**本科学生综合性实验报告**

**课程名称：数据库系统原理**

电子商务数据库设计

项目组长 学号 0061375

班 级 选课03班 小组 第12组

实验项目名称 乐购电子商城销售系统设计

指导教师

开课学期 2008 至 2009 学年 第一学期

完成时间 2008年 12 月 30 日

**目 录**

[1需求分析 1](#_Toc187341041)

[1.1编写目的 1](#_Toc187341042)

[1.2背景 1](#_Toc187341043)

[1.2.1电子商务的发展历史 1](#_Toc187341044)

[1.2.2乐购电子商城开发背景 2](#_Toc187341045)

[1.3定义 2](#_Toc187341046)

[1.4目标 3](#_Toc187341047)

[1.5需求分析 3](#_Toc187341048)

[1.5.1系统的功能描述 3](#_Toc187341049)

[1.5.2系统总体功能图 4](#_Toc187341050)

[1.5.3系统流程图 6](#_Toc187341051)

[1.5.4数据流图 7](#_Toc187341052)

[1.5.5实体与数据 7](#_Toc187341053)

[1.5.6联系与数据 7](#_Toc187341054)

[1.5.7数据字典 8](#_Toc187341055)

[2概念设计 14](#_Toc187341056)

[2.1实体图 14](#_Toc187341057)

[2.2 多个实体间的联系图 16](#_Toc187341058)

[2.3总体ER图 18](#_Toc187341059)

[3逻辑设计 19](#_Toc187341060)

[3.1关系设计 19](#_Toc187341061)

[3.2关系优化 20](#_Toc187341062)

[3.3约束的说明 22](#_Toc187341063)

[3.4基本表 23](#_Toc187341064)

[4物理设计 25](#_Toc187341065)

[4.1确定数据库的存储结构 25](#_Toc187341066)

[4.2确定数据库的存取方法 26](#_Toc187341067)

# 1需求分析

本电子商城的需求分析将从编写目的、背景、定义、目标、系统功能描述等几个方面来分析。

## 1.1编写目的

让该文档的使用成员成为ESS系统的开发人员及电子商城具体组织管理人员。

## 1.2背景

电子商务，是指通过电子手段来完成商业贸易活动过程的一种方式，其最初的形式是电话、电报、电子邮件和20多年前开始的EDI（电子数据交换），到今天已经发展为通过网络来实现商品的交易和结算，使网络真正成为电子商务的重要工具，从而形成了完整的电子商务概念。

### 1.2.1电子商务的发展历史

世界上电子商务的研究始于70年代末，电子商务的实际应用起源于1996年，时间并不长，但以其高效率、低支付、高收益和全球性的特点，很快得到企业和政府的重视，发展很快。到1996年底，全球最大的500家公司中，有80%拥有自己的网址，树立了公司的网络形象。到1997年，美国在网上开设的商店已经增加到2万家。通过网络实现的电子现金业务占通货流量的10%。1998年4月6日，美国商务部发表了名为“崛起的数字经济”的报告称，北美网上购物的人数达100万。预计到本世纪末，电子商务交易额将超过2000亿美元,每年还将以超过国民经济一倍的速度增长，到2002年电子商务的交易额将达到3000亿美元，相当于美国国内总产值的8%，信息业的从业人数将达到740万人。

1996年6月联合国国际贸易委员会提出了《电子商务示范法》。1997年4月，欧盟提出了《欧盟电子商务行动方案》，同年7月美国又提出了《全球电子商务框架》。还是这一年的12月，美国和欧盟共同发表了有关电子商务的联合宣言。电子商务发展初期国际上成功运用电子商务技术的有关案例如下：

著名的跨国公司—保洁公司是生产洗衣粉、肥皂的大公司，通过企业内联网协同分销中心、生产厂业务，使库存保持在很低的水平，降低了成本，定货量增长30%，退货率降低了60%。

1995年10月，全球第一家网络银行—美国安全第一网络银行在美国诞生，这家银行只是在国际互联网上有一个网址，10名员工，1996年存款达到400万美元，预计1999年将达到4亿美元，向世人证明网络银行将向传统的银行业务提出挑战。

  1994年7月开业的网上书店—亚马逊（Amazon）在短短的几年间，就发展成为世界最大的书店，它是由31岁Jeff Bezos在租来的一间地下室里创建的。仅1997年，即可提供250万种图书供选择，拥有客户多达150万。1998年第一季度利润增长31% 达到8700万美元，上半年达2.03亿美元。世界上最大的书店，却没有一本现成的图书。这是一个网上的虚拟书店，通过网络实现客户和出版商之间的买卖交易。选择范围大，购书方便，价格低廉，是亚马逊在竞争中制胜，迅速崛起的关键.

### 1.2.2乐购电子商城开发背景

本电子商城通过搭建B-C模式的网上交易平台来完成商家与客户的商品交易活动，商家可以利用互联网进行商品的信息发布和打开产品的供销渠道，缩短生产和消费之间的时间路径、空间路径和人际路径。从而加快信息的传递速度，减少企业成本，提高企业的生产效率，增强企业营销竞争力，给销售商带来更多的利润空间。

客户可以浏览商城开放的业务和信息，可以查询商城的商品信息，若客户要购买商品，则必须在本商城注册并登录后方可进行商品交易活动。当客户登录本商城系统时，客户可以查询或修改个人信息，可以浏览、查询并购买商品，可以管理自己的购物车，可以查询订单，也可享受商城提供的个性化服务以及优惠服务等。

本电子商城同样提供了一定的后台管理功能，商城管理员可以管理客户积分与等级，删除不合法客户；可以管理商品，包括商品信息入库、商品分类管理、商品信息删除、优惠商品信息、商品信息修改、退货单管理等；可以管理订单，包括订单统计、查询历史订单、配送单管理等。

项目名称：电子商城销售管理系统（ElectronicStoreSystem 简称ESS）

项目提出者：江财ESS项目开发人员

项目开发者：ESS项目开发小组。

项目鉴定者：待定

项目开始时间：2007-9-25

## 1.3定义

系统数据流程图的一些定义



## 1.4目标

当电子商城的管理人员使用了《电子商城销售管理系统ESS》软件后，可以通过计算机完成以下几方面的管理工作：商城注册客户管理、商品管理、订单管理。电子商城销售管理系统ESS用户分为三类：

（1）商家管理员：此类客户可以取得商城管理员的权限，可以浏览所有客户信息，查找客户，给客户分配合理的权限，删除不合法客户等。

（2）商城游客：只可以浏览商城开放的业务和信息，不可以进行网上交

也不为该类客户提供个性化服务，该类客户无需注册。

（3）商城正式客户： 必须在商城注册，登录本商城后，这类客户可以

览商城开放的业务和信息，可以进行网上交易，也可享受商城提供的个性化服务以及优惠服务等。

## 1.5需求分析

### 1.5.1系统的功能描述

作为在线购物商城，前台销售系统提供以下功能：客户信息管理,商品信息管理,购物车管理，订单信息管理。

（1）客户信息管理

①客户必须注册并登录本系统才能进行网上交易活动。一个客户只能拥有一个注册号（用户名），注册号可由客户根据自己的喜好自行定义，但必须唯一且在6-16位以内,且第一位必为字母C,其他只能由数字组成。

②同一时间内一个注册号不能在多处登录。客户所填资料必须真实，其中注册号、密码、姓名、性别、地址、邮编为必填资料。

③客户的积分将根据客户的订单金额逐次积累，即客户每购买一元的商品，则客户积分增加一分.其积分等级分为一钻,二钻,三钻,一钻客户为普通客户,积分为0-499分,不享受任何优惠;二钻客户积分为500-999分以内,所有商品九折优惠;三钻客户积分为1000分以上,所有商品八折优惠.当客户积分达到一定分数后,自动修改为相应等级。

④客户注册成功以后，其注册信息将自动被加入客户表中。登录系统后，客户可以查询或修改个人信息。

（2）商品信息管理

①客户登录本系统后，可以浏览本商城所展示的商品。

②客户登录本系统后，可以查找自己所需要的商品。

③客户登录本系统后，可以购买自己选中的商品。

（3）购物车管理

当客户选中某件商品时，可以将其放入购物车（生成一商品暂存表）我们在购物车设置一个“是否购买”字段（客户可以自己选择，用于确认），一个“商品数量”（客户自己填写）字段，一个商品编号，商品名称，商品单价，商品总额。这样客户就可以自己决定购买哪些商品，购买多少，若不想买，可以在购物车中将其删除。

（4）订单信息管理

①客户确认购买购物车中的商品后，提交购物清单，此时将自动生成一张商家配送单，配送单中商品编号、商品数量、配送单编号将自动插入配送表中，而客户姓名、地址、邮编、电话则设置为默认值，即客户可以修改其中的信息。

②当客户付款后，将自动生成一张订单明细表。明细表中包括商品价格和优惠价，同时自动生成一张订单总表，订单时间由系统自动生成,即系统当前时间;订单号由系统自动生成。

③生成订单后，一天后商家发出配送单，客户收到商品，若在一周以内提出退货商品且符合退货条件(商品存在严重质量问题)，则为客户办理退货业务，同时修改相应的订单明细表和订单总表并减去客户相应的积分,同时生成相应的退货单,退货单包含商品编号,商品名称,商品单价,商品数量,退货日期。

本电子商城的后台管理系统将提供客户管理，商品管理，订单统计管理等功能，具体描述如下：

（1）客户管理

①为客户建立一张基本表,用于添加客户个人信息，客户登录后可以维护

己的个人信息，并且在向网站发出订单时会自动填写自己的联系信息。

②为客户赋予查询或修改个人信息的权利。

（2）商品管理

①若商品接近保质期（3个月），把该商品设为特价商品。

②若商品库存量小于等于100，则提示要添加商品。

③若某种商品已不再销售时，应将该商品信息删除。

④若某种商品价格改变，则修改商品价格。

⑤当商品入库时，将商品按不同的种类分类管理，分类标准为：商品类别名，生产厂家。

（3）订单统计管理

①统计每种商品年销售总额，并显示销售总额排在前十名的商品以供客户浏览。

②统计商城所有订单的年销售总额，根据销售情况调整营销计划。

③统计每一地区的销售总额。

④统计每个客户年订单总额。

⑤统计商品上个月的销售总额，并显示销售总额排在前十名的商品供客户浏览。

### 1.5.2系统总体功能图

根据上节分析的系统功能需求，我们可以得到系统的功能模块，如图1.1所示。

图1.1 系统功能图

### 1.5.3系统流程图

本商城客户购买商品的系统流程图，如图1.2所示。



图 1.2 系统流程图

### 1.5.4数据流图

本商城的数据流图如图1.3所示。

图 1.3 数据流图

客户

登录

需要购买

产品描述

商品折后金额

客户积分状况

订单细节

订单明细表

客户付款

准备配送细节

通知客户

调整

订单明细表

退货款差额调整

订单总表

客户积分、等级

调整

优惠率

发票

### 1.5.5实体与数据

通过对电子商城各方面的分析，我们可以知道电子商城中的实体包括：客户，商品，仓库，订单，优惠表，商品暂存表，商品配送单，商品退货单。各实体包含的数据项分别如下：

（1）客户：注册号，密码，地址，注册日期，邮编，电话，性别，姓名。

（2）商品：商品编号，商品名称，商品生产日期，商品保质期，商品单价。

（3）商品类别：商品类别编号，商品类别名。

（4）生产厂家：生产厂家编号，生产厂家名。

（5）仓库：仓库编号，仓库名称。

### 1.5.6联系与数据

通过以上的实体与数据我们可以得到如下实体间的联系：

（1）订单：订单编号，注册号，订单总额，订货日期，配送日期，发票号码，订单状态，商品编号，商品单价，商品折后价，商品数量。

（2）优惠表：客户等级，优惠率，积分要求。

（3）商品暂存：购物车编号，注册号，商品编号，商品单价，商品折后价，商品数量，是否购买，商品总金额。

（4）商品配送：配送单编号,注册号，商品编号，商品数量，地址，姓名，邮编，电话，配送日期。

（5）商品退货：退货单编号，订单编号，注册号，姓名，配送日期，商品编号，商品数量，退货原因。

通过以上分析，我们作如下规定：

（1）一个客户可以购买多种商品，一种商品可以被多个客户购买；

（2）一个商品可以属于一种类别，一种类别的商品可以包含多个商品；

（3）一个商品可以由多个厂家生产，一个厂家可以生产多个商品；

（4）一个订单对应一个客户，一个客户对应多个订单；

（5）一个订单对应一个商品配送单，一个商品配送单对应一个订单；

（6）一个客户对应多个商品退货单，一个商品退货单对应一个客户；

（7）一个仓库可以存放多种商品，一种商品可以存放在多个仓库；

（8）一个商品暂存表对应一个订单，一个订单对应一个商品暂存表。

实体之间的联系有：

（1）客户与商品之间（M：N）

（2）商品与商品类别之间（1：N）

（3）商品与生产厂家之间（M：N）

（4）订单与客户之间（1：M）

（5）订单与商品配送单之间（1：1）

（6）客户与商品退货单之间（1：M）

（7）仓库与商品之间（M：N）

（8）商品暂存表与订单之间（1：1）

### 1.5.7数据字典

数据字典包括数据项、数据结构、数据流、数据处理4个部分。其中数据项是数据的最小组成单位，若干个数据项可以组成一个数据结构，数据字典通过对数据项和数据结构的定义来描述数据流、数据存储的逻辑内容。

（1）数据项，如表1.1所示。

表 1.1 数据项表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据项 | 数据类型 | 别名 | 含义说明 | 取值范围 | 取值含义 |
| cust\_id | char(6) | 注册号 | 唯一标识每一客户 | C00001-C99999 | C代表客户，后五位按顺序编号 |
| cust\_code | char(12) | 密码 |  | 6-12位，由字母和数字组成 | 输错次数不得超过三次 |
| addr | char(40) | 地址 |  | 不得超过20字 | 用文字说明 |
| zip | char(6) | 邮编 |  | 000000-999999 |  |
| tel\_no | char(11) | 电话 |  | 00000000000-99999999999 | 若输入家庭号码,则要输入地区号 |
| sex | char(2) | 性别 |  | f/m | f代表男性,m代表女性 |
| discount | numeric  (1,2) | 优惠率 |  | 1.00/0.90/0.80 | 一钻不打折,二钻打九折,三钻打八折 |
| sco\_re | char(10) | 积分要求 |  | 0-499/500-999/1000以上 | 0-499:一钻,  500-999:二钻,  1000以上:三钻 |
| prod\_id | char(6) | 商品编号 | 唯一标识每一商品 | P00001-P99999 | P代表商品，后五位按顺序编号 |
| prod\_name | char(20) | 商品名称 |  | 不得超过20字 | 用文字说明 |
| pro\_date | datetime | 商品生产日期 |  | yyyy-mm-dd | 生产日期必须在系统日期之前 |
| keep\_date | char(6) | 商品保质期 |  | yyyy-mm-dd | 若保质期小于3个月,将其推为特价商品 |
| unit\_  price | numeric  (7,2) | 商品单价 |  | 0000000.00-  9999999.99 | 商品单价必须小于商品总价格 |
| sup\_name | char(30) | 生产厂家名 |  | 不得超过30字 | 用文字说明 |
| order\_no | int | 订单编号 | 唯一标识每一订单 | 1000000001-9999999999 | 按顺序依次递增自动获得流水编号 |
| tot\_amt | numeric  (7,2) | 订单总额 |  | 0000000.00-  9999999.99 | 一张订单的总金额 |
| order\_  date | getdate() | 订货日期 |  | yyyy-mm-dd | 订货日期必须在配送日期之前 |
| invoice\_  no | char(10) | 发票号码 | 唯一标识每一发票 | 2000000001-  9999999999 | 按顺序依次递增自动获得流水编号 |
| order\_status | char(2) | 订单状态 | 标识货物是否已发出 | 1和0 | 1代表货物已发出，0代表货物还未发出 |
| ware\_no | char(5) | 仓库编号 | 唯一标识每个仓库 | w0001-w9999 | w代表仓库，后四位按顺序编号 |
| qty | int | 商品数量 |  | 00001-35536 | 客户所购商品数量不得小于库存量 |
| shop\_no | char(5) | 购物车编号 | 唯一标识每一购物车 | S0001-S9999 | S代表购物车,后四位按顺序编号 |
| buy | char(4) | 是否购买 |  | 是/否 | 是代表购买, 否代表不购买 |
| pro\_  totamt | numeric  (7,2) | 商品总金额 |  | 0000000.00-  9999999.99 | 一种商品的总金额 |
| deliv\_no | char(10) | 配送单编号 | 唯一标识每一配送单 | E000000001-  E999999999 | E代表配送单,后九位按顺序编号 |
| deliv\_  date | getdate() | 配送日期 |  | yyyy-mm-dd | 配送日期在订货日期之后（隔一天） |
| chan\_no | char(10) | 退货单编号 | 唯一标识每一退货单 | T000000001-  T999999999 | T代表退货单,后九位按顺序编号 |
| chan\_rea | char(50) | 退货原因 |  | 不得超过25个字 | 用文字说明 |

（2）数据结构

①数据结构：客户

含义说明：是客户管理子系统的主体数据结构，定义了一个客户的有关信息

组成：注册号，密码，地址，注册日期，邮编，电话，性别，姓名

②数据结构：优惠表

含义说明：是优惠子系统的主体数据结构，定义了客户是否能享受优惠服务的

信息

组成：客户等级，优惠率，积分要求

③数据结构：商品

含义说明：是商品管理子系统的主体数据结构，定义了一个商品的有关信息

组成：商品编号，商品名称，商品生产日期，商品保质期，商品单价

④数据结构：商品类别

含义说明：定义了一个商品属于哪种类别的有关信息

组成：商品类别编号，商品类别名

⑤数据结构：生产厂家

含义说明：定义了一个商品是哪个厂家生产的有关信息

组成：生产厂家编号，生产厂家名

⑥数据结构：订单表

含义说明：是订单管理子系统的主体数据结构，定义了一个订单的有关信息

组成：订单编号，注册号，订单总额，订货日期，发票号码，商品编号，商品单价，商品折后价，商品数量，配送日期

⑦数据结构：商品暂存

含义说明：是购物车管理子系统的主体数据结构，定义了一张商品暂存表的有关信息

组成：购物车编号，注册号，商品编号，商品单价，商品折后价，商品数量，是否购买，商品总金额

⑧数据结构：商品配送

含义说明：是商品配送管理子系统的主体数据结构，定义了一张商品配送表的有关信息

组成：配送单编号,注册号，商品编号，商品数量，地址，姓名，邮编，电话，配送日期

⑨数据结构：商品退货

含义说明：是商品退货管理子系统的 主体数据结构，定义了一张商品退货表的有关信息

组成：退货单编号，订单编号，注册号，姓名，配送日期，退货原因，商品编号，商品数量

⑩数据结构：仓库

含义说明：是商品仓库管理子系统的主体数据结构，定义了一张仓库的有关信息

组成：仓库编号，仓库名称

（3）数据流

①数据流： 客户的个人信息

说明： 客户在注册时所登记的个人信息

数据流来源： 注册

数据流去向： 保留在客户表中

组成： 注册号，密码，地址，注册日期，邮编，电话，性别，姓名

②数据流： 客户的优惠信息

说明： 客户购买商品时所享受的优惠价格

数据流来源： 客户在订单表中的总金额

数据流去向： 保留在优惠表中

组成： 客户等级，优惠率，积分要求

③数据流： 商品的基本信息

说明： 当乐购电子购物平台增加、修改或是删除商品的时候对商品信息的更新

数据流来源： 当商品的信息发生变动的时候，由管理员执行的对商品表的增加、修改和删除的操作

数据流去向： 保存在商品表中

组成： 商品编号，商品名称，商品生产日期，商品保质期，商品单价

④数据流： 商品的类别信息

说明： 当乐购电子购物平台增加、修改或是删除商品的时候对商品类别信息的更新

数据流来源： 当商品的信息发生变动的时候，由管理员执行的对商品类别表的增加、修改和删除的操作

数据流去向： 保存在商品类别表中

组成： 商品类别编号，商品类别名

⑤数据流： 生产厂家信息

说明： 当乐购电子购物平台增加、修改或是删除厂家的时候对生产厂家信息的更新

数据流来源： 当厂家的信息发生变动的时候，由管理员执行的对生产厂家表的增加、修改和删除的操作

数据流去向： 保存在生产厂家表中

组成： 生产厂家编号，生产厂家名

⑥数据流： 订单信息

说明： 客户所选购的商品的一些基本信息

数据流来源： 当客户把选购的商品放到购物车里，点击确认以后，自动生成订单

数据流去向： 保存在订单表中

组成： 订单编号，注册号，订单总额，订货日期，发票号码，商品编号，商品单价，商品折后价，商品数量

⑦数据流： 商品暂存信息

说明： 即购物车管理系统的一些基本信息,在购物车里客户可以任意修改商品信息

数据流来源： 客户把选购的商品暂时存放到购物车里

数据流去向： 保存在商品暂存表中

组成： 购物车编号，注册号，商品编号，商品单价，商品折后价，商品数量，是否购买，商品总金额

⑧数据流： 商品配送信息

说明： 客户确认购买商品后,商家负责把商品送到客户手中

数据流来源： 订单的一些信息和客户的一些基本信息

数据流去向： 保存在商品配送表中

组成： 配送单编号,订单编号，注册号，商品编号，商品数量，地址，姓名，邮编，电话，配送日期

⑨数据流： 商品退货信息

说明： 客户若所选购的商品如有质量或者其他问题,客户可以要求退货

数据流来源： 商品的基本信息和订单的一些基本信息

数据流去向： 保存在商品退货表中

组成： 退货单编号，订单编号，注册号，姓名，配送日期，退货原因，商品编号，商品数量

⑩数据流： 仓库信息

说明： 存放各种商品

数据流来源： 当商品增加或减少时,仓库的商品库存量作相应的改变

数据流去向： 保存在仓库表中

组成： 仓库编号，仓库名称

（4）数据处理

数据处理过程如表1.2所示。

表 1.2 数据处理表

|  |  |  |
| --- | --- | --- |
| 数据存储名 | 功能 | 处理说明 |
| insert\_cust | 插入新的客户信息 | 当客户成功注册后，自动在客户表中添加该客户的注册信息 |
| select\_cust | 查询客户个人信息 | 登录系统后，客户可以查询个人信息 |
| update\_cust | 修改客户个人信息 | 登录系统后，客户可以修改个人信息 |
| delete\_cust | 删除客户信息 | 若客户离最后一次登录的时间超过一年，则管理员删除该客户信息 |
| chan\_level | 修改客户等级 | 当客户积分达到一定要求，自动修改客户等级 |
| add\_score | 增加客户积分 | 生成订单后，则根据订单金额自动增加积分 |
| code\_detect | 检测客户的注册密码是否符合要求 | 当客户输入密码时，自动检测密码是否在6-12位之间，否则将提示其重新输入 |
| select\_sales | 查询订单 | 当客户查询时，只能查询其个人订单信息 |
| insert\_product | 添加新的商品信息 | 在存储过程中用insert语句添加新的商品信息 |
| insert\_kind | 添加新的商品类别信息 | 添加新的商品信息时，若其属于新的类别，则应先在类别表中添加该商品类别 |
| insert\_sup | 添加新的生产厂家信息 | 添加新的商品信息时，若其为新的厂家生产，则应先在生产厂家表中添加该生产厂家名 |
| delete\_sup | 删除生产厂家 | 用delete操作删除某个生产厂家，此时商品应做级联删除操作 |
| delete\_product | 删除商品信息 | 当商品不再销售时，删除商品信息 |
| add\_prod | 提示添加商品信息 | 当商品库存量小于100时，提示要添加商品 |
| chan\_price | 修改商品价格 | 当商品接近保质期（用有效日期减去系统时间后，所得的值小于3个月），则把商品设为特价商品（根据商品类别修改商品价格） |
| kind\_select | 按商品类别名查询商品信息 | 客户可以输入商品类别名查询自己想要的商品 |
| sup\_select | 按生产厂家名查询商品信息 | 客户可以输入生产厂家名查询自己想要的商品 |
| prod\_name\_select | 按商品名称查询商品信息 | 客户可以输入商品名称查询自己想要的商品 |
| add\_shopcart | 生成商品暂存表（即往购物车中放入商品） | 客户点击“购买”商品时，自动生成一张商品暂存表 |
| delete\_shopcart | 删除商品暂存表（即清空购物车） | 当客户提交购物清单后，自动把购物车中的商品信息清除 |
| calculate\_shop\_amt | 计算暂存表中的商品总额 | 根据商品暂存表中的商品数量和折后价计算客户购买的商品总额 |
| update\_buy | 决定是否购买商品 | 通过update操作客户可以确认自己是否想购买该商品 |
| update\_qty | 修改购物车中的商品数量 | 在商品暂存表中客户可以通过自己update操作决定自己购买多少数量的商品 |
| delete\_shop\_prod | 删除客户不想购买的商品 | 若商品暂存表中的“是否购买”字段为“否”，则删除该商品 |
| calculate\_price | 计算商品折后价 | 根据客户等级计算商品暂存表和订单明细表的当客户提交购物清单后，商品折后价 |
| add\_deli | 生成商品配送单 | 当客户提交购物清单后，自动生成一张商品配送单 |
| update\_deli | 修改配送单中的客户信息 | 对配送单中的客户信息，客户自己可以修改 |
| insert\_item | 生成订单明细表和订单总表 | 当客户付款后，根据商品暂存表自动生成一张订单明细表同时生成一张订单总表 |
| calculate\_tot | 计算订单总表中的tot\_amt值 | 根据订单明细表中的商品数量和单价计算订单总表中的tot\_amt值 |
| status\_tri | 修改订单状态 | 当商家发出配送单时，订单状态自动修改为‘1’，表示货物已发送 |
| insert\_return | 生成退货单 | 为客户办理退货业务时，生成一张退货单 |
| update\_tri | 修改订单 | 当退货单生成后，同时修改相应的订单明细表和订单总表，即级联更新 |
| totamount | 统计在某一地区的销售总额 | 根据客户地址统计在某一地区的销售总额 |
| cal\_cust | 统计每个客户年订单总额 | 根据订单总表统计每个客户年订单总额 |
| sum\_tot | 统计商城所有订单总额 | 统计所有订单总表的订单金额 |
| 触发器名 | 功能 | 处理说明 |
| cust\_tri | 仅允许dbo用户删除客户信息 | 当删除客户信息时,如果为客户,他没有删除的权限,若为管理员,则可以删除 |
| detect\_qty | 检测客户输入的商品数量是否超过库存量 | 当客户输入商品数量时，检测其是否超过库存量，若是，则提示“您所购买的数量超过库存量,请重新输入!”，否则在商品表中库存量应做相应的减少 |
| prod\_date\_detect | 检测商品生产日期 | 添加新的商品信息时，检测其生产日期是否在系统日期之前，否则提示“生产日期不可能大于系统日期，请重新输入！” |
| deliv\_date\_detect | 检测配送日期 | 生成订单时，检测其订货日期是否在配送日期之前，否则提示“订货日期不可能大于配送日期，请重新输入！” |
| update\_sale\_item | 更新订单总表 | 当修改订单明细表的数量或单价时，订单总表的tot\_amt值应做相应地修改 |
| 视图名 | 功能 | 处理说明 |
| cust\_hero\_view | 生成客户积分排行榜（英雄排行榜） | 把客户表中积分排在前二十位的客户放在视图表中，以供客户浏览 |
| calculate\_year\_sale\_view | 统计商品年销售总额 | 年终时统计商品本年度销售总额，并显示前十名的商品 |
| prod\_well\_view | 浏览热门商品 | 统计上个月商品销售总额，并显示前十名的商品，供客户浏览 |

# 2概念设计

通过对用户需求进行综合、归纳与抽象，形成一个独立于具体DBMS的概念模型，并采用自底向上的方法．用E-R图表示各实体之间的联系。

## 2.1实体图

通过需求分析我们得到客户，商品，优惠表和仓库等最基本的实体，画出它们的实体图如下：

（1）客户实体图，如图2.1所示。



图 2.1 客户实体图

（2）商品实体图，如图2.2所示。



图 2.2 商品实体图

（3）优惠表实体图，如图2.3所示。



图 2.3 优惠表实体图

（4）仓库实体图，如图2.4所示。



图 2.4 仓库实体图

## 2.2 多个实体间的联系图

实体与实体间有多种联系，画出各种实体间的联系图如下：

（1）客户与优惠表间的联系图，如图2.5所示。



图 2.5 客户与优惠

（2）商品与商品类别间的联系图，如图2.6所示。



图 2.6 商品与商品类别表联系图

（3）商品与生产厂家之间的联系图，如图2.7所示。



图 2.7 商品与生产厂家联系图

（4）商品与仓库实体之间的联系图，如图2.8所示。



图 2.8 商品与仓库联系图

（5）客户与商品实体之间的联系图，如图2.9所示。



图 2.9 客户与商品联系图

（6）客户与商品退货之间的关系图，如图2.10所示。



图 2.10 客户与商品退货单联系图

（7）订单与商品配送之间的联系图，如图2.11所示。



图 2.11 订单与商品配送单联系图

## 2.3总体ER图

通过以上分ER图，我们得到下面的总ER图，如图2.12所示。



图 2.12 总ER图

# 3逻辑设计

本次设计的数据库在SQL Server 2000上实现，将概念结构设计中的E-R图转换成SQL Sever 2000支持的关系数据模型。

## 3.1关系设计

关系设计包括实体转换和联系转换。

（1）实体转换

将每个单独的实体转换为一张单独的表，具体转换如下：

客户：注册号，密码，地址，注册日期，邮编，电话，性别，姓名

商品：商品编号，商品名称，商品生产日期，商品保质期，商品单价。

商品类别：商品类别编号，商品类别名。

生产厂家：生产厂家编号，生产厂家名。

仓库：仓库编号，仓库名称。

（2）联系转换

一个实体型转换为关系模式，实体的属性就是关系的属性，实体的码就是关系的码。对于实体间的联系有以下几种不同的情况：

① 多对多联系转换为一张单独的关系表

商品表：商品编号，商品名称，商品生产日期，商品保质期，商品单价，库存量，生产厂家编号。

② 一对多的联系通过在多表中添加属性来实现

商品表：商品编号，商品名称，商品生产日期，商品保质期，商品单价，库存量，生产厂家编号，商品类别编号。

客户表：注册号，密码，地址，注册日期，邮编，电话，性别，姓名，客户积分，客户等级。

订单表：订单编号，注册号，订单总额，订货日期，配送日期，发票号码，订单状态，商品编号，商品单价，商品折后价，商品数量。

商品退货单：退货单编号，订单编号，注册号，姓名，配送日期，商品编号，商品数量，退货原因。

③ 一对一的联系转化为一张单独的关系表

商品暂存表：购物车编号，注册号，商品编号，商品单价，商品折后价，商品数量，是否购买，商品总金额。

商品配送单：配送单编号，注册号，商品编号，商品数量，地址，姓名，邮编，电话，配送日期，订单编号。

## 3.2关系优化

为了减少以上表数据的冗余，将表优化，结果如下：

（1）客户表：

函数依赖集为F1={注册号→（密码，地址，注册日期，邮编，电话，性别，姓名

客户等级，客户积分）}∈2NF

主属性：注册号

非主属性：密码，地址，注册日期，邮编，电话，性别，姓名，客户等级，客户积分

主键：注册号

（2）优惠表：

函数依赖集为F2={客户等级→（优惠率，积分要求）}∈3NF

主属性：客户等级

非主属性：优惠率，积分要求

主键：客户等级

（3）商品表：

函数依赖集为F3={商品编号→（商品名称，商品类别编号，生产厂家编号，商品库存量，商品生产日期，商品保质期，商品单价）} ∈3NF

主属性：商品编号

非主属性：商品名称，商品类别编号，生产厂家编号，商品库存量，商品生产日期，商品保质期，商品单价

主键：商品编号

（4）商品类别表：

函数依赖集为F4={商品类别编号→商品类别名}∈3NF

主属性：商品类别编号

非主属性：商品类别名

主键：商品类别编号

（5）生产厂家表：

函数依赖集为F5={生产厂家编号→生产厂家名}∈3NF

主属性：生产厂家编号

非主属性名：生产厂家名

主键：生产厂家编号

（6）订单总表：

函数依赖集为F6={订单编号→（注册号，订单总额，订货日期，发货日期，订单状态，发票号码）}∈3NF

主属性：订单编号

非主属性：注册号，订单总额，订货日期，发票号码，发货日期，订单状态。

主键：订单编号

（7）订单明细表：

函数依赖集为F6={（订单编号，商品编号）→（订货日期，商品单价，商品折后价，商品数量）}∈3NF

主属性：订单编号，商品编号

非主属性：订单总额，订货日期，发票号码，发货日期，商品单价，商品折后价，商品数量

主键：订单编号，商品编号

（8）商品暂存表：

函数依赖集为F8={（购物车编号，商品编号）→（注册号，商品单价，商品折后价，商品数量，是否购买，商品总金额）} ∈3NF

主属性：购物车编号，商品编号

非主属性：注册号，商品单价，商品折后价，商品数量，是否购买，商品总金额

主键：购物车编号，商品编号

（9）商品配送单：

函数依赖集为F9={配送单编号→（商品编号，商品数量，地址，姓名，邮编，电话，配送日期）}∈1NF

主属性：配送单编号

非主属性：商品编号，注册号，订单编号，商品数量，地址，姓名，邮编，电话，配送日期

主键：配送单编号，注册号，订单编号

为了方便客户查看自己订购的商品信息，故在配送单中加入商品单价和商品总金额两个非主属性。

（10）商品退货单：

函数依赖集为F10={（退货单编号，注册号，订单编号）→（姓名，配送日期，退货原因，商品编号，商品数量）}∈1NF

主属性：退货单编号，注册号，订单编号

非主属性：姓名，配送日期，退货原因，商品编号，商品数量

主键：退货单编号，注册号

（11）仓库表：

函数依赖集为F11={仓库编号→（仓库名）}∈3NF

主属性：仓库编号

非主属性：仓库名

主键：仓库编号

## 3.3约束的说明

根据参照完整性，表与表之间有主键、外键、用户自定义约束。

1. 注册号为客户表的主键，且在6-16位以内,且第一位必为字母C,其他只能由数字组成，即同时对注册号设立用户自定义约束。
2. 性别只能为‘m’或’f’,对性别设立用户自定义约束。
3. 客户密码必须在6-12位之间。
4. 客户表中的客户等级应参照优惠表中的客户等级。
5. 优惠表中客户等级为主键。
6. 仓库表中仓库编号为主键。
7. 商品表中商品编号为主键。
8. 商品类别表中商品类别编号为主键。
9. 生产厂家表中生产厂家编号为主键。
10. 商品表中的商品类别编号应参照商品类别表中的类别编号。
11. 商品表中的生产厂家编号应参照生产厂家表中的生产厂家编号。
12. 商品暂存表中购物车编号和商品编号为主键。
13. 商品暂存表中的客户编号应参照客户表中的客户编号。
14. 商品暂存表中的“是否购买”字段默认为“否”，当客户确认购买以后，再将其修改为“是”。
15. 商品暂存表中的商品数量默认为“0”。
16. 订单总表中订单编号为主键。
17. 订单明细表中订单编号和商品编号为主键。
18. 订单明细表中的订单编号应参照订单总表中的订单编号。
19. 订单明细表中的商品编号应参照商品表中的商品编号。
20. 订单总表中的订货日期应早于配送日期。
21. 发票号码必须唯一，建立unique约束。
22. 商品折后价和订单总额均设置默认值为0。
23. 订单状态设置默认值为0。
24. 商品配送单中配送单编号为主键。
25. 商品配送单中的客户信息默认为已注册客户的信息。
26. 商品退货单中退货单编号、注册号、订单编号为主键。
27. 一个客户购买某种商品的数量不能超过该商品的库存量，当客户购买之后，该商品的库存量应自动减少。
28. 当生成订单明细表后，订单总表要自动生成，订单时间由系统自动生成，当修改订单明细表时，订单总表也要做相应的修改。
29. 在生成订单总表后,自动生成商品配送单。
30. 为客户办理退货业务后，应自动生成一张退货单，同时在订单明细表和订单总表中要做相应的修改。

## 3.4基本表

通过上面的关系转换与关系优化,我们最终得到11张基本表。

（1）客户表（customer表）如表 3.1所示。

表 3.1 客户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| cust\_id | char(6) | not null | 注册号 | primary key |
| cust\_code | char(12) | not null | 密码 |  |
| addr | char(40) | not null | 地址 |  |
| regis\_date | getdate() | null | 注册日期 |  |
| zip | char(6) | not null | 邮编 |  |
| tel\_no | char(11) | not null | 电话 |  |
| sex | char(2) | not null | 性别 |  |
| cust\_name | Char（8） | not null | 姓名 |  |
| cust\_level | char(8) | not null | 客户等级 |  |
| cust\_sco | int | not null | 客户积分 |  |

（2）优惠表（discount表）如表 3.2所示。

表 3.2 优惠表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| cust\_level | char(4) | not null | 客户等级 | primary key |
| discount | numeric(7,2) | not null | 优惠率 |  |
| sco\_re | char(20) | not null | 积分要求 |  |

（3）商品表（product表）如表 3.3所示。

表 3.3 商品表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| prod\_id | char(6) | not null | 商品编号 | primary key |
| prod\_name | char(20) | not null | 商品名称 |  |
| kind\_no | char(6) | not null | 商品类别编号 |  |
| sup\_no | char(6) | not null | 生产厂家编号 |  |
| storage | int | not null | 商品库存量 |  |
| pro\_date | datetime | not null | 商品生产日期 |  |
| keep\_date | char(6) | not null | 商品保质期 |  |
| unit\_price | numeric(7,2) | not null | 商品单价 |  |
| supply | int | not null | 供货量 |  |

（4）商品类别表（kind表）如表 3.4 所示。

表 3.4 商品类别表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| kind\_no | char(6) | not null | 商品类别编号 | primary key |
| kind\_name | char(15) | not null | 商品类别名 |  |

（5）生产厂家表（company表）如表 3.5所示。

表 3.5 生产厂家表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| sup\_no | char(6) | not null | 生产厂家编号 | primary key |
| sup\_name | char(30) | not null | 生产厂家名 |  |

（6）订单总表（sales表）如表 3.4.6所示。

表 3.6 订单总表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| order\_no | char(10) | not null | 订单编号 | primary key |
| cust\_id | char(16) | not null | 注册号 |  |
| tot\_amt | numeric(7,2) | null | 订单总额 |  |
| order\_date | getdate() | not null | 订货日期 |  |
| invoice\_no | char(15) | not null | 发票号码 |  |
| order\_status | char(2) | not null | 订单状态 |  |
| deliv\_date | datetime | not null | 配送日期 |  |

（7）订单明细表（sale\_item表）如表 3.7所示。

表 3.7 订单明细表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| order\_no | char(10) | not null | 订单编号 | primary key |
| prod\_id | char(6) | not null | 商品编号 | primary key |
| unit\_price | numeric(7,2) | not null | 商品单价 |  |
| dis\_price | numeric(7,2) | not null | 商品折后价 |  |
| qty | int | not null | 商品数量 |  |
| order\_date | datetime | not null | 订货日期 |  |

（8）商品暂存表（shopcart表）如表 3.8所示。

表 3.8 商品暂存表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| shop\_no | char(5) | not null | 购物车编号 | primary key |
| cust\_id | char(6) | not null | 注册号 |  |
| prod\_id | char(6) | not null | 商品编号 | primary key |
| unit\_price | numeric(7,2) | not null | 商品单价 |  |
| dis\_price | numeric(7,2) | not null | 商品折后价 |  |
| qty | int | not null | 商品数量 |  |
| buy | char(4) | not null | 是否购买 |  |
| pro\_totamt | numeric(7,2) | not null | 商品总金额 |  |

(9)商品配送单（delivery表）如表 3.9所示。

表 3.9 商品配送单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| deliv\_no | char(10) | not null | 配送单编号 | primary key |
| cust\_id | char(6) | not null | 注册号 |  |
| order\_no | char(10) | not null | 订单编号 |  |
| prod\_id | char(6) | not null | 商品编号 |  |
| qty | int | not null | 商品数量 |  |
| unit\_price | numeric(7,2) | not null | 商品单价 |  |
| tot\_amt | numeric(9,2) | not null | 订单总额 |  |
| zip | char(6) | not null | 邮编 |  |
| addr | char(40) | not null | 地址 |  |
| tel\_no | char(11) | not null | 电话 |  |
| deliv\_date | getdate() | not null | 配送日期 |  |
| cust\_name | char(8) | not null | 姓名 |  |

（10）商品退货表（return表）如表3.10所示。

表 3.10 商品退货单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| chan\_no | char(10) | not null | 退货单编号 | primary key |
| cust\_id | char(6) | not null | 注册号 | primary key |
| order\_no | char(10) | not null | 订单编号 | primary key |
| deliv\_date | getdate() | not null | 配送日期 |  |
| chan\_reason | char(50) | not null | 退货原因 |  |
| prod\_id | char(6) | not null | 商品编号 |  |
| qty | int | not null | 商品数量 |  |
| cust\_name | char(8) | not null | 姓名 |  |

（11）仓库表（warehouse）如表 3.11所示。

表 3.11 仓库表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 属性名 | 数据类型 | 是否为空 | 含义 | 是否为主键 |
| wh\_no | char(5) | not null | 仓库编号 | primary |
| wh\_name | char(10) | not null | 仓库名称 |  |

# 4物理设计

数据库在物理设备上的存储结构与存取方法称为数据库的物理结构，它依赖于选定的数据库管理系统。为一个给定的逻辑数据模型选取一个最适合应用要求的物理结构的过程，就是数据库的物理设计。它包括如下几个方面：

## 4.1确定数据库的存储结构

确定数据库物理结构主要指确定数据的存放位置和存储结构，包括：确定关系、索引、聚簇、日志、备份等的存储安排和存储结构，确定系统配置等。

（1）确定数据的存放位置

为了提高系统性能，我们根据应用情况将数据的易变部分与稳定部分、经常存取部分和存取频率较低部分分开存放。

即把优惠表、仓库表作为数据稳定和存取频率较低部分；而把客户表、商品表、商品类别表、生产厂家表、订单总表、订单明细表、商品暂存表、商品配送单、商品退货单作为数据易变和经常存取部分，分开存放。

（2）确定系统配置

在进行物理设计时，根据应用环境，将数据库的大小的参数值设置为：事务日志的分配空间为1.00MB，文件按10百分比自动增长，并将文件增长限制为10.00MB；数据文件的分配空间为2.00MB，文件按10百分比自动增长，并将文件增长限制为10.00MB。

## 4.2确定数据库的存取方法

数据库系统是多用户共享的系统，对同一个关系要建立多条存取路径才能满足多用户的多种应用要求。物理设计的任务之一就是要确定哪些存取方法，即建立哪些存取路径。常用的存取方法有三类。第一类是索引方法，目前主要是B+树索引方法；第二类是聚簇方法；第三类是HASH方法。

（1）索引存取方法的选择

①由于客户表的cust\_id经常在查询和连接操作的条件中出现，故在cust\_id上建立B+树索引。

②由于商品表的prod\_id、prod\_name、kind\_no、sup\_no经常在查询和连接操作的条件中出现，故在这些属性上建立B+树索引。

③由于商品类别表中kind\_no、kind\_name经常在查询和连接操作的条件中出现，故在这些属性上建立B+树索引。

④由于生产厂家表中的sup\_no、sup\_name经常在查询和连接操作的条件中出现，故在这些属性上建立B+树索引。

⑤由于订单总表中的order\_no、cust\_id经常在查询和连接操作的条件中出现，故在这些属性上建立B+树索引。

⑥由于订单总表的tot\_amt和订单明细表的qty、dis\_price经常作为最大值、最小值以及求和等聚集函数的参数，故在这些属性上建立B+树索引。

⑦由于订单明细表的order\_no、order\_date经常在查询和连接操作的条件中出现,故在这些属性上建立B+树索引。

⑧对商品配送单中的deliv\_no建立B+树索引。

⑨对商品退货单中的chan\_no、cust\_id、order\_no建立B+树索引。

（2）聚簇存取方法的选择

①由于订单总表和订单明细表的order\_no经常要进行连接操作，为了提高order\_no的查询速度，则对此关系建立一个聚簇索引。

②由于订单明细表和商品表的prod\_id经常要进行连接操作，为了提高prod\_id的查询速度，故对此关系建立一个聚簇索引。

③由于订单总表和客户表的cust\_id经常要进行连接操作，为了提高cust\_id的查询速度，故对此关系建立一个聚簇索引。

（3）HASH存取方法的选择

①由于优惠表中的cust\_level主要在等值连接条件中出现且该表大小不变，故在此属性上建立HASH存取方法。

②由于仓库表中的ware\_no主要在等值连接条件中出现且该表大小不变，故在此属性上建立HASH存取方法。