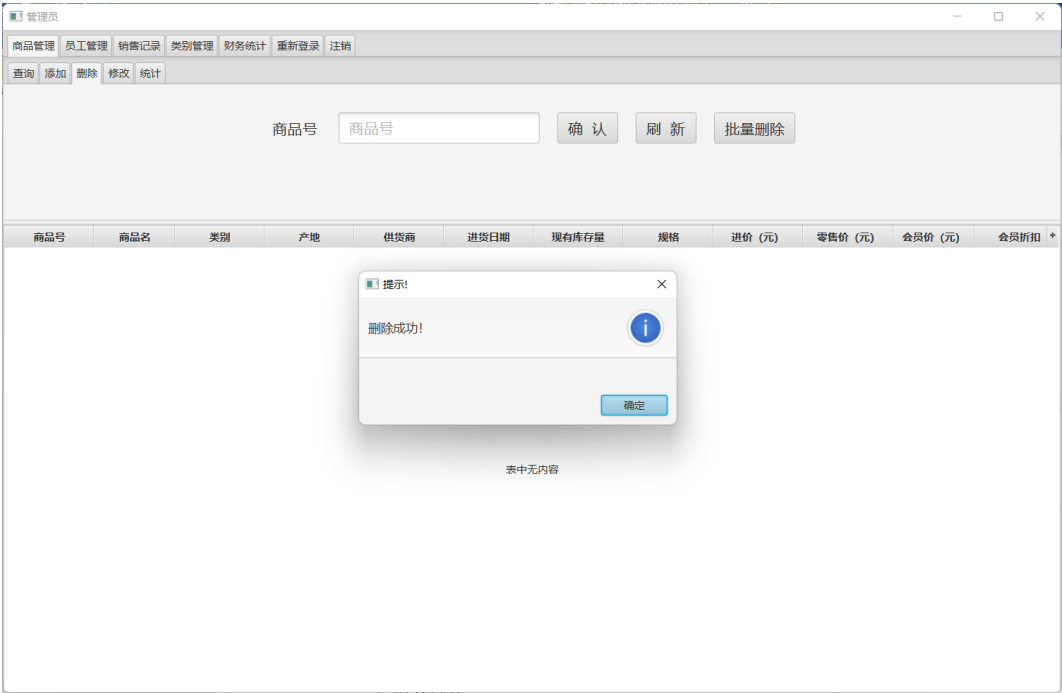


图 54 删除商品界面

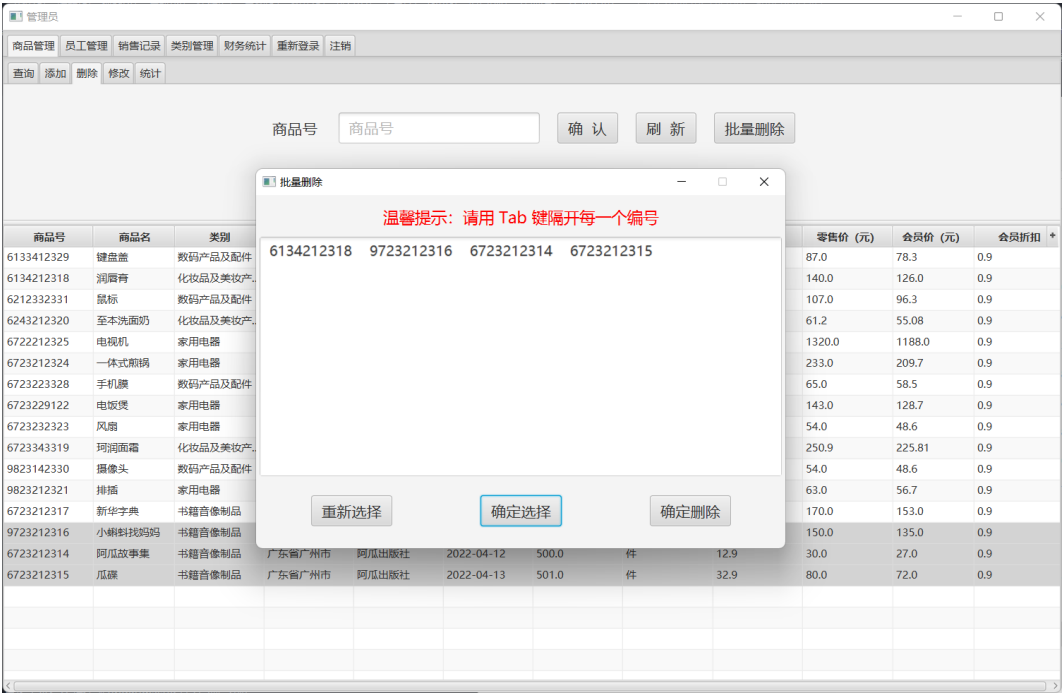


图 55 批量删除界面

为了方便管理员对商品的查询，系统提供了根据商品号，商品名，商品规格，进货日期等六种不同范围的查询，管理员选择查询范围后输出查询内容，点击确定即可以显示查询结果，如图 56 商品查询结果所示。

[illegible]

图 56 商品查询结果

系统针对商品管理还提供了本年度内不同月份商品数目的数量和进价的统计，实现第一个查询设计，如图 57 商品统计界面所示。

[illegible]

图 57 商品统计界面

除了以上的基本功能之外,系统还提供管理员将商品数据以 Excel 文件的格式导出保存在本地电脑,方便管理员对商品的管理。

### 3) 员工管理

员工管理界面主要提供了对员工信息的添加，删除，修改，查询以及一键导出数



据，批量添加与批量删除等

管理员选择操作类型之后录入正确的员工信息，其中员工编号由系统根据一定的规则生成，无需管理员输入，点击确定即可以将员工信息添加到系统中，如图 58 添加员工界面所示。添加成功后其中密码与员工编号一致，联系电话默认为空，年龄默认基于 1970-01-01 的日期计算，后续员工可在系统的个人中心中修改。

系统提供了各种不同范围对员工信息进行查询，实现了第二个查询设计。

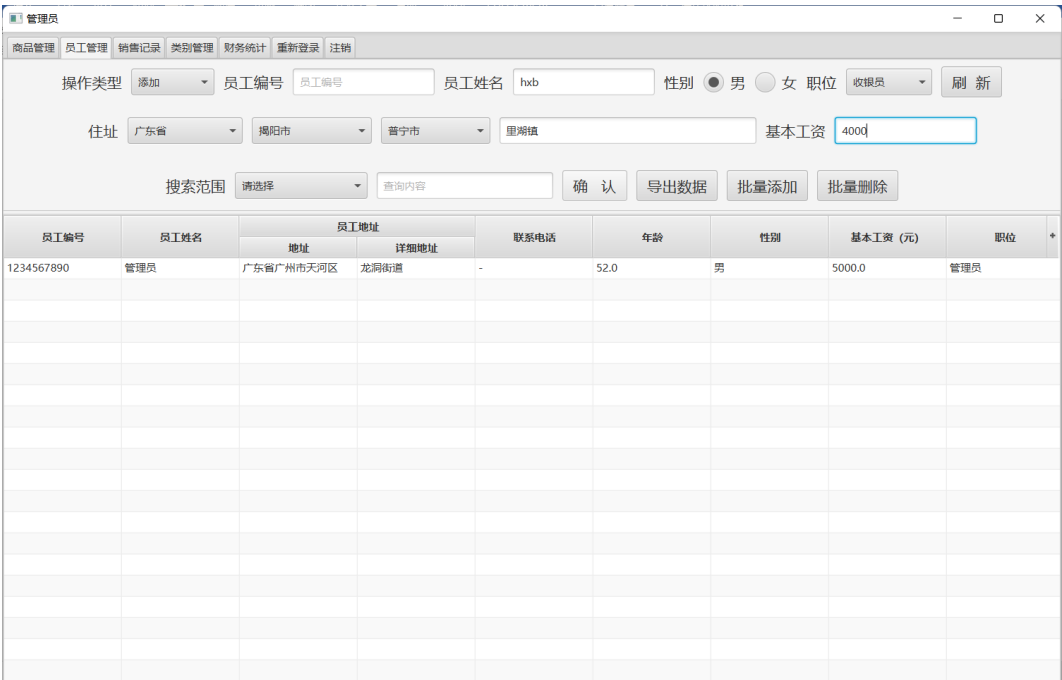


图 58 添加员工界面

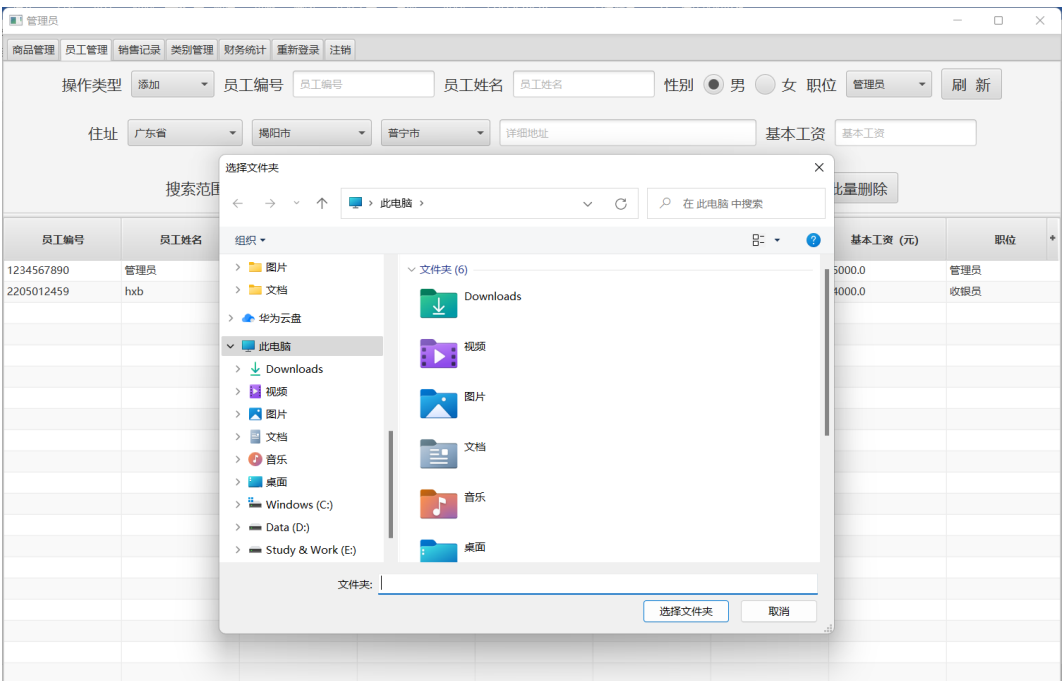


图 59 导出员工数据界面

#### 4) 销售记录

管理员在销售记录中可以管理整个超市的所有销售记录，系统提供了小票编号，商品编号，购买时间以及经手人的查询，管理员还可以将查询结果导出数据到本地电脑中方便管理，除此之外，系统提供简单的销售记录统计，实现了第五个查询设计，如图 60 管理员销售记录所示。

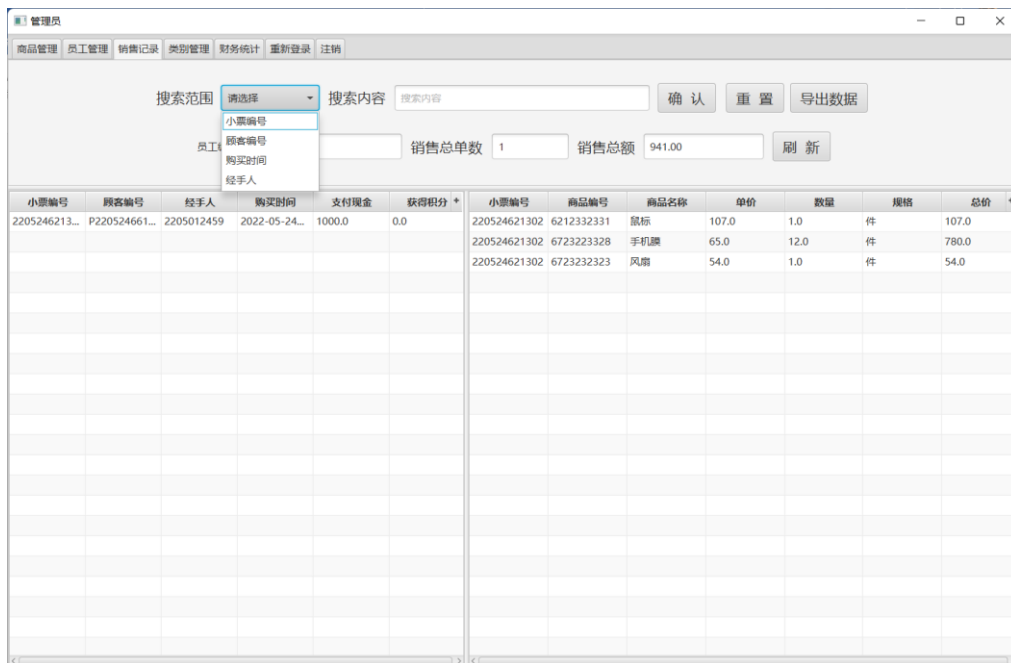


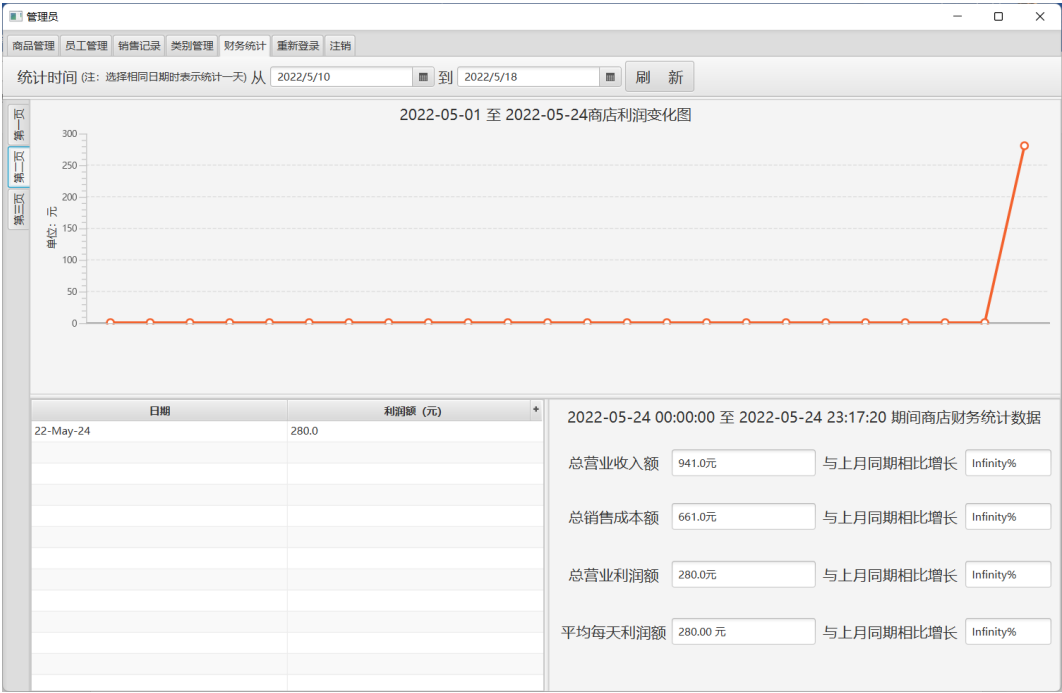
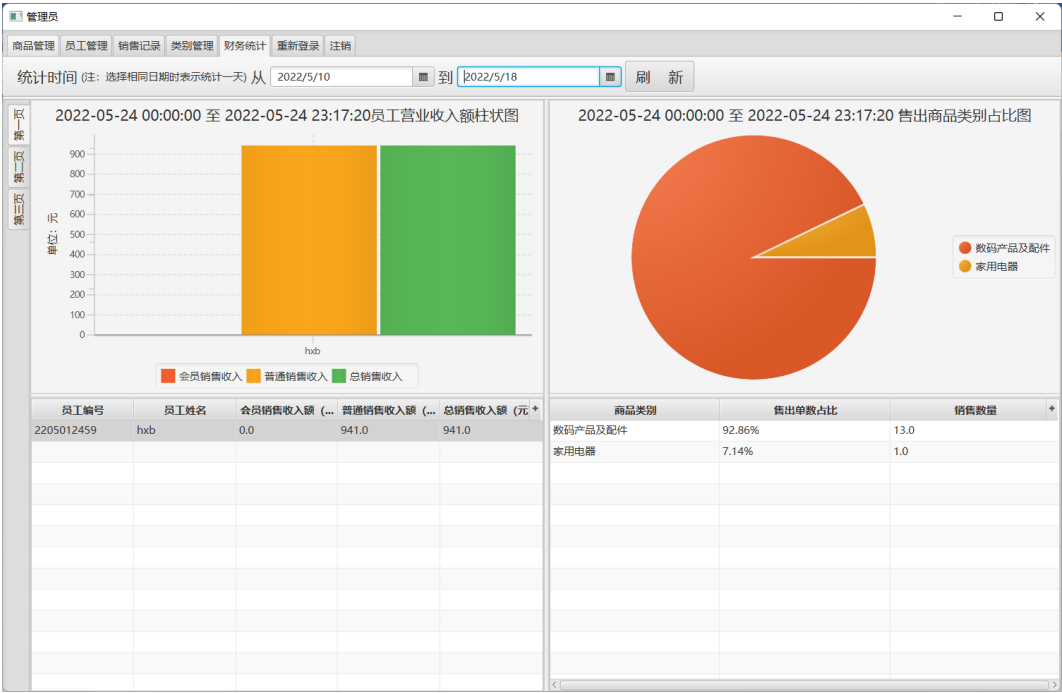
图 60 管理员销售记录

管理员双击左侧的销售记录即可以将该订单的详细信息显示在右边表格中，包括本订单包含的商品编号，商品名称，单价，购买数量，商品规格以及总价。

5) 财务统计

在财务统计界面，管理员可以对超市一定时间范围内的财务进行简单的统计，显示一定时间范围的每位员工营业收入额柱状图（实现第五个查询设计），售出商品类别占比图（实现第四个查询设计），超市利润变化折线图，年度内每月份超市利润收入变化折线图，如图 61 销售统计界面所示。

除此之外，提供了查询一定时间范围内的财务统计。



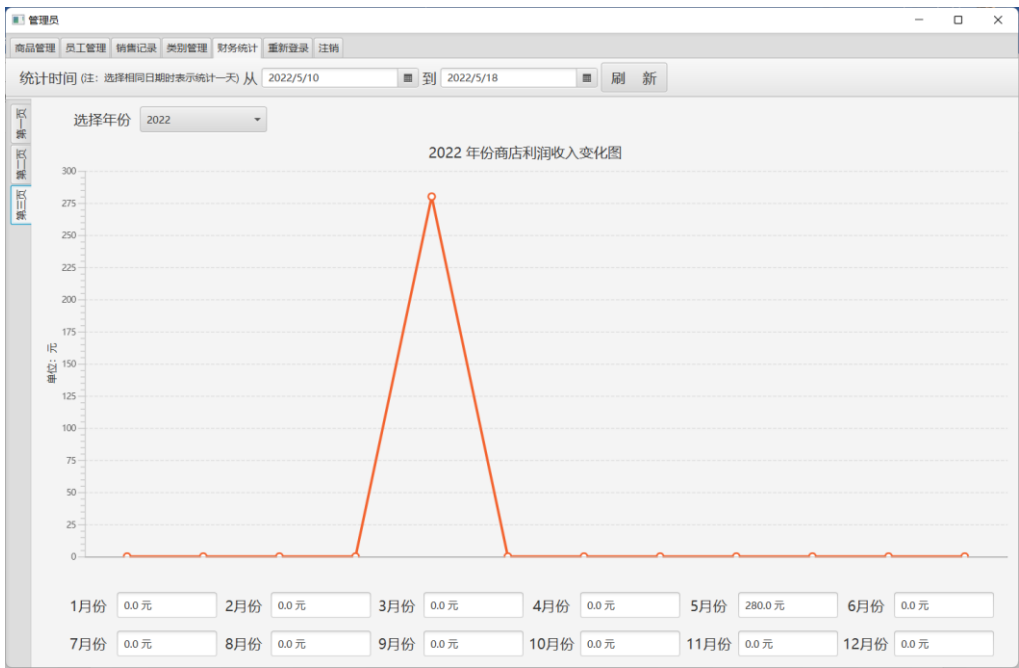


图 61 销售统计界面

收银员模块

1) 收银中心

收银中心为收银员的核心界面，收银员在这可以输入商品编号和购买数量，点击确定即可将商品数量合计，总价合计显示在界面上，当所有的商品添加完毕之后，输入顾客结账的实付金额，点击确认结账，即可以结束收银并生成小票，记录销售记录。如果顾客是会员顾客，可以选择会员并输入正确的会员编号，总价合计将更新会员价格。如图 62 收银界面所示。

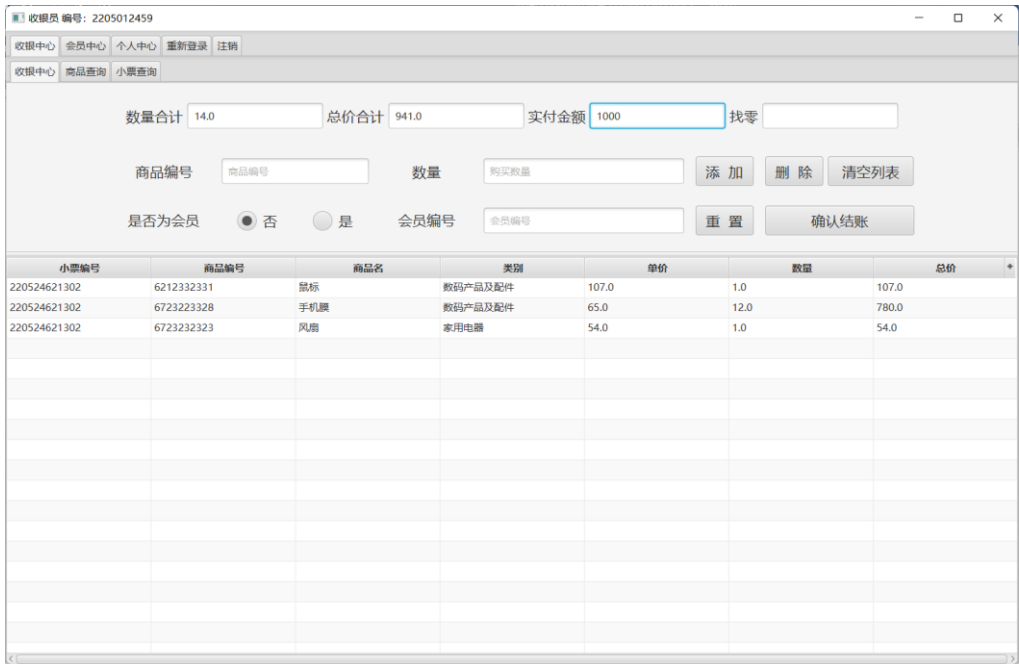


图 62 收银界面展示了收银员的核心操作界面。界面顶部显示收银员编号为2205012459。下方有“数量合计”、“总价合计”、“实付金额”和“找零”输入框。中间部分有“商品编号”、“数量”、“购买数量”输入框，以及“添加”、“删除”、“清空列表”按钮。下方有“是否为会员”选择框（默认否）和“会员编号”输入框，以及“重置”、“确认结账”按钮。底部是一个商品列表表格，显示如下：

小票编号	商品编号	商品名	类别	单价	数量	总价
220524621302	6212332331	鼠标	数码产品及配件	107.0	1.0	107.0
220524621302	6723223328	手机膜	数码产品及配件	65.0	12.0	780.0
220524621302	6723232323	风扇	家用电器	54.0	1.0	54.0

图 62 收银界面



图 63 生成小票界面

在小票查询区域，收银员可以查询到超市内的所有订单信息，双击左侧的每个订单可以在右侧中显示订单的详细信息，也可以点击查看小票显示小票的详细信息，除此之外，系统还提供了小票编号，顾客编号等四种不同的查询范围，收银员可以快速找到顾客的订单信息，实现了第六个查询设计，如图 64 小票查询界面所示。

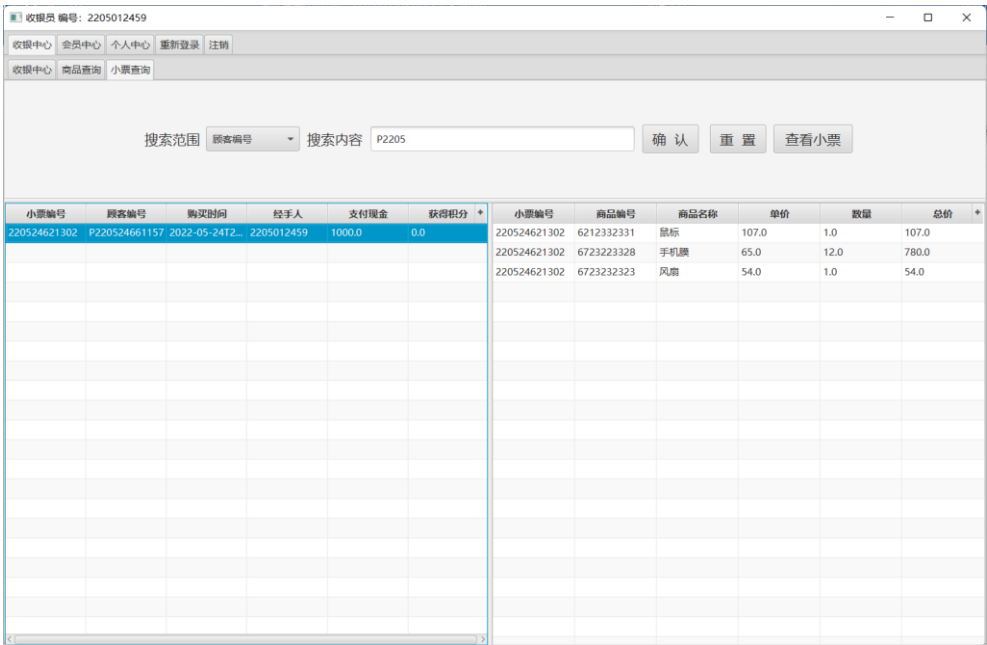


图 64 小票查询界面

## 2) 会员中心

在会员中心界面，系统提供收银员对会员信息的管理，包括注册，删除，修改，查询，会员续费，积分兑换等六种功能。如图 65 会员中心界面所示。

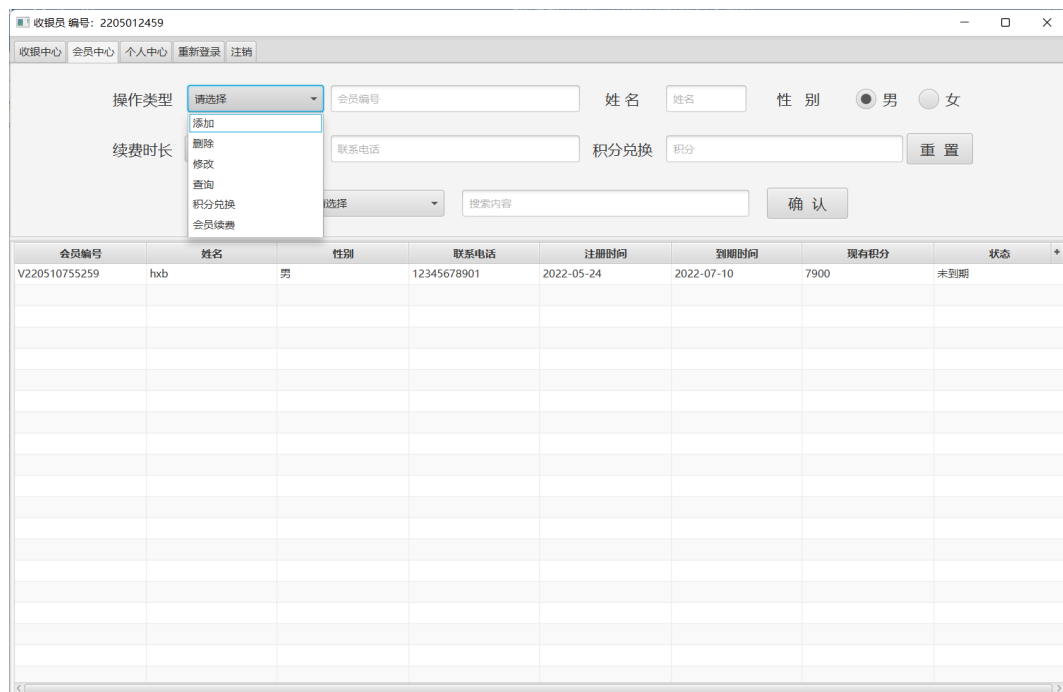


图 65 会员中心界面

### 3) 个人中心

在收银员个人中心，主要包括个人销售记录的管理和个人信息的管理

其中，个人销售记录管理中提供小票编号，顾客编号和购买时间三种不同范围的查询，收银员选择查询范围后，输入查询信息，即可以在左侧表格显示销售记录，收银员双击销售记录即可以在右侧表格显示详细信息，如图 66 个人销售记录界面所示。

在个人销售记录中还提供了简单的财务统计,包括个人销售单数和销售总额,如果进行了搜索,那么只统计搜索结果的数据,如果双击左侧表格的其中一条记录,右侧显示这次销售的详细情况,以及在统计界面显示这次销售的总销售额。

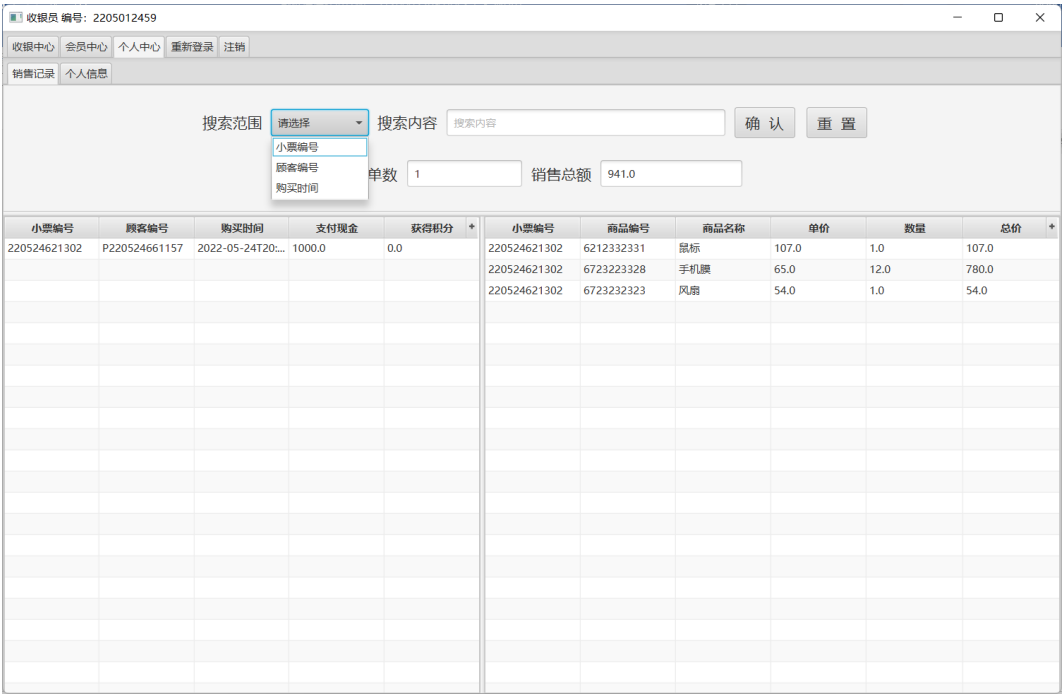


图 66 个人销售记录界面

除此之外，系统还提供简单的个人销售记录的统计，包括销售总单数和销售总额。

在个人中心界面，收银员可以对个人信息的每一项进行修改，以及显示收银员在不同年份不同月份的销售记录，包括总销售单数以及与上月增长，总销售额以及与上月增长，收银员可以选择不同年份和月份并实时更新统计数据，如图 67 个人中心界面所示。

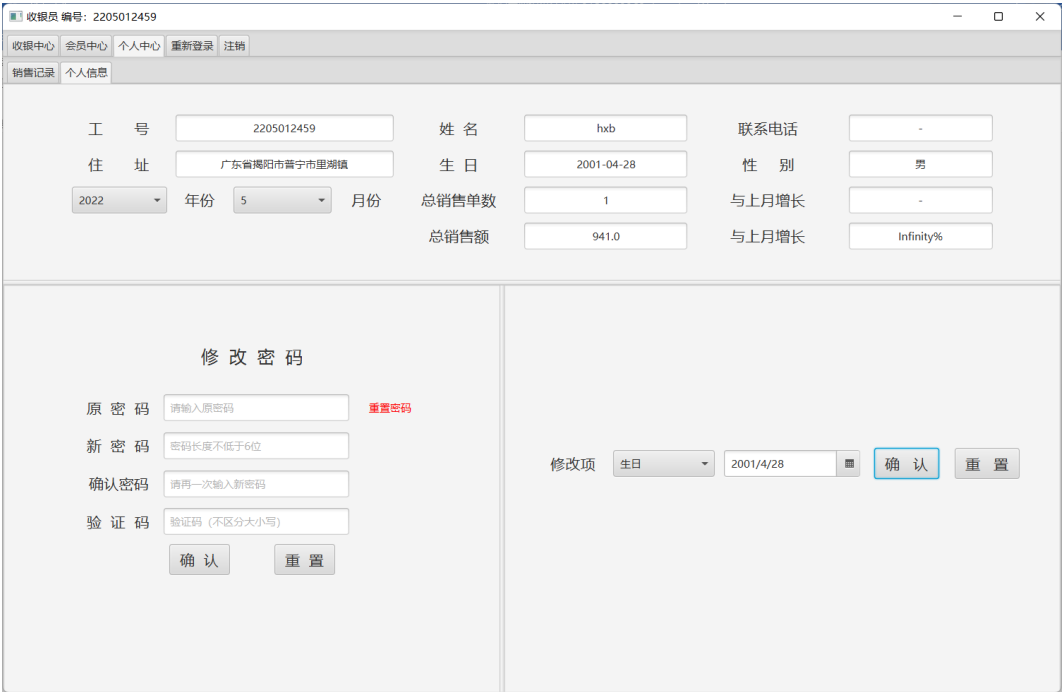


图 67 个人中心界面

## 7.3 系统评价与体会

### 7.3.1 系统评价

实现该系统使用的语言：Java + MySQL + FXML

实现该系统使用的技术：JavaFx + JDBC

该系统实现功能相对完善，设计上的功能全部已经实现，并且经过调试基本正常，实现该系统使用的主要模式为 Dao 模式。能够让执行数据库事务的 SQL 语句从事件处理中分离出来，方便后续的改进。并且都使用了接口的形式实现数据库事务，方便后续对 SQL 语句的修改或扩展业务。

该系统主要的特色在于实现了超市中管理员对超市商品和员工的管理以及收银员主要的收银业务，贯穿整个超市的运行过程，提高超市运行效率，方便超市管理，并且提供了批量插入，批量删除以及导出数据等批量操作，方便管理员的使用。除此之外，还提供下载批量插入的 Excel 模板，保证管理员正常导入数据。

实现该系统在技术上借助了 SceneBuilder 界面布局工具进行对界面设计，以及使用 Git 进行代码管理，提高开发效率。

系统仍需改善的地方主要有优化上，提供管理员个人中心，便于修改个人信息，收银结账流程需要进一步简化。功能上提供连锁店管理，商品信息标签打印等。代码上，需要对代码进行优化，精简，提取重复代码，减少代码重复度，并且将借助 Maven 依赖管理工具进行管理依赖。

### 7.3.2 体会

在本次完成课程设计的整个过程中，体会到了数据库设计应该如何从需求到实现系统，收获最大的还是能够对本课程所学知识进行一个总体的回顾，包括如何设计 ER 图，如何编写 SQL 语句实现业务等，并且能够在实践中提升自己开发能力。其中对于概念模型的定义以及概念模型的设计，关系模型，ER 图设计，实体间关系以及关系模型转化等等的一连串知识有了更深刻的理解。

但在完成本次课程设计的过程中也暴露出一些问题，最明显的是对于数据流图的设计不熟悉，对于索引的理解不深刻以及对于一些复杂的 SQL 语句依旧不熟悉等。后续我将对这些问题进行回顾与实践。

最后，我还将对该系统进行多思考，多创新，优化系统的功能，实现更多的业务，



满足超市的经营要求。并且在以后课程学习中同时思考如何将学习的理论知识进行迁移与利用，做到更好。

## 参考文献

- [1]高航. (2012). 基于会员的超市管理系统分析与设计[J]. 电子设计工程,2012,20(13):47-4953.
- [2]李迪禹. (2020). 超市管理信息系统设计与实现[J]. 北方经贸,2020,0(1):63-66.
- [3]梅瑞泽王静. (2021). 超市信息管理系统的设计与实现[J]. 电子测试,2021(18):78-80.
- [4]伍明星郑多玲,关玉蓉. (2006). 基于 Java 的超市管理系统的开发与研究[J]. 科技情报开发与经济,2006,16(24):241-243.
- [5]张珍宝刘升. (2002). 超市管理系统的分析与设计[J]. 高等函授学报：自然科学版,2002,15(3):49-51.