

《图书馆管理系统》

数据库设计说明书

学生姓名：

学 号：

专 业：

年 级：

**目 录**

[1 数据库概念结构设计 6](#_Toc118896144)

[2 数据库逻辑结构设计 6](#_Toc118896145)

[3 数据库物理结构设计 6](#_Toc118896146)

[3.1 数据库物理结构关联图 6](#_Toc118896147)

[3.2 数据表结构详细设计 6](#_Toc118896148)

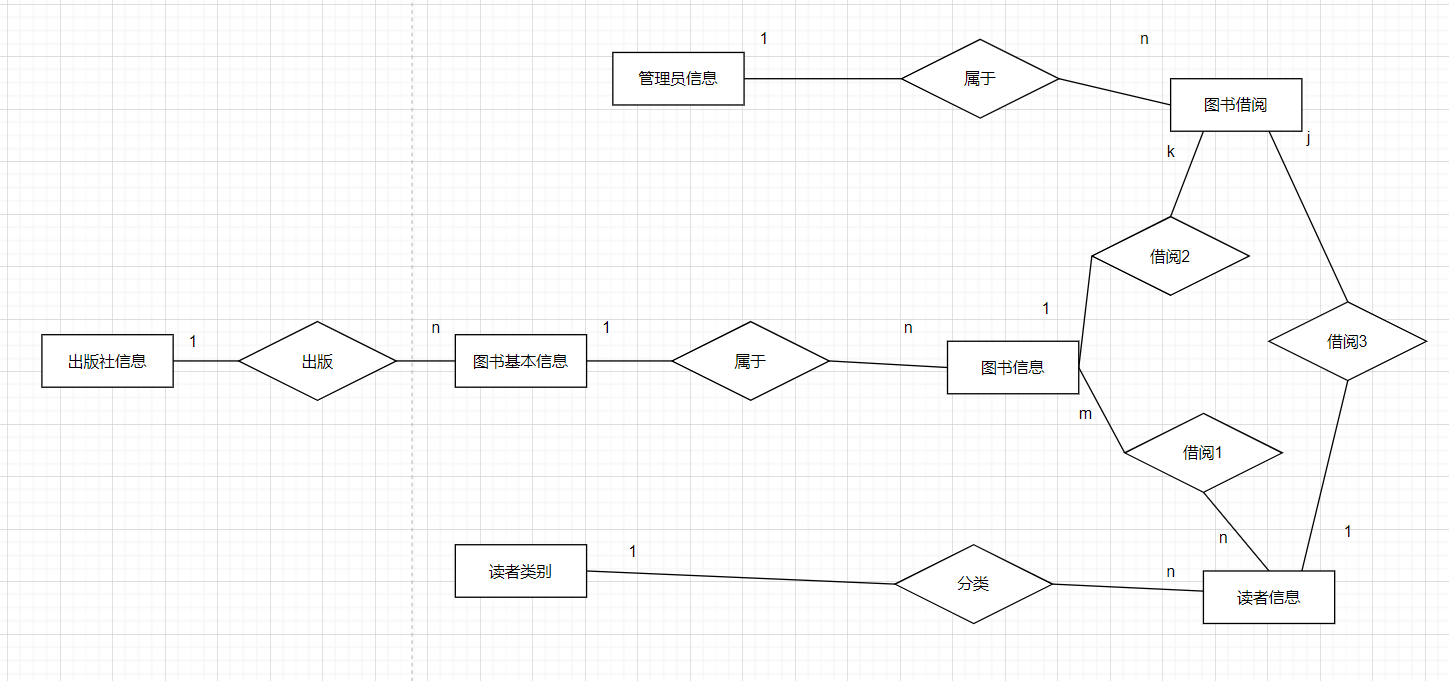
[4 数据库SQL定义语句 6](#_Toc118896149)

[4.1 MySQL定义语句 6](#_Toc118896150)

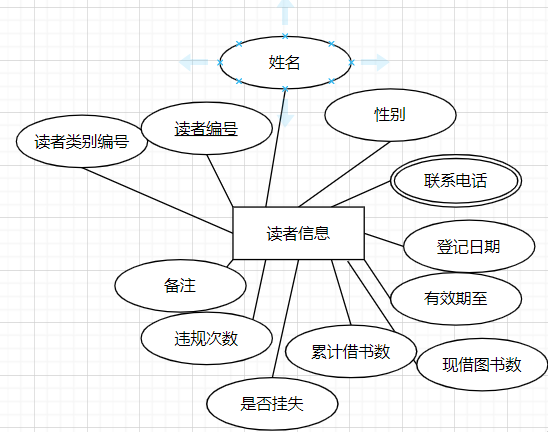
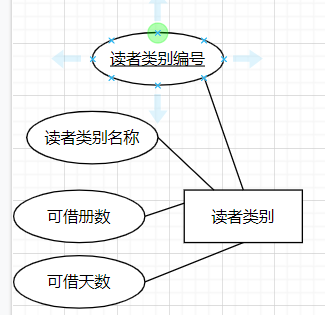
[4.2 SQL Server定义语句 6](#_Toc118896151)

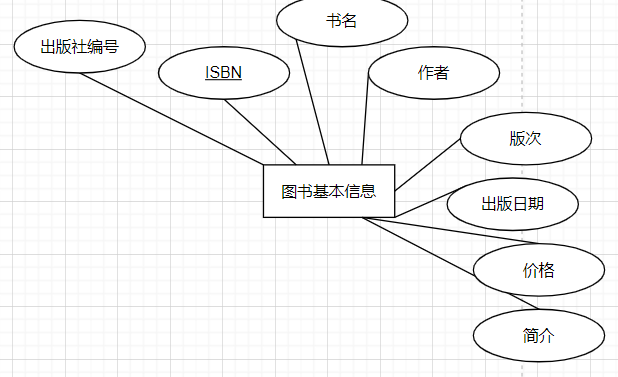
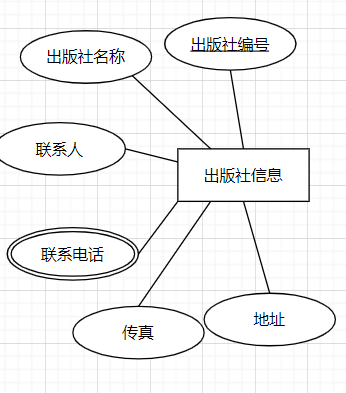
# 1 数据库概念结构设计

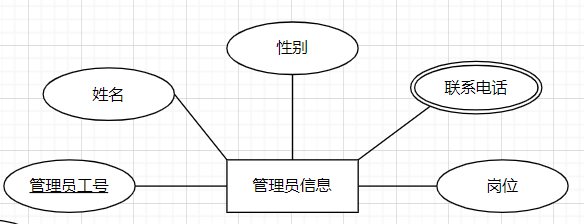
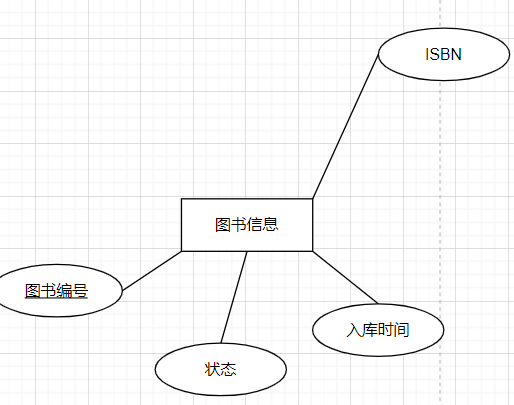
各实体间的联系如下（省略属性）

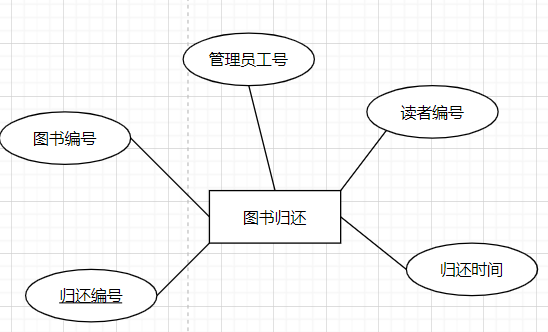
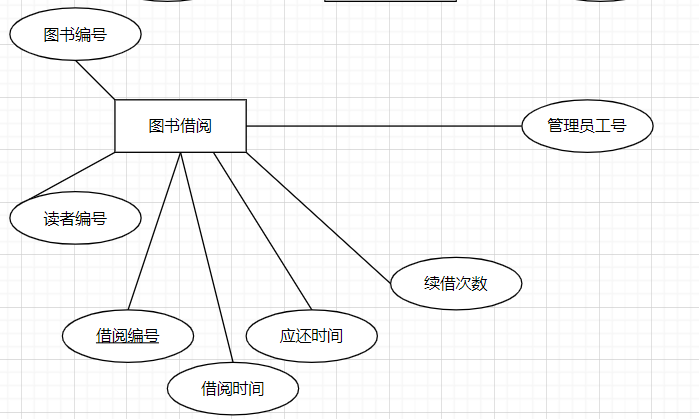


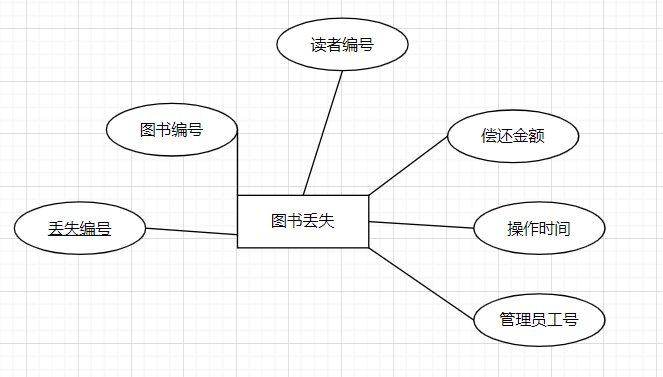
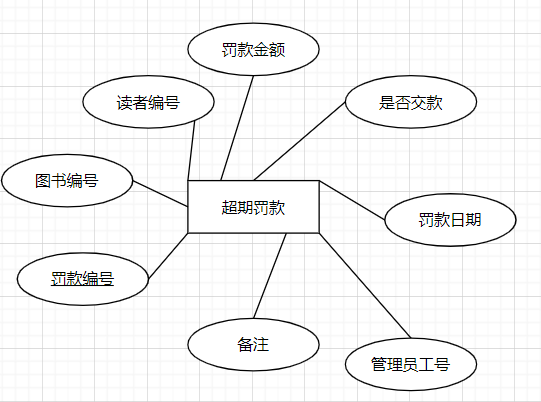
各实体和属性间的关系











# 2 数据库逻辑结构设计

关系模式

（1）读者类别

读者类别（读者类别编号，读者类别名称，可借册数，可借天数）

其中，“读者类别编号”是主键

（2）读者信息

读者信息（读者编号，读者类别编号，姓名，性别，联系电话，登记日期，有效期至，现借图书数，累计借书数，是否挂失，违规次数，备注）

其中，“读者编号”是主键，“读者类别编号”是外键

（3）图书基本信息

图书基本信息（ISBN，书名，类型，作者，出版社编号，版次，出版日期，价格，简介，现在库数量，库存总量）

其中，“ISBN”是主键，“出版社编号”是外键

（4）出版社信息

出版社信息（出版社编号，出版社名称，联系人，联系电话，传真，地址）

其中，“出版社编号”是主键

（5）图书信息

图书信息（图书编号，ISBN，入库时间，图书状态）

其中，“图书编号”是主键，“ISBN”是外键

（6）管理员信息

管理员信息（管理员工号，姓名，性别，联系电话，岗位）

其中，“管理员工号”是主键

（7）图书借阅

图书借阅（借阅编号，图书编号，读者编号，借阅时间，应还时间，续借次数，管理员工号）

其中，“借阅编号”是主键，“图书编号”、“读者编号”、“管理员工号”是外键

（8）图书归还

图书归还（归还编号，图书编号，读者编号，归还时间，管理员工号）

其中，“归还编号”是主键，“图书编号”、“读者编号”、“管理员工号”是外键

（9）超期罚款

超期罚款（罚款编号，图书编号，读者编号，罚款金额，是否交款，罚款日期，管理员工号，备注）

其中，“罚款编号”是主键， “图书编号”、“读者编号”、“管理员工号”是外键

（10）图书丢失

图书丢失（丢失编号，图书编号，读者编号，偿还金额，操作时间，管理员工号）

其中，“丢失编号”是主键，“图书编号”、“读者编号”、“管理员工号”是外键

# 3 数据库物理结构设计

## 3.1 数据库物理结构关联图



## 3.2 数据表结构详细设计

（1）读者类别表

读者类别表是用于存储本系统中所有读者的类别。本系统的读者类别可以分为教师、职工、学生。用户角色表的结构如表3.1所示。

表3.1 读者类别表(Reader\_Type)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | RTID | Int | 4 | 主键 | 否 | 读者类别编号 |
| 2 | RTName | Varchar | 30 |  | 否 | 读者类别名称 |
| 3 | BNum | Int | 4 |  | 否 | 可借册数 |
| 4 | BDay | Int | 4 |  | 否 | 可借天数 |

（2）读者信息表

读者信息表是用于存储本系统中所有读者的信息。读者信息表的结构如表3.2所示。

表3.2 读者信息表(Reader\_Info)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | RID | Int | 4 | 主键 | 否 | 读者编号 |
| 2 | RTID | Int | 4 | 外键 | 否 | 读者类别编号 |
| 3 | RName | Varchar | 30 |  | 否 | 姓名 |
| 4 | Sex | Bit | 1 |  | 否 | 性别 |
| 5  6  7  8  9  10  11  12 | Phone  ReDate  ValDate  CurBNum  TotBNum  IsLost  VioNum  Remark | Tinyint  Datetime  Datetime  Int  Int  Bool  Int  Text | 11  4  4  4 |  | 否  否  否  是  是  是  是  是 | 联系电话  登记日期  有效期至  现借图书数  累计借书数  是否挂失  违规次数  备注 |

（3）图书归还表

图书归还表是用于存储本系统中图书归还的记录。每个图书归还记录包括归还编号，管理员工号，图书编号，读者编号，归还时间。图书归还表的结构如表3.3所示。

表3.3 图书归还表(Book\_Return)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | ReID | Int | 4 | 主键 | 否 | 归还编号 |
| 2 | MID | Int | 4 | 外键 | 否 | 管理员工号 |
| 3 | BID | Int | 4 | 外键 | 否 | 图书编号 |
| 4 | RID | Int | 4 | 外键 | 否 | 读者编号 |
| 5 | ReTime | Datetime |  |  | 否 | 归还时间 |

（4）超期罚款表

超期罚款表是用于存储本系统中的罚款记录。读者丢失，损坏书籍后会产生罚款记录，并且记录读者是否交款。超期罚款表结构如表3.4所示。

表3.4 超期罚款表(Out\_Fine)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | FID | Int | 4 | 主键 | 否 | 罚款编号 |
| 2 | BID | Int | 4 | 外键 | 否 | 图书编号 |
| 3 | RID | Int | 4 | 外键 | 否 | 读者编号 |
| 4 | MID | Int | 4 | 外键 | 否 | 管理员工号 |
| 5  6  7  8 | FinePrice  IsFin  FineTime  Remark | Int  Bool  Datetime  text | 4 |  | 否  否  是  是 | 罚款金额  是否交款  罚款日期  备注 |

（5）管理员信息表

管理员信息表是用于存储本系统中管理员的信息。包括管理员工号，姓名，性别，联系电话，岗位。管理员信息表的结构如表3.5所示。

表3.5 管理员信息表(Manager\_Info)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | MID | Int | 4 | 主键 | 否 | 管理员工号 |
| 2 | MName | Varchar | 30 |  | 否 | 姓名 |
| 3 | Sex | Bit | 1 |  | 否 | 性别 |
| 4 | Phone | Tinyint | 11 |  | 否 | 联系电话 |
| 5 | Post | Varchar | 20 |  | 否 | 岗位 |

（6）图书借阅表

图书借阅表是用于存储本系统中图书借阅记录。读者的借阅操作会在表中产生数据。图书借阅表的结构如表3.6所示。

表3.6 图书借阅表(Book\_Borrow)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | BoID | Int | 4 | 主键 | 否 | 借阅编号 |
| 2 | BID | Int | 4 | 外键 | 否 | 图书编号 |
| 3 | RID | Int | 4 | 外键 | 否 | 读者编号 |
| 4 | MID | Int | 4 | 外键 | 否 | 管理员工号 |
| 5  6  7 | BoTime  ShouldReTime  ReNewTimes | Datetime  Datetime  int | 4 |  | 否  否  是 | 借阅时间  应还时间  续借次数 |

（7）图书丢失表

图书丢失表是用于存储图书丢失的记录。图书丢失表的结构如表3.7所示。

表3.7 图书丢失表(Book\_Lost)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | LID | Int | 4 | 主键 | 否 | 丢失编号 |
| 2 | BID | Int | 4 | 外键 | 否 | 图书编号 |
| 3 | RID | Int | 4 | 外键 | 否 | 读者编号 |
| 4 | MID | Int | 4 | 外键 | 否 | 管理员工号 |
| 5  6 | PayBack  OperTime | Int  datetime | 4 |  | 否  否 | 偿还金额  操作时间 |

（8）图书信息表

图书信息表是用于存储本系统中图书信息。与图书基本信息不同，图书信息主要描述图书在馆内的状态而非图书本身的信息，包括图书编号，ISBN，状态，入库时间。图书信息表的结构如表3.8所示。

表3.8 图书信息表(Book\_Info)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | BID | Int | 4 | 主键 | 否 | 图书编号 |
| 2 | ISBN | Varchar | 18 | 外键 | 否 | ISBN |
| 3 | BCondition | Bool |  |  | 否 | 状态 |
| 4 | WareTime | Datetime |  |  | 否 | 入库时间 |

（9）图书基本信息表

图书基本信息表是用于存储本系统中图书的基本信息。包括ISBN，出版社编号，书名，作者等。图书基本信息表的结构如表3.9所示。

表3.9 图书基本信息表(Book\_BasicInfo)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | ISBN | Varchar | 18 | 主键 | 否 | ISBN |
| 2 | PID | Int | 4 | 外键 | 否 | 出版社编号 |
| 3 | BName | Varchar | 30 |  | 否 | 书名 |
| 4 | Write | Varchar | 30 |  | 否 | 作者 |
| 5  6  7  8 | Version  PubDate  Price  Intro | Int  Date  Int  Text | 4  4 |  | 否  否  是  是 | 版次  出版日期  价格  简介 |

（10）出版社信息表

出版社信息表是用于存储本系统中出版社的信息。出版社信息表的结构如表3.10所示。

表3.10 出版社信息表(Press\_Info)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 列名 | 数据类型 | 长度 | 键值 | 允许空 | 说明 |
| 1 | PID | Int | 4 | 主键 | 否 | 出版社编号 |
| 2 | PName | Varchar | 30 |  | 否 | 出版社名称 |
| 3 | Contact | Varchar | 30 |  | 否 | 联系人 |
| 4 | Phone | Tinyint | 11 |  | 否 | 联系电话 |
| 5  6 | Fax  Address | Text  Varchar | 50 |  | 是  否 | 传真  地址 |

# 4 数据库SQL定义语句

根据数据库物理结构设计文档，分别编写用于MySQL和SQL Server的创建《图书馆管理系统》数据库和数据表的SQL定义语句。

## 4.1 MySQL定义语句

# 在创建表之前先删掉已有的表

drop table if exists Book\_BasicInfo;

drop table if exists Book\_Borrow;

drop table if exists Book\_Info;

drop table if exists Book\_Lost;

drop table if exists Book\_Return;

drop table if exists Manager\_Info;

drop table if exists Out\_Fine;

drop table if exists Press\_Info;

drop table if exists Reader\_Info;

drop table if exists Reader\_Type;

#创建图书基本信息表

/\*==============================================================\*/

/\* Table: Book\_BasicInfo \*/

/\*==============================================================\*/

create table Book\_BasicInfo

(

ISBN varchar(18) not null,

PID int not null,

BName varchar(30) not null,

Writer varchar(30) not null,

Version int not null,

PubDate date not null,

Price int,

Intro text,

primary key (ISBN)

);

#创建图书借阅表

/\*==============================================================\*/

/\* Table: Book\_Borrow \*/

/\*==============================================================\*/

create table Book\_Borrow

(

BoID int not null auto\_increment,

BID int,

RID int,

MID int,

BoTime datetime,

ShouldReTime datetime,

ReNewTimes int,

primary key (BoID)

);

#创建图书信息表

/\*==============================================================\*/

/\* Table: Book\_Info \*/

/\*==============================================================\*/

create table Book\_Info

(

BID int not null,

ISBN varchar(18) not null,

BCondition bool not null,

WareTime datetime not null,

primary key (BID)

);

#创建图书丢失表

/\*==============================================================\*/

/\* Table: Book\_Lost \*/

/\*==============================================================\*/

create table Book\_Lost

(

LID int not null auto\_increment,

BID int not null,

RID int not null,

MID int not null,

PayBack int not null,

OperTime datetime not null,

primary key (LID)

);

#创建图书归还表

/\*==============================================================\*/

/\* Table: Book\_Return \*/

/\*==============================================================\*/

create table Book\_Return

(

ReID int not null auto\_increment,

MID int not null,

BID int not null,

RID int not null,

ReTime datetime not null,

primary key (ReID)

);

#创建管理员信息表

/\*==============================================================\*/

/\* Table: Manager\_Info \*/

/\*==============================================================\*/

create table Manager\_Info

(

MID int not null,

MName varchar(30) not null,

Sex bit not null,

Phone tinyint not null,

Post varchar(20) not null,

primary key (MID)

);

#创建超期罚款表

/\*==============================================================\*/

/\* Table: Out\_Fine \*/

/\*==============================================================\*/

create table Out\_Fine

(

FID int not null auto\_increment,

BID int not null,

RID int not null,

MID int not null,

FinePrice int not null,

IsFin bool not null,

FineTime datetime,

Remark text,

primary key (FID)

);

#创建出版社信息表

/\*==============================================================\*/

/\* Table: Press\_Info \*/

/\*==============================================================\*/

create table Press\_Info

(

PID int not null,

PName varchar(30) not null,

Contact varchar(30) not null,

Phone tinyint not null,

Fax text,

Address varchar(50) not null,

primary key (PID)

);

#创建读者信息表

/\*==============================================================\*/

/\* Table: Reader\_Info \*/

/\*==============================================================\*/

create table Reader\_Info

(

RID int not null auto\_increment,

RTID int not null,

RName varchar(30) not null,

Sex bit not null,

Phone tinyint not null,

ReDate datetime not null,

ValDate datetime not null,

CurBNum int,

TotBNum int,

IsLost bool,

VioNum int,

Remark text,

primary key (RID)

);

#创建读者类别表

/\*==============================================================\*/

/\* Table: Reader\_Type \*/

/\*==============================================================\*/

create table Reader\_Type

(

RTID int not null,

RTName varchar(30) not null,

BNum int not null,

BDay int not null,

primary key (RTID)

);

#设置图书基本信息表外键PID

alter table Book\_BasicInfo add constraint FK\_Reference\_2 foreign key (PID)

references Press\_Info (PID) on delete restrict on update restrict;

#设置图书借阅表外键BID

alter table Book\_Borrow add constraint FK\_Reference\_4 foreign key (BID)

references Book\_Info (BID) on delete restrict on update restrict;

#设置图书借阅表外键RID

alter table Book\_Borrow add constraint FK\_Reference\_5 foreign key (RID)

references Reader\_Info (RID) on delete restrict on update restrict;

#设置图书借阅表外键MID

alter table Book\_Borrow add constraint FK\_Reference\_6 foreign key (MID)

references Manager\_Info (MID) on delete restrict on update restrict;

#设置图书信息表外键ISBN

alter table Book\_Info add constraint FK\_Reference\_3 foreign key (ISBN)

references Book\_BasicInfo (ISBN) on delete restrict on update restrict;

#设置图书丢失表外键BID

alter table Book\_Lost add constraint FK\_Reference\_13 foreign key (BID)

references Book\_Info (BID) on delete restrict on update restrict;

#设置图书丢失表外键RID

alter table Book\_Lost add constraint FK\_Reference\_14 foreign key (RID)

references Reader\_Info (RID) on delete restrict on update restrict;

#设置图书丢失表外键MID

alter table Book\_Lost add constraint FK\_Reference\_15 foreign key (MID)

references Manager\_Info (MID) on delete restrict on update restrict;

#设置图书归还表外键MID

alter table Book\_Return add constraint FK\_Reference\_7 foreign key (MID)

references Manager\_Info (MID) on delete restrict on update restrict;

#设置图书归还表外键BID

alter table Book\_Return add constraint FK\_Reference\_8 foreign key (BID)

references Book\_Info (BID) on delete restrict on update restrict;

#设置图书归还表外键RID

alter table Book\_Return add constraint FK\_Reference\_9 foreign key (RID)

references Reader\_Info (RID) on delete restrict on update restrict;

#设置超期罚款表外键BID

alter table Out\_Fine add constraint FK\_Reference\_10 foreign key (BID)

references Book\_Info (BID) on delete restrict on update restrict;

#设置超期罚款表外键RID

alter table Out\_Fine add constraint FK\_Reference\_11 foreign key (RID)

references Reader\_Info (RID) on delete restrict on update restrict;

#设置超期罚款表外键MID

alter table Out\_Fine add constraint FK\_Reference\_12 foreign key (MID)

references Manager\_Info (MID) on delete restrict on update restrict;

#设置读者信息表外键RTID

alter table Reader\_Info add constraint FK\_Reference\_1 foreign key (RTID)

references Reader\_Type (RTID) on delete restrict on update restrict;

## 4.2 SQL Server定义语句

/\*在创建表之前先删掉已有的表\*/

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_BasicInfo') and o.name = 'FK\_BOOK\_BAS\_REFERENCE\_PRESS\_IN')

alter table Book\_BasicInfo

drop constraint FK\_BOOK\_BAS\_REFERENCE\_PRESS\_IN

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Borrow') and o.name = 'FK\_BOOK\_BOR\_REFERENCE\_BOOK\_INF')

alter table Book\_Borrow

drop constraint FK\_BOOK\_BOR\_REFERENCE\_BOOK\_INF

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Borrow') and o.name = 'FK\_BOOK\_BOR\_REFERENCE\_READER\_I')

alter table Book\_Borrow

drop constraint FK\_BOOK\_BOR\_REFERENCE\_READER\_I

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Borrow') and o.name = 'FK\_BOOK\_BOR\_REFERENCE\_MANAGER\_')

alter table Book\_Borrow

drop constraint FK\_BOOK\_BOR\_REFERENCE\_MANAGER\_

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Info') and o.name = 'FK\_BOOK\_INF\_REFERENCE\_BOOK\_BAS')

alter table Book\_Info

drop constraint FK\_BOOK\_INF\_REFERENCE\_BOOK\_BAS

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Lost') and o.name = 'FK\_BOOK\_LOS\_REFERENCE\_BOOK\_INF')

alter table Book\_Lost

drop constraint FK\_BOOK\_LOS\_REFERENCE\_BOOK\_INF

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Lost') and o.name = 'FK\_BOOK\_LOS\_REFERENCE\_READER\_I')

alter table Book\_Lost

drop constraint FK\_BOOK\_LOS\_REFERENCE\_READER\_I

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Lost') and o.name = 'FK\_BOOK\_LOS\_REFERENCE\_MANAGER\_')

alter table Book\_Lost

drop constraint FK\_BOOK\_LOS\_REFERENCE\_MANAGER\_

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Return') and o.name = 'FK\_BOOK\_RET\_REFERENCE\_MANAGER\_')

alter table Book\_Return

drop constraint FK\_BOOK\_RET\_REFERENCE\_MANAGER\_

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Return') and o.name = 'FK\_BOOK\_RET\_REFERENCE\_BOOK\_INF')

alter table Book\_Return

drop constraint FK\_BOOK\_RET\_REFERENCE\_BOOK\_INF

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Book\_Return') and o.name = 'FK\_BOOK\_RET\_REFERENCE\_READER\_I')

alter table Book\_Return

drop constraint FK\_BOOK\_RET\_REFERENCE\_READER\_I

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Out\_Fine') and o.name = 'FK\_OUT\_FINE\_REFERENCE\_BOOK\_INF')

alter table Out\_Fine

drop constraint FK\_OUT\_FINE\_REFERENCE\_BOOK\_INF

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Out\_Fine') and o.name = 'FK\_OUT\_FINE\_REFERENCE\_READER\_I')

alter table Out\_Fine

drop constraint FK\_OUT\_FINE\_REFERENCE\_READER\_I

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Out\_Fine') and o.name = 'FK\_OUT\_FINE\_REFERENCE\_MANAGER\_')

alter table Out\_Fine

drop constraint FK\_OUT\_FINE\_REFERENCE\_MANAGER\_

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('Reader\_Info') and o.name = 'FK\_READER\_I\_REFERENCE\_READER\_T')

alter table Reader\_Info

drop constraint FK\_READER\_I\_REFERENCE\_READER\_T

go

if exists (select 1

from sysobjects

where id = object\_id('Book\_BasicInfo')

and type = 'U')

drop table Book\_BasicInfo

go

if exists (select 1

from sysobjects

where id = object\_id('Book\_Borrow')

and type = 'U')

drop table Book\_Borrow

go

if exists (select 1

from sysobjects

where id = object\_id('Book\_Info')

and type = 'U')

drop table Book\_Info

go

if exists (select 1

from sysobjects

where id = object\_id('Book\_Lost')

and type = 'U')

drop table Book\_Lost

go

if exists (select 1

from sysobjects

where id = object\_id('Book\_Return')

and type = 'U')

drop table Book\_Return

go

if exists (select 1

from sysobjects

where id = object\_id('Manager\_Info')

and type = 'U')

drop table Manager\_Info

go

if exists (select 1

from sysobjects

where id = object\_id('Out\_Fine')

and type = 'U')

drop table Out\_Fine

go

if exists (select 1

from sysobjects

where id = object\_id('Press\_Info')

and type = 'U')

drop table Press\_Info

go

if exists (select 1

from sysobjects

where id = object\_id('Reader\_Info')

and type = 'U')

drop table Reader\_Info

go

if exists (select 1

from sysobjects

where id = object\_id('Reader\_Type')

and type = 'U')

drop table Reader\_Type

go

/\*创建图书基本信息表\*/

/\*==============================================================\*/

/\* Table: Book\_BasicInfo \*/

/\*==============================================================\*/

create table Book\_BasicInfo (

ISBN varchar(18) not null,

PID int not null,

BName varchar(30) not null,

Writer varchar(30) not null,

Version int not null,

PubDate date not null,

Price int null,

Intro text null,

constraint PK\_BOOK\_BASICINFO primary key nonclustered (ISBN)

)

go

/\*创建图书借阅表\*/

/\*==============================================================\*/

/\* Table: Book\_Borrow \*/

/\*==============================================================\*/

create table Book\_Borrow (

BoID int identity,

BID int null,

RID int null,

MID int null,

BoTime datetime null,

ShouldReTime datetime null,

ReNewTimes int null,

constraint PK\_BOOK\_BORROW primary key nonclustered (BoID)

)

go

/\*创建图书信息表\*/

/\*==============================================================\*/

/\* Table: Book\_Info \*/

/\*==============================================================\*/

create table Book\_Info (

BID int not null,

ISBN varchar(18) not null,

BCondition bool not null,

WareTime datetime not null,

constraint PK\_BOOK\_INFO primary key nonclustered (BID)

)

go

/\*创建图书丢失表\*/

/\*==============================================================\*/

/\* Table: Book\_Lost \*/

/\*==============================================================\*/

create table Book\_Lost (

LID int identity,

BID int not null,

RID int not null,

MID int not null,

PayBack int not null,

OperTime datetime not null,

constraint PK\_BOOK\_LOST primary key nonclustered (LID)

)

go

/\*创建图书归还表\*/

/\*==============================================================\*/

/\* Table: Book\_Return \*/

/\*==============================================================\*/

create table Book\_Return (

ReID int identity,

MID int not null,

BID int not null,

RID int not null,

ReTime datetime not null,

constraint PK\_BOOK\_RETURN primary key nonclustered (ReID)

)

go

/\*创建管理员信息表\*/

/\*==============================================================\*/

/\* Table: Manager\_Info \*/

/\*==============================================================\*/

create table Manager\_Info (

MID int not null,

MName varchar(30) not null,

Sex bit not null,

Phone tinyint not null,

Post varchar(20) not null,

constraint PK\_MANAGER\_INFO primary key nonclustered (MID)

)

go

/\*创建超期罚款表\*/

/\*==============================================================\*/

/\* Table: Out\_Fine \*/

/\*==============================================================\*/

create table Out\_Fine (

FID int identity,

BID int not null,

RID int not null,

MID int not null,

FinePrice int not null,

IsFin bool not null,

FineTime datetime null,

Remark text null,

constraint PK\_OUT\_FINE primary key nonclustered (FID)

)

go

/\*创建出版社信息表\*/

/\*==============================================================\*/

/\* Table: Press\_Info \*/

/\*==============================================================\*/

create table Press\_Info (

PID int not null,

PName varchar(30) not null,

Contact varchar(30) not null,

Phone tinyint not null,

Fax text null,

Address varchar(50) not null,

constraint PK\_PRESS\_INFO primary key nonclustered (PID)

)

go

/\*创建读者信息表\*/

/\*==============================================================\*/

/\* Table: Reader\_Info \*/

/\*==============================================================\*/

create table Reader\_Info (

RID int identity,

RTID int not null,

RName varchar(30) not null,

Sex bit not null,

Phone tinyint not null,

ReDate datetime not null,

ValDate datetime not null,

CurBNum int null,

TotBNum int null,

IsLost bool null,

VioNum int null,

Remark text null,

constraint PK\_READER\_INFO primary key nonclustered (RID)

)

go

/\*创建读者类别表\*/

/\*==============================================================\*/

/\* Table: Reader\_Type \*/

/\*==============================================================\*/

create table Reader\_Type (

RTID int not null,

RTName varchar(30) not null,

BNum int not null,

BDay int not null,

constraint PK\_READER\_TYPE primary key nonclustered (RTID)

)

go

/\*设置图书基本信息表外键PID\*/

alter table Book\_BasicInfo

add constraint FK\_BOOK\_BAS\_REFERENCE\_PRESS\_IN foreign key (PID)

references Press\_Info (PID)

go

/\*设置图书借阅表外键BID\*/

alter table Book\_Borrow

add constraint FK\_BOOK\_BOR\_REFERENCE\_BOOK\_INF foreign key (BID)

references Book\_Info (BID)

go

/\*设置图书借阅表外键RID\*/

alter table Book\_Borrow

add constraint FK\_BOOK\_BOR\_REFERENCE\_READER\_I foreign key (RID)

references Reader\_Info (RID)

go

/\*设置图书借阅表外键MID\*/

alter table Book\_Borrow

add constraint FK\_BOOK\_BOR\_REFERENCE\_MANAGER\_ foreign key (MID)

references Manager\_Info (MID)

go

/\*设置图书信息表外键ISBN\*/

alter table Book\_Info

add constraint FK\_BOOK\_INF\_REFERENCE\_BOOK\_BAS foreign key (ISBN)

references Book\_BasicInfo (ISBN)

go

/\*设置图书丢失表外键BID\*/

alter table Book\_Lost

add constraint FK\_BOOK\_LOS\_REFERENCE\_BOOK\_INF foreign key (BID)

references Book\_Info (BID)

go

/\*设置图书丢失表外键RID\*/

alter table Book\_Lost

add constraint FK\_BOOK\_LOS\_REFERENCE\_READER\_I foreign key (RID)

references Reader\_Info (RID)

go

/\*设置图书丢失表外键MID\*/

alter table Book\_Lost

add constraint FK\_BOOK\_LOS\_REFERENCE\_MANAGER\_ foreign key (MID)

references Manager\_Info (MID)

go

/\*设置图书归还表外键MID\*/

alter table Book\_Return

add constraint FK\_BOOK\_RET\_REFERENCE\_MANAGER\_ foreign key (MID)

references Manager\_Info (MID)

go

/\*设置图书归还表外键BID\*/

alter table Book\_Return

add constraint FK\_BOOK\_RET\_REFERENCE\_BOOK\_INF foreign key (BID)

references Book\_Info (BID)

go

/\*设置图书归还表外键RID\*/

alter table Book\_Return

add constraint FK\_BOOK\_RET\_REFERENCE\_READER\_I foreign key (RID)

references Reader\_Info (RID)

go

/\*设置超期罚款表外键BID\*/

alter table Out\_Fine

add constraint FK\_OUT\_FINE\_REFERENCE\_BOOK\_INF foreign key (BID)

references Book\_Info (BID)

go

/\*设置超期罚款表外键RID\*/

alter table Out\_Fine

add constraint FK\_OUT\_FINE\_REFERENCE\_READER\_I foreign key (RID)

references Reader\_Info (RID)

go

/\*设置超期罚款表外键MID\*/

alter table Out\_Fine

add constraint FK\_OUT\_FINE\_REFERENCE\_MANAGER\_ foreign key (MID)

references Manager\_Info (MID)

go

/\*设置读者信息表外键RTID\*/

alter table Reader\_Info

add constraint FK\_READER\_I\_REFERENCE\_READER\_T foreign key (RTID)

references Reader\_Type (RTID)

go