**网上书城需求分析及数据库结构**

**一、需求分析**

1、图书查询：图书查询的目的是为了帮助用户快速地找到想要购买的图书。为了适合用户的不同需求，提供了不同的查询方法，可以按照书名、作者、出版社等方式查询，用户只需输入部分信息即可。

用户还可以进行分类浏览，在页面上，应向用户显示书籍类别，用户选择了某个类别之后，应能显示该类别下的实际书籍（如果有）的信息：书籍封面图片、书籍名称、定价、当书籍数目过多时，要提供分页浏览功能。

2、图书展台：图书展台的目的是为了将网上书店中的图书有序地罗列出来。图书展台分为：新品图书、推荐图书、畅销图书、排行榜。（图书详细资料：当用户点击某图书后，页面将显视出该图书书名、作者、出版社、出版日期、开本、ISBN、图书图片、图书详细介绍、图书目录等相关信息）。

3、购物车：购物车中要列出用户所选择的书籍名称、市场价、数量，总计金额，共节省金额。用户可以：

（1）删除购物车中的书籍；

（2）修改某个书籍的数量、书籍的金额和总计金额；

（3）清空购物车；

（4）结算：到结算页面确认并下订单。

4、订单生成与管理：购物后，确认支付和配送方法后生成订单。在生成订单的同时，系统发邮件通知书店管理员。还可以查询自己订单的处理情况和历史订单。

验证合法的用户可查看在网站上成交的订单，包括订单的个数，总计金额，订单的列表。列表项包括：订单号、订单日期、收货人、订单金额、订单状态，当订单数目过多时，要提供分页浏览功能。对于列表的订单，用户可以：

（1）查看订单详细信息：生成一个订单的详细信息页面，包括：订单号、收货人、联系电话、送货地址、用户订购的书籍列表（序号、书籍编号、原价、订购数量、折扣、销售价）、金额总计、订购日期。在页面上提供直接打印的功能；

（2）修改订单：只允许对未审核的订单进行修改（包括修改收货人、联系电话、送货地址、用户订购的书籍列表）；

（3）删除订单：只允许对未审核的订单进行删除。

5、结算功能：结算前必须再次验证用户，验证合法后，进入结算页面，显示订单的日期、收货信息、购买书籍清单，用户还可以再次修改收货人信息，修改或删除已选书籍，当确认后，可以选择下订单。

6、在线帮助：简单介绍网站的购物流程，以及一些常见问题。

**二、数据库表结构**

客户（客户编号，客户名，客户密码，真实姓名，性别，客户电话，E-mail，地址）

tb\_customerinfo客户信息表

create table tb\_customerinfo (

   CustomerId    int     identity,

   CustomerName   varchar(20)   not null,

   CustomerPwd    varchar(20)   not null,

   Customertruename  varchar(20)   not null,

   CustomerSex    varchar(2)    not null

 constraint CKC\_CUSTOMERSEX\_TB\_CUSTO check (CustomerSex in('男','女')),

   CustomerTel    varchar(20)   not null,

   CustomerEmail   varchar(20)   not null,

   CustomerAddr   varchar(20)   not null,

   constraint PK\_TB\_CUSTOMERINFO primary key(CustomerId, CustomerName)

)

管理员（管理员编号，管理员名称，密码）

create table tb\_manager (

   AdminId  int    identity,

   AdminName varchar(10) not null,

   AdminPwd  varchar(20) not null,

   constraint PK\_TB\_MANAGER primary key (AdminId)

)

图书类型（图书类型编号，类别名称）

create table tb\_booktypeinfo (

   BookTypeId   int     identity,

   BookTypeName  varchar(50)  not null,

   constraint PK\_TB\_BOOKTYPEINFO primary key (BookTypeId)

)

图书（图书编号，图书类别，图书名字，作者、出版社、出版日期、开本、ISBN、图书价格，图书图片、图书详细介绍）

create table tb\_bookinfo (

   BookId     int    identity,

   BookTypeId   int    not null,

   BookName   varchar(20) not null,

BookAuthor   varchar(10) not null,

   BookPress   varchar(20) not null,

   BookPubDate  datetime   not null,

   BookSize    varbinary(10) not null,

   Bookisbn    varchar(20)  not null,

   BookPrice    money    not null,

  BookPic    money     not null

   BookOutline  varchar(200)  not null,

   constraint PK\_TB\_BOOKINFO primary key (BookId)

购物车零时订购信息（客户编号，图书编号，订购数量，总价，是否付款）

create table tb\_shopbook (

   CustomerId   int    not null,

   BookId     int    not null,

   ordermount   int    not null,

   price      money  not null,

   ispay      varchar(10)  not null default '未付款'

   constraint CKC\_ISPAY\_TB\_SHOPB check (ispay in ('未付款','已付款')),

   constraint PK\_TB\_SHOPBOOK primary key (CustomerId, BookId)

)

订单（Id，订单号，客户编号，下单日期，总订购数量，送货方式，支付方式，收货人姓名，收货地址，收货人联系方式，备注，总卖出价）

create table tb\_order (

   Id     int    identity,

   OrderId    varchar(20)  not null,

   CustomerId  int      not null,

   Orderdate   datetime   not null,

   Ordermount int      not null,

   postmethod  varchar(100)  not null,

   paymethod  varchar(100)  not null,

   recevername varchar(10)   not null,

   receveraddr  varchar(20)  not null,

   recevertel  varchar(10)   not null,

   memo   varchar(100)  null,

   totalprice  money    not null,

   constraint PK\_TB\_ORDER primary key (OrderId)

)

详细订单（Id，详细订单编号，订单号，图书编号，订购数量，发货状态，收货状态，卖出总价）

create table tb\_orderdetail (

   id      int     identity,

   OrderDetailId  varchar(20)  not null,

   OrderId    varchar(20)   not null,

   BookId    int      not null,

   ordermount  int      not null,

   poststatus  varchar(10)   not null default '未发货'

constraint CKC\_POSTSTATUS\_TB\_ORDER check (poststatus in ('未发货','已发货')),

   Recevstatus  varchar(10)   not null default '未收货'

constraint CKC\_RECEVSTATUS\_TB\_ORDER check(Recevstatus in ('未收货','已收货')),

   sigletotalprice money    not null,

   constraint PK\_TB\_ORDERDETAIL primary key (OrderDetailId)

)