**M 2023-2024学年第一学期**

**《数据库系统实验》实验报告 (3)**

|  |  |  |  |
| --- | --- | --- | --- |
| 实验序号 | 第10次 | 所在学院 | 软件工程学院 |
| 实验名称 | 关系数据库设计 | 任课教师 | 陈建国 |
| 实验地点 | A305 | 年级班级 | 2021级 |
| 学号 | 21311138 | 学生姓名 | 叶文熙 |

**一、实验目的及内容**

【实验目的】

1. 掌握关系数据库设计的基本工作流程；
2. 掌握数据库概念结构设计和逻辑结构设计的原理和流程；
3. 掌握数据库设计规范化（第一范式、第二范式、第三范式）；
4. 掌握数据库物理结构设计的原理和流程。

【实验内容】

1. 完成《图书馆管理系统》数据库的概念设计和逻辑设计；
2. 使用 Power Designer 设计《图书馆管理系统》数据库物理结构，包括数据表、列属性；
3. 设置表格主键约束、外键约束；
4. 使用 Power Designer 生成数据库 SQL 代码。；

【实验准备】

1. 安装Microsoft Visio设计工具或者使用上线设计网站http://draw.io。
2. 安装 Power Designer 数据库设计工具。
3. 关系数据库设计理论知识。

**二、实验步骤**

# 1 《图书馆管理系统》数据库概念设计

项目背景：某高校图书馆准备建设《图书馆管理系统》，主要用户角色和核心业务分析如下。

|  |
| --- |
| 用户角色：  1）读者：有三类：  教师：一次能借20本书，能借60天；  职工：一次能借15本书，能借60天；  学生：一次能借10本书，能借30天；  2）管理员：图书馆工作人员，负责办理图书借阅归还等业务。  核心业务：  （1）图书借阅：读者到图书馆借书；  （2）图书归还：读者到图书馆还书；  （3）超期罚款：借书超期未还要罚款，并记入个人违规记录；  （4）图书丢失：读者丢失图书，要赔钱，并记入个人违规记录。  (以上业务都要由管理员经办) |

根据系统需求分析和数据字典分析结果，使用理论课程第15讲的关系数据库设计知识，绘制系统数据库的概念结构，并绘制出各个实体和联系的具体属性信息。

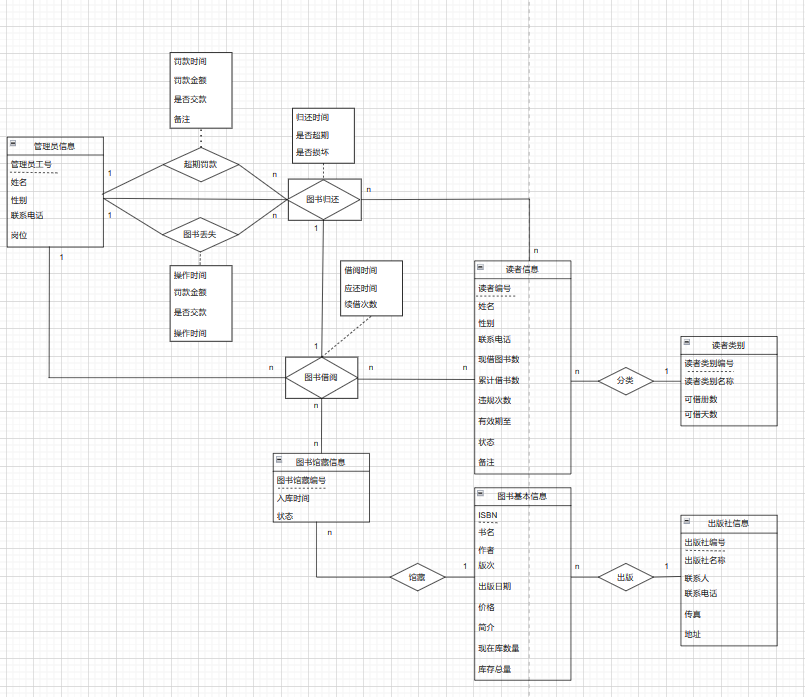


图2.1 《图书馆管理系统》E-R图

## 2 《图书馆管理系统》数据库逻辑结构设计

根据系统概念设计结果，撰写系统数据库的逻辑结构，将E-R模型转化为关系模型，设计具体的属性，以及实体之间、属性之间的联系。

**（1）关系模式转化**：根据6种转化方法（强实体集转化、弱实体集转化、联系集转化、复合属性及多值属性转化、类层次转化、聚合转化），将E-R图转化为关系模式。（根据需要，可以定义新的实体）。

读者类别（**读者类别编号**，读者类别名称，可借册数，可借天数）

读者信息（**读者编号**，*读者类别编号*，学号，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）

外键：*读者类别编号*

图书基本信息表（**ISBN**，*书籍类别编号，*书名，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）

外键：*出版社编号，书籍类别编号*

图书类别信息表（**书籍类别编号**，书籍类别名称）

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）

图书馆藏信息（**图书编号**，*ISBN*，入库时间，图书状态）

外键：*ISBN*

管理员信息（**管理员工号**，姓名，性别，联系电话，岗位）

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）*

外键：*图书编号*，*读者编号，管理员工号*

图书归还（**归还编号**，*借阅编号*，*读者编号*，归还时间，是否超期，是否损坏，是否丢失，*管理员工号）*

外键：*图书编号*，*读者编号，管理员工号*

超期罚款（**罚款编号**，*归还编号*，罚款金额，是否交款，罚款日期，*管理员工号*，备注）

外键：*归还记录编号，读者编号，管理员工号*

图书损失（**损失编号**，*归还编号*，罚款金额，操作时间，是否交款，*管理员工号*，备注）

外键：*归还记录编号，读者编号，管理员工号*

图书丢失（**丢失编号**，*归还编号*，罚款金额，操作时间，是否交款，*管理员工号*，备注）

外键：*归还记录编号，读者编号，管理员工号*

**（2）关系模式优化**：依次考虑第一范式、第二范式、第三范式。

**第一范式：（修改的部分）**

出版社信息（**出版社编号**，出版社名称，联系人，联系电话，传真，省，市，街道地址）

**第二范式：**满足第二范式！

**第三范式：**满足第三范式！

# 3 《图书馆管理系统》数据库物理结构设计

根据逻辑结构设计方案，使用Power Designer工具进行物理结构设计。提供物理结构设计图，并对**每一张表的具体结构**分别进行阐述。



图3.1 《图书馆管理系统》物理结构设计图

**（1）读者信息表**

高校图书馆的读者用户包括广大学生、教师和职工。读者信息表的示例如表3.1所示。

表3.1 读者信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| RID | int |  | 是 |  | 否 | 读者编号 |
| RTNo | Int(4) | 4 |  | 是 | 否 | 读者类型编号 |
| SNo | Char(8) | 8 |  |  | 否 | 学号 |
| RName | Varchar(40) | 40 |  |  | 否 | 读者姓名 |
| Sex | Tinyint(1) | 1 |  |  |  | 读者性别 |
| Rtelephone | Char(11) | 11 |  |  |  | 读者联系电话 |
| RDTime | datetime |  |  |  | 否 | 登记日期 |
| RVDTime | Datetime |  |  |  | 否 | 有效期至 |
| Rreads | int |  |  |  | 否 | 现借图书数 |
| RTreads | int |  |  |  | 否 | 累计借书数 |
| RLstatus | Tinyint(1) |  |  |  | 否 | 是否挂失 |
| RFTimes | int |  |  |  | 否 | 违规次数 |
| RNotion | text |  |  |  |  | 备注 |
| RPwd | Varchar(11) |  |  |  | 否 | 密码 |

**（2）读者类别表**

表3.2 读者类型表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| RTNo | Int(4) | 4 | 是 |  | 否 | 读者类别编号 |
| RTName | Varchar(40) | 40 |  |  | 否 | 读者类别名称 |
| RAreads | int |  |  |  | 否 | 可借册数 |
| RAdays | int |  |  |  | 否 | 可借天数 |

**（3）图书基本信息表**

表3.3 图书基本信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 精度 | 主键 | 外键 | 是否为空 | 字段说明 |
| ISBN | varchar | 13 |  | 是 | 否 | 否 | 国际标准书号 |
| PID | int |  |  |  | 是 |  | 出版社编号 |
| BName | varchar | 255 |  |  |  | 否 | 书名 |
| BTNo | int |  |  |  |  | 否 | 图书类型编号 |
| BWriter | varchar | 30 |  |  |  |  | 作者 |
| BEdition | varchar | 30 |  |  |  |  | 版次 |
| BPDate | varchar |  |  |  |  |  | 出版日期 |
| BPrice | float | 10, | 2 |  |  | 否 | 价格 |
| BIntro | text |  |  |  |  |  | 简介 |
| BExnum | int |  |  |  |  | 否 | 现在库数量 |
| BTotal | int |  |  |  |  | 否 | 库存总量 |

**（4）图书馆藏信息表**

表3.4 图书馆藏信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| BSNo | int |  | 是 |  | 否 | 馆藏图书编号 |
| ISBN | varchar | 13 |  | 是 | 否 | 国际标准书号 |
| BSDate | date |  |  |  | 否 | 入库时间 |
| BSStatus | tinyint | 2 |  |  | 否 | 图书状态(1表示借出，0表示在库，2表示丢失，3表示损坏) |

**（5）出版社信息表**

表3.5 出版社信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| PID | int |  | 是 |  | 否 | 出版社的唯一标识 |
| PName | varchar | 60 |  |  | 否 | 出版社名称 |
| PLinkman | varchar | 30 |  |  |  | 联系人 |
| Ptelephone | char | 11 |  |  |  | 联系电话 |
| PFax | char | 8 |  |  |  | 传真 |
| PProvince | varchar | 20 |  |  |  | 地址：省 |
| PCity | varchar | 40 |  |  |  | 地址：市 |
| PAddr | varchar | 60 |  |  |  | 地址：详细地址 |

**（6）管理员信息表**

表3.6 管理员信息表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| MID | int |  | 是 |  | 否 | 管理员工号的唯一标识 |
| MName | varchar | 30 |  |  | 否 | 姓名 |
| MSex | tinyint | 1 |  |  |  | 性别 |
| Mtelephone | char | 11 |  |  | 否 | 联系电话 |
| MJob | varchar | 30 |  |  | 否 | 岗位 |
| MPwd | varchar | 16 |  |  | 否 | 管理员密码 |

**（7）图书借阅表**

表3.7 图书借阅表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| BrNo | int |  | 是 |  | 否 | 借阅编号的唯一标识 |
| BSNo | int |  |  | 是 | 否 | 图书编号的外键 |
| SNo | int |  |  | 是 | 否 | 读者学号/教职工号的外键 |
| MID | int |  |  | 是 | 是 | 管理员工号的外键 |
| BrDtime | Datetime |  |  |  | 否 | 借阅时间 |
| BgDTime | datetime |  |  |  | 否 | 应还时间 |
| BrTimes | int |  |  |  | 否 | 续借次数 |
| BrStatus | Tinyint(1) |  |  |  | 否 | 办理状态 |
| BrtStatus | Tinyint(1) |  |  |  | 否 | 是否归还 |

**（8）图书归还表**

表3.8 图书归还表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 主键 | 外键 | 是否为空 | 字段说明 |
| RtNo | int |  | 是 |  | 否 | 归还编号的唯一标识 |
| BrNo | int |  |  | 是 | 否 | 借阅编号的外键 |
| SNo | int |  |  | 是 | 否 | 读者编号的外键 |
| MID | int |  |  | 是 | 否 | 管理员工号的外键 |
| RtDTime | Datetime |  |  |  | 否 | 归还时间 |
| BtStatus | Tinyint(1) |  |  |  | 否 | 办理状态 |
| RtOverdue | bool |  |  |  | 否 | 是否超期 |
| OverdueDays | int |  |  |  | 否 | 超期时间 |
| RtDamaged | bool |  |  |  | 否 | 是否损坏 |
| RtLost | bool |  |  |  | 否 | 是否丢失 |

**（9）超期罚款表**

表3.9 **超期罚款**表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 精度 | 主键 | 外键 | 是否为空 | 字段说明 |
| ONo | int |  |  | 是 |  | 否 | 罚款编号的唯一标识 |
| RtNo | int |  |  |  | 是 | 否 | 归还记录编号的外键 |
| MID | int |  |  |  | 是 | 否 | 管理员工号的外键 |
| OFines | float | 10 | 2 |  |  | 否 | 罚款金额 |
| OFDTime | datetime |  |  |  |  | 否 | 罚款时间 |
| OFStatus | bool |  |  |  |  | 否 | 是否交款 |
| ONotion | text |  |  |  |  |  | 备注 |

**（10）图书丢失表**

表3.10 图书丢失表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 精度 | 主键 | 外键 | 是否为空 | 字段说明 |
| LNo | int |  |  | 是 |  | 否 | 丢失编号的唯一标识 |
| RtNo | int |  |  |  | 是 | 否 | 归还记录编号的外键 |
| MID | int |  |  |  | 是 | 否 | 管理员工号的外键 |
| LFines | float | 10 | 2 |  |  | 否 | 罚款金额 |
| LFDTime | datetime |  |  |  |  | 否 | 罚款时间 |
| LFStatus | bool |  |  |  |  | 否 | 是否交款 |
| LNotion | text |  |  |  |  |  | 备注 |

**（11）图书损失表**

表3.11 图书损失表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 数据类型 | 字段长度 | 精度 | 主键 | 外键 | 是否为空 | 字段说明 |
| DNo | int |  |  | 是 |  | 否 | 损失编号的唯一标识 |
| RtNo | int |  |  |  | 是 | 否 | 归还记录编号的外键 |
| MID | int |  |  |  | 是 | 否 | 管理员工号的外键 |
| DFines | float | 10 | 2 |  |  | 否 | 罚款金额 |
| DFDTime | datetime |  |  |  |  | 否 | 罚款时间 |
| DFStatus | bool |  |  |  |  | 否 | 是否交款 |
| DNotion | text |  |  |  |  |  | 备注 |

# 4 生成数据库SQL语句

关系数据库物理结构设计完成之后，将表结构导出成SQL数据定义语句。

|  |
| --- |
| -/\*==============================================================\*/  /\* DBMS name: MySQL 5.0 \*/  /\* Created on: 2023/11/8 22:54:53 \*/  /\*==============================================================\*/  drop table if exists Book\_Info;  drop table if exists Book\_StoreInfo;  drop table if exists Book\_TypeInfo;  drop table if exists Borrow\_Info;  drop table if exists Damaged\_Info;  drop table if exists Fines\_Info;  drop table if exists Lost\_Info;  drop table if exists Manager\_Info;  drop table if exists Publisher\_Info;  drop table if exists Reader\_Info;  drop table if exists Reader\_Type;  drop table if exists Retrun\_Info;  /\*==============================================================\*/  /\* Table: Book\_Info \*/  /\*==============================================================\*/  create table Book\_Info  (  ISBN varchar(13) not null,  BTNo int not null,  PID int,  BName varchar(255) not null,  BWriter varchar(30),  BEdition varchar(30),  BPDate date,  BPrice float(10,2) not null,  BExnum int not null,  BIntro text,  BTotal int not null,  primary key (ISBN)  );  /\*==============================================================\*/  /\* Table: Book\_StoreInfo \*/  /\*==============================================================\*/  create table Book\_StoreInfo  (  BSNo int not null auto\_increment,  ISBN varchar(13) not null,  BSDate date not null,  BSStatus tinyint(2) not null,  primary key (BSNo)  );  /\*==============================================================\*/  /\* Table: Book\_TypeInfo \*/  /\*==============================================================\*/  create table Book\_TypeInfo  (  BTNo int not null auto\_increment,  BTName varchar(60) not null,  primary key (BTNo)  );  /\*==============================================================\*/  /\* Table: Borrow\_Info \*/  /\*==============================================================\*/  create table Borrow\_Info  (  BrNo int not null auto\_increment,  MID int not null,  BSNo int not null,  RID int not null,  BrDtime datetime not null,  BgDTime datetime not null,  BrTimes int not null,  primary key (BrNo)  );  /\*==============================================================\*/  /\* Table: Damaged\_Info \*/  /\*==============================================================\*/  create table Damaged\_Info  (  DNo int not null auto\_increment,  RtNo int not null,  MID int not null,  RID int not null,  DFines float(10,2) not null,  DFStatus bool not null,  DFDTime datetime not null,  DNotion text,  primary key (DNo)  );  /\*==============================================================\*/  /\* Table: Fines\_Info \*/  /\*==============================================================\*/  create table Fines\_Info  (  ONo int not null auto\_increment,  RtNo int not null,  MID int not null,  RID int not null,  OFines float(10,2) not null,  OFStatus bool not null,  OFDTime datetime not null,  ONotion text,  primary key (ONo)  );  /\*==============================================================\*/  /\* Table: Lost\_Info \*/  /\*==============================================================\*/  create table Lost\_Info  (  LNo int not null auto\_increment,  RtNo int not null,  MID int not null,  RID int not null,  LFines float(10,2) not null,  LFStatus bool not null,  LFDTime datetime not null,  LNotion text,  primary key (LNo)  );  /\*==============================================================\*/  /\* Table: Manager\_Info \*/  /\*==============================================================\*/  create table Manager\_Info  (  MID int not null auto\_increment,  MName varchar(30) not null,  MSex tinyint(1),  MTelephone char(11) not null,  MJob varchar(30) not null,  primary key (MID)  );  /\*==============================================================\*/  /\* Table: Publisher\_Info \*/  /\*==============================================================\*/  create table Publisher\_Info  (  PID int not null auto\_increment,  PName varchar(60) not null,  PLinkman varchar(30),  Ptelephone char(11),  PFax char(8),  PProvince varchar(20),  PCity varchar(40),  PAddr varchar(60),  primary key (PID)  );  /\*==============================================================\*/  /\* Table: Reader\_Info \*/  /\*==============================================================\*/  create table Reader\_Info  (  RID int not null auto\_increment,  RTNo int(4),  SNo char(8) not null,  RName varchar(40) not null,  Sex tinyint(1),  Rtelephone char(11),  RDTime datetime not null,  RVDTime datetime not null,  Rreads int not null,  RTreads int not null,  RLStatus bool not null,  RFTimes int not null,  RNotion text,  primary key (RID)  );  /\*==============================================================\*/  /\* Table: Reader\_Type \*/  /\*==============================================================\*/  create table Reader\_Type  (  RTNo int(4) not null auto\_increment,  RTName varchar(40) not null,  RAreads int not null,  RAdays int not null,  primary key (RTNo)  );  /\*==============================================================\*/  /\* Table: Retrun\_Info \*/  /\*==============================================================\*/  create table Retrun\_Info  (  RtNo int not null auto\_increment,  BrNo int not null,  RID int not null,  MID int not null,  RtDTime datetime not null,  RtOverdue bool not null,  RtDamaged bool not null,  RtLost bool not null,  primary key (RtNo)  );  alter table Book\_Info add constraint FK\_Reference\_2 foreign key (BTNo)  references Book\_TypeInfo (BTNo) on delete restrict on update restrict;  alter table Book\_Info add constraint FK\_Reference\_3 foreign key (PID)  references Publisher\_Info (PID) on delete restrict on update restrict;  alter table Book\_StoreInfo add constraint FK\_Reference\_4 foreign key (ISBN)  references Book\_Info (ISBN) on delete restrict on update restrict;  alter table Borrow\_Info add constraint FK\_Reference\_5 foreign key (MID)  references Manager\_Info (MID) on delete restrict on update restrict;  alter table Borrow\_Info add constraint FK\_Reference\_6 foreign key (BSNo)  references Book\_StoreInfo (BSNo) on delete restrict on update restrict;  alter table Borrow\_Info add constraint FK\_Reference\_7 foreign key (RID)  references Reader\_Info (RID) on delete restrict on update restrict;  alter table Damaged\_Info add constraint FK\_Reference\_15 foreign key (RtNo)  references Retrun\_Info (RtNo) on delete restrict on update restrict;  alter table Damaged\_Info add constraint FK\_Reference\_16 foreign key (MID)  references Manager\_Info (MID) on delete restrict on update restrict;  alter table Damaged\_Info add constraint FK\_Reference\_18 foreign key (RID)  references Reader\_Info (RID) on delete restrict on update restrict;  alter table Fines\_Info add constraint FK\_Reference\_11 foreign key (RtNo)  references Retrun\_Info (RtNo) on delete restrict on update restrict;  alter table Fines\_Info add constraint FK\_Reference\_12 foreign key (MID)  references Manager\_Info (MID) on delete restrict on update restrict;  alter table Lost\_Info add constraint FK\_Reference\_13 foreign key (RtNo)  references Retrun\_Info (RtNo) on delete restrict on update restrict;  alter table Lost\_Info add constraint FK\_Reference\_14 foreign key (MID)  references Manager\_Info (MID) on delete restrict on update restrict;  alter table Reader\_Info add constraint FK\_Reference\_1 foreign key (RTNo)  references Reader\_Type (RTNo) on delete restrict on update restrict;  alter table Retrun\_Info add constraint FK\_Reference\_10 foreign key (MID)  references Manager\_Info (MID) on delete restrict on update restrict;  alter table Retrun\_Info add constraint FK\_Reference\_8 foreign key (BrNo)  references Borrow\_Info (BrNo) on delete restrict on update restrict; |

# 5 实验报告要求

1. 实验报告内容：本次实验内容将和上一次实验内容（数据库概念设计和逻辑结构设计）一起合并提交，形成一份《图书馆管理系统数据库设计报告》。
2. 实验报告提交方式：坚果云收件箱。https://send2me.cn/fLaYiCc\_/SWqYz6utxYc1wg
3. 实验报告提交时间：2023-11-9日(下周四) 23:59