北京工业大学

2020- 2021学年 第一学期

软件学院

|  |  |  |  |
| --- | --- | --- | --- |
| 课程名称： | 数据库应用实验 | | |
| 报告性质： | □作业报告 实验报告 | | |
| 学号： | 18080216 | 姓名： | 齐子逸 |
| 任课教师： | 邵勇 | 课程性质： | 专业限选课 |
| 学分： | 2.0 | 学时： | 32 |
| 班级： | 180803 | 成绩： |  |
| 小组成员与分工： | 齐子逸18080216：SQL设计+SQL实现+实验报告  孙晨伟18080128：需求分析+前端设计+SQL测试  卜馨怡18080318：概念模型设计+ppt制作+报告总结 | | |
|  |  | | |

2020年 10 月 24 日

目 录

**1需求分析**．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．1

1.1需求分析过程．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．1

1.2数据字典．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．3

**2概念模式设计**．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．5

2.1实体．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．5

2.2 局部视图．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．5

2.3视图集成．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．7

**3逻辑模式设计**．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．8

3.1将E-R图转化为关系模式．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．8

3.2数据模型的优化．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．8

**4检验是否满足用户需求**．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．11

4.1调查用户需求．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．11

**5其它数据库对象(物理数据库设计)的考虑**．．．．．．．．．．．．．．．．．．．．．．．．．．．11

5.1建表．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．11

5.2合法用户名字、权限、角色．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．13

5.3视图．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．15

5.4触发器．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．16

5.5索引．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．17

**6数据库实施**．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．17

6.1表的建立与数据的载入．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．17

6.2数据库运行．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．．20

**1需求分析**

1.1需求分析过程

（1）流程

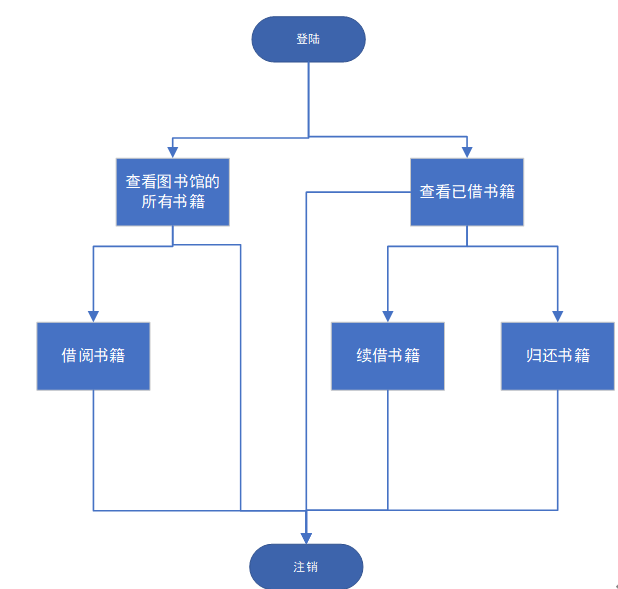
读者到图书借阅处查询图书，看看是否有满足读者要求的图书。如果有满足读者要求的图书，查看该图书是否在馆，如果在馆，则图书管理员在系统记录读者信息并记录借出图书信息，如果不在馆，通知读者。如果读者要求的图书无法提供，则通知读者无法满足。

(2)功能：

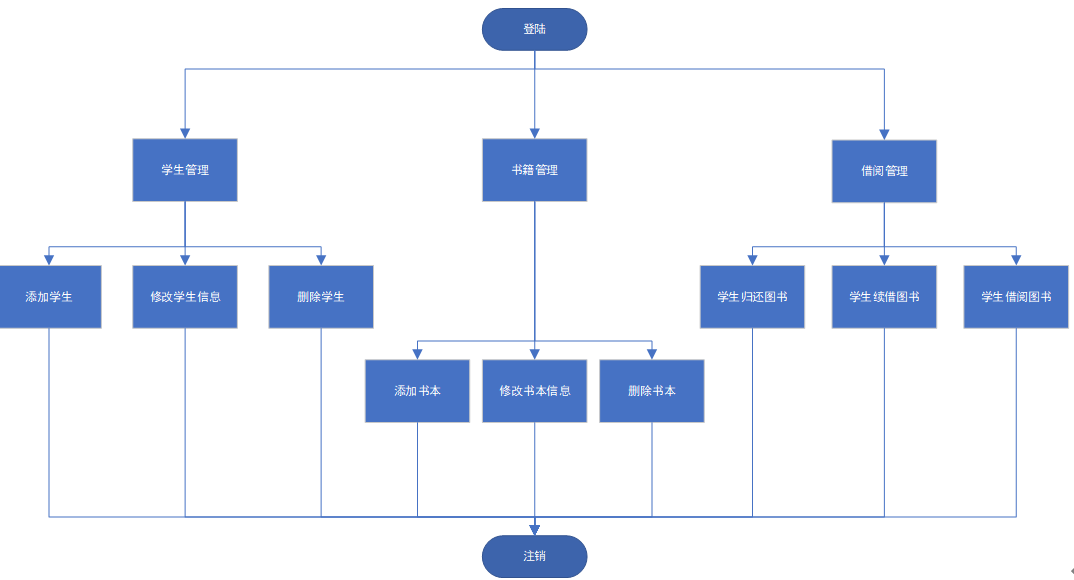
实现图书的查询，借入和归还功能，而且能保存记录。

(3)流程图：

读者操作流程：



图书管理员操作流程：



读者借阅图书，系统查询图书是否在馆。图书已借出，显示已借出；未借出显示下一步，图书管理员管理图书，登记读者信息，借出图书。

1.2数据字典：

（1）数据项：

表1.1

|  |
| --- |
| 数据项名 含义说明 数据类型 长度 与其他数据项的逻辑关系 |
| 读者学号 读者相关信息 字符型 20  读者姓名 读者相关信息 字符型 6  读者性别 读者相关信息 字符型 2  联系电话 读者相关信息 字符型 11  所在系部 读者相关信息 字符型 10  联系电话 读者相关信息 字符型 11  违章状况 读者相关信息 字符型 6  借书累计 读者相关信息 整数型 2  书 名 图书相关信息 字符型 20  ISBN 图书相关信息 字符型 20  作 者 图书相关信息 字符型 20  出版社 图书相关信息 字符型 20  出版日期 图书相关信息 日期  书籍分类 图书相关信息 字符型 20  在架数量 图书相关信息 整数型 4  管理员工作号 管理员相关信息 字符型 12  管理员姓名 管理员相关信息 字符型 6  管理员性别 管理员相关信息 字符型 2  借书日期 借书相关信息 日期  预计还书日期 换书相关信息 日期  实际还书日期 换书相关信息 日期 |

（2）数据结构：

|  |
| --- |
| 数据结构名 含义说明 组成 |
| 图书信息表 定义了图书的相关信息 ISBN,书名,作者,出版社,出版日期,书籍分类，在架数量  读者信息表 定义了读者有关信息 读者学号，读者姓名，读者性别，联系电话，所在系部  管理员信息表 定义了管理员有关信息 管理员工作号，姓名，性别，电话，工资  图书借阅表 定义了借阅图书的有关信息 读者学号,IBSN，借书日期,逾期归还时间  图书归还表 定义了归还图书的有关信息 读者学号,IBSN，借书日期,实际还书日期  管理员-书籍表 定义了管理员与书籍的有关信息 工作号,ISBN,添加时间,是否在馆  管理员-读者表 定义了管理员与学生的有关信息 工作号,读者学号,借还确认，违章状况，借书累计 |

表1.2

（3）数据流：

表1.3

|  |
| --- |
| 数据流名 来源 去向 |
| 读者信息 读者 图书管理员的管理处  图书信息 图书 图书管理员的管理处  借阅信息 借阅图书 图书管理员的管理处  归还信息 归还图书 图书管理员的管理处 |

（4）数据存储：

表1.4

|  |
| --- |
| 数据存储名 含义说明 组成 |
| 读者信息 存储了读者的相关信息 读者编号，姓名，性别，联系电话，系别  图书信息 存储了图书的相关信息 ISBN,书名,作者,出版社,出版日期,简介  借阅信息 存储了借阅图书的有关信息 ISBN,读者学号,借书日期,还书日期,预期还书日期  归还信息 存储了归还图书的有关信息 ISBN,读者学号,借书日期,实际还书日期 |

（5）处理过程：

表1.5

|  |
| --- |
| 处理过程名 含义说明 输入 输出 |
| 借书过程 查看图书是否被借出 图书信息 借阅信息  还书过程 查看借阅图书的信息（视图） 读者信息 图书信息 |

2概念模式设计

2.1实体：

读者信息，图书信息，图书管理员信息

2.2局部视图：

1. 读者借阅—归还E-R图

图 书

读 者

借 阅

学号

姓名

性别

所在系

归还日期

书名

出版社

出版日期

ISBN

简介

作者

1

n

n

1

图书归还表表

归 还

借阅日期

图书借阅表

图2-1借阅归还关系E-R图

事务规则：一个读者可以借阅归还多本图书（1：n）

一类图书可以被一个读者借阅归还(1：n)

（2）管理员—管理—读者

管理员

读 者

管 理

编号

姓名

性别

所在系

违章数

累计借书

借还确认

姓名

电话

工作号

工资

性别

1

n

图2-2管理员\_读者关系E-R图

事务规则：一个图书管理员可以管理多个读者信息（1：n）

一个读者信息只能被一个图书管理员管理（1：1）

（3）管理员—管理—图书

工作号

图 书

管理员

管 理

姓名

性别

工资

电话借书

是否在馆

添加时间

书名

出版社

出版日期

ISBN

数量

作者

m

n

图2-3管理员\_书籍关系E-R图

事务规则：一本图书可以被多个图书管理员管理（m：n）

一个图书管理员可以管理多本图书（m：n）

2.3视图集成

所有局部ER模式都设计好了后，接下来就是把它们综合成单一的全局概念结构。全局概念结构不仅要支持所有局部ER模式，而且必须合理地表示一个完整、一致的数据库概念结构。

1)确定公共实体类型

为了给多个局部ER模式的合并提供开始合并的基础，首先要确定各局部结构中的公共实体类型。在这一步中我们仅根据实体类型名和键来认定公共实体类型。一般把同名实体类型作为公共实体类型的一类候选，把具有相同键的实体类型作为公共实体类型的另一类候选。

2)局部ER模式的合并

合并的原则是：首先进行两两合并；先和合并那些现实世界中有联系的局部结构；合并从公共实体类型开始，最后再加入独立的局部结构。

3)消除冲突

冲突分为三类：属性冲突、结构冲突、命名冲突。

设计全局ER模式的目的不在于把若干局部ER模式形式上合并为一个ER模式，而在于消除冲突，使之成为能够被所有用户共同理解和接受的同一的概念模型。  
4)全局ER模式的优化

在得到全局ER模式后，为了提高数据库系统的效率，还应进一步依据处理需求对ER模式进行优化。一个好的全局ER模式，除能准确、全面地反映用户功能需求外，还应满足下列条件：实体类型的个数要尽可能的少；实体类型所含属性个数尽可能少；实体类型间联系无冗余。综上所述，“图书管理系统”的全局ER模式如下图所示。

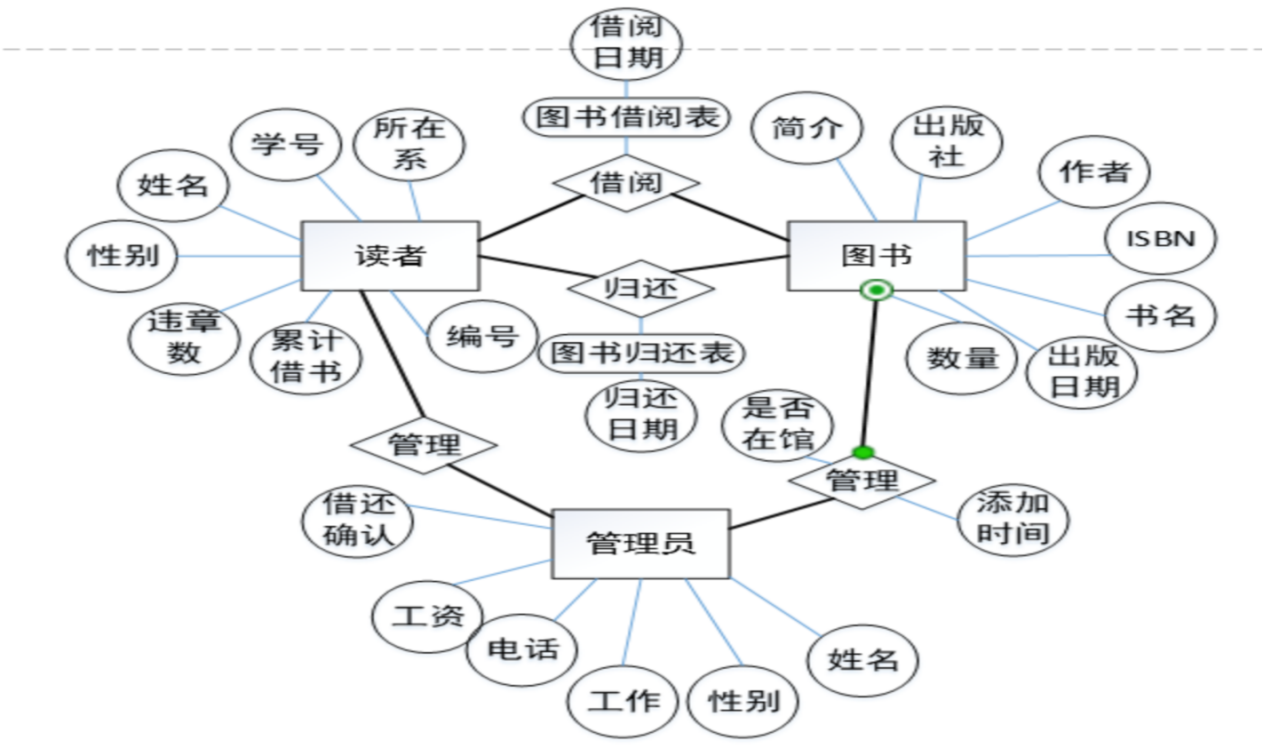


图2.4系统全局ER图

3逻辑模式设计

3.1将E-R图转化为关系模式

由于概念设计的结果是ER图，DBMS一般采用关系模型，因此数据库的逻辑设计过程就是把ER图转化为关系模式的过程。由于关系模型所具有的优点，逻辑设计可以充分运用关系数据库规范化理论，使设计过程形式化地进行。设计结果是一组关系模式的定义。

将图2-4总体概念结构E-R图转化成关系模型。

1、读者（读者编号，读者姓名，读者性别，联系电话，所在系，）

2、书籍**（**ISBN，书名，作者，出版社，出版日期，简介）

3、管理员（工作号**，**姓名，性别，电话，工资）

4、借阅（读者编号，ISBN，借书日期，预期还书日期）

5、归还（读者编号，ISBN，借书日期，实际还书日期）

6、管理员\_书籍（工作号，ISBN，添加时间，是否在馆）

7、管理员\_读者（工作号，读者编号，确认借还，违章状况，累计借书）

3.2数据模型的优化

将转化的关系模式进行优化，最终达到第三范式。

优化后的关系模式如下：

1、读者（读者编号，读者姓名，读者性别，联系电话，所在系）

2、书籍**（**ISBN，书名，作者，出版社，出版日期，简介）

3、管理员（工作号**，**姓名，性别，电话，工资）

4、借阅（读者编号，ISBN，借书日期，预期还书日期）

5、归还（读者编号，ISBN，借书日期，实际还书日期）

5、管理员\_书籍（工作号，ISBN，添加时间，是否在馆）

6、管理员\_读者（工作号，读者编号，确认借还，违章状况，累计借书）

表3.1 图书信息表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性  数据类型  长度  约束  描述 |
| ISBN char 20 NOT NULL 主键  书名 char 20 NOT NULL 非主属性  作者 char 20 NOT NULL 非主属性  出版社 char 20 非主属性  出版日期 date 非主属性  书籍分类 char 20 非主属性  在架数量 int 4 NOT NULL 非主属性 |

表3.2 读者信息表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性  数据类型  长度  约束  描述 |
| 读者学号 char 20 NOT NULL 主键  读者姓名 char 6 NOT NULL 非主属性  读者性别 char 2 NOT NULL 非主属性  所在系部 char 10 NOT NULL 非主属性  联系电话 char 11 NULL 非主属性 |

表3.3 管理员信息表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性  数据类型  长度  约束  描述 |
| 工作号 char 12 NOT NULL 主键  姓名 char 6 NOT NULL 非主属性  性别 char 2 NOT NULL 非主属性  电话 int 11 非主属性  工资 int 5 非主属性 |

表3.4 图书借阅表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性 |
| 读者学号 char 20 NOT NULL 主键  ISBN char 20 NOT NULL 主键  借书日期 date NOT NULL 非主属性  预期还书日期 date NOT NULL 非主属性 |

表3.5 图书归还表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性 |
| 读者学号 char 20 NOT NULL 主键  ISBN char 20 NOT NULL 主键  借书日期 date NOT NULL 非主属性  实际还书日期 date NOT NULL 非主属性 |

表3.6 管理员\_书籍表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性 |
| 工作号 char 12 NOT NULL 主键  ISBN char 20 NOT NULL 主键  添加时间 date 非主属性  是否在馆 char 2 NOT NULL 非主属性 |

表3.7 管理员\_学生表

|  |
| --- |
| 字段名称 数据类型 长度 约束 属性 |
| 工作号 char 12 NOT NULL 主键  读者学号 char 20 NOT NULL 主键  借还确认 char 4 NOT NULL 非主属性  违章状况 char 6 非主属性  累计借书 int 2 非主属性 |

4检验是否满足用户需求

4.1调查用户需求

①读者需求

1. 借阅功能：能够方便地借阅图书、续借图书、归还图书
2. 查询功能：

1）按图书编号查找图书

2）按书籍名称查找图书

3）按书籍作者查找图书

4）能够按照自己的图书证编号查询自己的状态信息

②图书管理员需求

（1）读者注册功能：能够对新的读者进行登记，或注销读者的信息信息。

（2）图书录入功能：能够将新到的图书信息输入到系统的图书信息库中。

（3）统计功能：

1）按图书编号统计书籍数量

2）按书籍名称统计数量

3）按作者统计书籍数量

5 数据库的物理设计

5.1建表

1. 图书信息表

CREATE TABLE `图书信息表`

(

`ISBN` char(20) CHARACTER SET latin1 NOT NULL,

`书名` char(20) NOT NULL,

`作者` char(20) NOT NULL,

`出版社` char(20) NOT NULL,

`出版日期` date DEFAULT NULL,

`书籍分类` char(20) NOT NULL,

`在架数量` int(4) NOT NULL,

PRIMARY KEY (`ISBN`),

UNIQUE KEY `书名` (`书名`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（2）读者信息表

CREATE TABLE `读者信息表`

(

`读者学号` char(20) CHARACTER SET latin1 NOT NULL,

`读者姓名` char(6) NOT NULL,

`读者性别` char(2) NOT NULL,

`所在系部` char(10) NOT NULL,

`联系电话` char(11) CHARACTER SET latin1 DEFAULT NULL,

PRIMARY KEY (`读者学号`),

UNIQUE KEY `读者姓名` (`读者姓名`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

(3) 管理员信息表

CREATE TABLE `管理员信息表`

(

`管理员工作号` char(12) CHARACTER SET latin1 NOT NULL,

`管理员姓名` char(6) NOT NULL,

`管理员性别` char(2) NOT NULL,

`管理员电话` int(11) DEFAULT NULL,

`工资` int(5) DEFAULT NULL,

PRIMARY KEY (`管理员工作号`),

UNIQUE KEY `姓名` (`管理员姓名`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（4）图书借阅表

CREATE TABLE `图书借阅表`

(

`读者学号` char(20) CHARACTER SET latin1 NOT NULL,

`ISBN` char(20) CHARACTER SET latin1 NOT NULL,

`借书日期` date NOT NULL,

`预期还书日期` date NOT NULL,

PRIMARY KEY (`读者学号`,`ISBN`),

KEY `ISBN` (`ISBN`),

CONSTRAINT `图书借阅表\_ibfk\_1` FOREIGN KEY (`读者学号`) REFERENCES `读者信息表` (`读者学号`),

CONSTRAINT `图书借阅表\_ibfk\_2` FOREIGN KEY (`ISBN`) REFERENCES `图书信息表` (`ISBN`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（4）图书归还表

CREATE TABLE `图书归还表`

(

`读者学号` char(20) CHARACTER SET latin1 NOT NULL,

`ISBN` char(20) CHARACTER SET latin1 NOT NULL,

`借书日期` date NOT NULL,

`实际还书日期` date NOT NULL,

PRIMARY KEY (`读者学号`,`ISBN`),

KEY `ISBN` (`ISBN`),

CONSTRAINT `图书归还表\_ibfk\_1` FOREIGN KEY (`ISBN`) REFERENCES `图书信息表` (`ISBN`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（5）管理员-书籍表

CREATE TABLE `管理员-书籍表`

(

`工作号` char(12) CHARACTER SET latin1 NOT NULL,

`ISBN` char(20) CHARACTER SET latin1 NOT NULL,

`添加时间` date DEFAULT NULL,

`是否在馆` char(2) NOT NULL,

PRIMARY KEY (`工作号`,`ISBN`),

KEY `ISBN` (`ISBN`),

CONSTRAINT `管理员-书籍表\_ibfk\_1` FOREIGN KEY (`ISBN`) REFERENCES `图书信息表` (`ISBN`),

CONSTRAINT `管理员-书籍表\_ibfk\_2` FOREIGN KEY (`工作号`) REFERENCES `管理员信息表` (`管理员工作号`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

（6）管理员-读者表

CREATE TABLE `管理员-读者表`

(

`工作号` char(12) CHARACTER SET latin1 NOT NULL,

`读者学号` char(20) CHARACTER SET latin1 NOT NULL,

`借还确认` char(4) NOT NULL,

`违章状况` char(6) DEFAULT NULL,

`借书累计` int(2) DEFAULT NULL,

PRIMARY KEY (`工作号`,`读者学号`),

KEY `读者学号` (`读者学号`),

CONSTRAINT `管理员-读者表\_ibfk\_1` FOREIGN KEY (`工作号`) REFERENCES `管理员信息表` (`管理员工作号`),

CONSTRAINT `管理员-读者表\_ibfk\_2` FOREIGN KEY (`读者学号`) REFERENCES `读者信息表` (`读者学号`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8;

5.2合法用户名字、权限、角色

（1）创建学生账户,账户名：学生账号（例），并且授予权限：

sql：

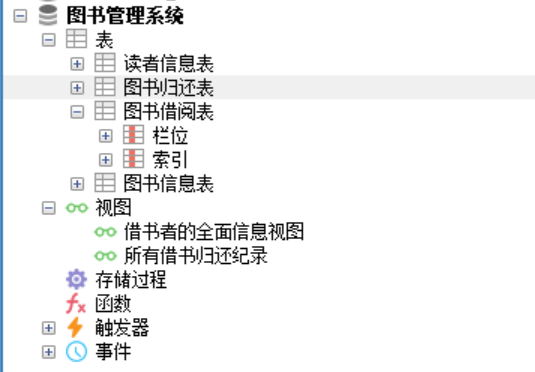
create user '学生账号'@'localhost';

grant insert,select on 读者信息表 to '学生账号'@'localhost';

grant select on 图书信息表 to '学生账号'@'localhost';

grant insert,select,update on 图书借阅表 to '学生账号'@'localhost';

grant insert,select,update on 图书归还表 to '学生账号'@'localhost';



（2）创建图书管理员账号，账户名：图书管理员账号（例）, 并且授予权限：

sql：

create user '管理员账户'@'localhost' identified by '123';

grant insert,select,update on 读者信息表 to '图书管理员账号'@'localhost';

grant insert,select,update on 图书信息表 to '图书管理员账号'@'localhost';

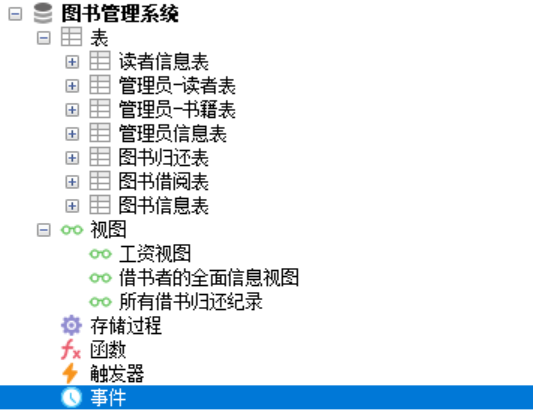
grant insert,select,update on 图书借阅表 to '图书管理员账号'@'localhost';

grant insert,select,update on 图书信息表 to '图书管理员账号'@'localhost';

grant select on 管理员信息表 to '图书管理员账号'@'localhost';

grant insert,select,update on 管理员-读者表 to '图书管理员账号'@'localhost';

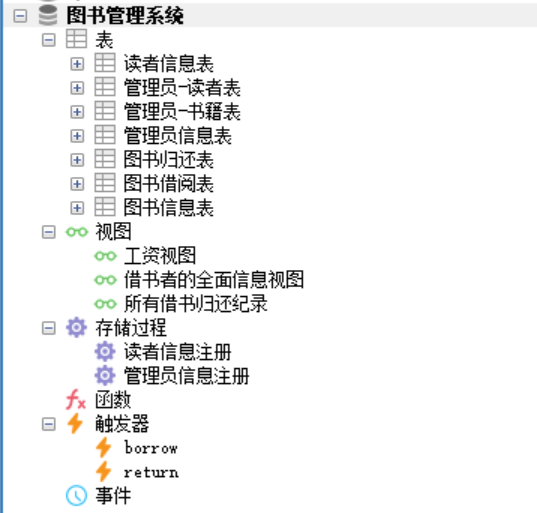
grant insert,select,update on 管理员-书籍表 to '图书管理员账号'@'localhost';



（3）创建数据库管理员账号，账户名：数据库管理员账号, 并且授予全部权限：

create user '管理员账户'@'localhost' identified by '123';

grant all on library\_management to '管理员账户'@'localhost';



5.3视图

在管理员信息表上建立工资视图

CREATE TABLE `工资视图`

(

`管理员姓名` char(6) ,

`管理员性别` char(2) ,

`工资` int(5)

)

所有借书归还纪录视图

CREATE TABLE `所有借书归还纪录`

(

`读者学号` char(20) ,

`读者姓名` char(6) ,

`ISBN` char(20) ,

`书名` char(20) ,

`借书日期` date ,

`实际还书日期` date

)

借书者的全面信息视图

CREATE ALGORITHM=UNDEFINED DEFINER=`数据库管理员账号`@`localhost` SQL SECURITY DEFINER VIEW `借书者的全面信息视图` AS

(

select `读者信息表`.`读者学号` AS `读者学号`,

`图书信息表`.`ISBN` AS `ISBN`,

`图书信息表`.`书名` AS `书名`,

`图书借阅表`.`借书日期` AS `借书日期`,

`图书借阅表`.`预期还书日期` AS `预期还书日期`

from ((`读者信息表` join `图书信息表`) join `图书借阅表`)

where ((`读者信息表`.`读者学号` = `图书借阅表`.`读者学号`)

and (`图书信息表`.`ISBN` = `图书借阅表`.`ISBN`))

)

5.4触发器

（1）设计触发器borrow, 借书成功后，对应的书籍的在架数量减1

CREATE

TRIGGER `borrow` AFTER INSERT ON `图书借阅表`

FOR EACH ROW BEGIN

UPDATE `图书信息表` SET `在架数量`=`在架数量`-1

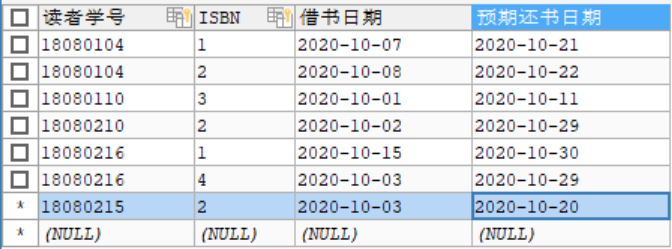
WHERE `ISBN`=new.`ISBN`;

END;

借书前图书信息：



进行操作借书：



借书后图书信息：



（2）设计触发器return, 还书成功后，对应的书籍的在架数量加1

CREATE

TRIGGER `return` AFTER INSERT ON `图书归还表`

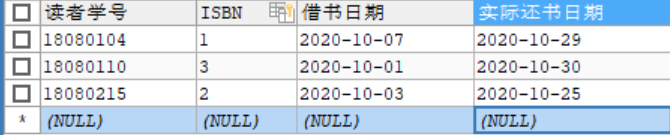
FOR EACH ROW BEGIN

UPDATE `图书信息表` SET `在架数量`=`在架数量`+1

WHERE `ISBN`=new.`ISBN`;

END;

让该同学马上进行还书操作：



还书后图书信息：



5.5索引

在图书信息表上建立唯一索引 书名

Create unique index 书名ON 图书信息表;

在图书管理员表上建立唯一索引 姓名

Create unique index 姓名ON图书管理员表;

在读者信息表上建立唯一索引 读者姓名

Create unique index 读者姓名ON读者信息表;

6数据库的实施

6.1表的创建与数据的载入

1．读者信息表

insert into `读者信息表`(`读者学号`,`读者姓名`,`读者性别`,`所在系部`,`联系电话`) values

('18080104','朱粤岫','女','数字媒体','19843210409'),

('18080110','应润凯','男','计算机科学','12879821721'),

('18080210','王纪晨','男','信息学部','18324432525'),

('18080215','孙漪威','男','信息学部','13821092849'),

('18080216','齐子逸','男','信息学部','18390290121'); 

2．管理员信息表

insert into `管理员信息表`(`工作号`,`姓名`,`性别`,`电话`,`工资`)

values

('1','孟宁宁','女',1851032566,4000),

('2','胡萧文','男',1214349093,5000),

('3','葛嘉文','男',1215632414,4500),

('4','余生','女',1947983702,5000);



3.图书信息表

insert into `图书信息表`(`ISBN`,`书名`,`作者`,`出版社`,`出版日期`,`书籍分类`,`在架数量`)

values

('1','论语','孔子','清华大学出版社','2011-10-01','国语类',4),

('2','三体','刘慈欣','中国科技出版社','2013-08-01','科技类',2),

('3','高等数学-下','杨晓霞','同济大学出版社','2020-10-03','教学类',6),

('4','数据结构','陈洪丽','中国商务出版社','2019-11-04','教学类',5),

('5','狂人日记','鲁迅','北京大学出版社','2000-01-25','国语类',6);



4．图书借阅表

insert into `图书借阅表`(`读者学号`,`ISBN`,`借书日期`,`预期还书日期`) values

('18080104','1','2020-10-07','2020-10-21'),

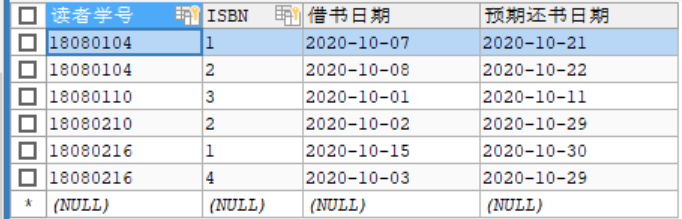
('18080104','2','2020-10-08','2020-10-22'),

('18080110','3','2020-10-01','2020-10-11'),

('18080210','2','2020-10-02','2020-10-29'),

('18080216','1','2020-10-15','2020-10-30'),

('18080216','4','2020-10-03','2020-10-29');

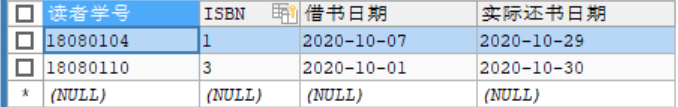


5. 图书归还表

insert into `图书归还表`(`读者学号`,`ISBN`,`借书日期`,`实际还书日期`) values

('18080104','1','2020-10-07','2020-10-29'),

('18080110','3','2020-10-01','2020-10-30');



6. 管理员-书籍表

insert into `管理员-书籍表`(`工作号`,`ISBN`,`添加时间`,`是否在馆`)

values

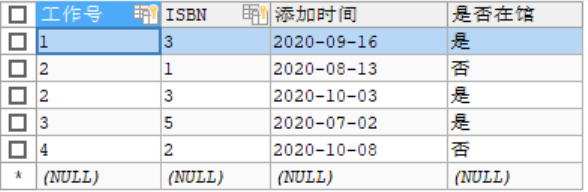
('1','3','2020-09-16','是'),

('2','1','2020-08-13','否'),

('2','3','2020-10-03','是'),

('3','5','2020-07-02','是'),

('4','2','2020-10-08','否');



7． 管理员-读者表

insert into `管理员-读者表`(`工作号`,`读者学号`,`借还确认`,`违章状况`,`借书累计`) values

('2','18080110','已换','暂无',4),

('3','18080104','已换','暂无',6),

('3','18080110','未还','暂无',4),

('3','18080215','已换','违章1次',10),

('4','18080210','未还',NULL,NULL);



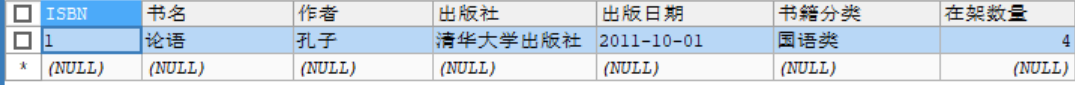
6.2**数据库的操作与查询**





**图6.2.1. 查找女读者的全部信息**





**图6.2.2 查询清华大学出版社的图书及作者**



**图6.2.3. 管理员工资视图**



**图6.2.4.借书者的全面信息视图**



**图6.2.5.所有借书归还信息视图**

**总结**

在本次课程设计中，我们小组合力完成了一个图书管理系统。一开始我们其实设计的相对简单了，概念模型里并没有考虑管理员，只保留了学生—<借阅>—书籍。经过小组同学的讨论与互相启发，我们在已有的基础上进行不断修改和迭代。从需求分析、数据库设计到上机编程、调试、应用，资料查询也从漫无目的变得更加具有针对性，也更加精确有方向。在实验的过程中遇到的一些问题可以通过报错信息的查询进行学习和解决。总的来说，这是一次具有挑战性的课程设计，合作中的磨合与扶持也在漫长艰苦的旅程中带来十足的惊喜感，大家都受益良多。缺点就是因时间原因，有一些表间的严谨逻辑关系未能用触发器表达出来，图书馆图形界面也只是用控制台程序表达，但是并不影响我们在大作业上的收获。