

**NANCHANG UNIVERSITY**

**数据库期中大作业**



**题 目:** 图书馆借书管理系统

**学 院：** 软件学院

**专 业：** 信息安全

**班 级：** 信息安全1903

**学生姓名：** 丁俊

**学生学号：** 8003119100

**起讫日期：** 2020.11.26 — 2020. 12.10

**任课教师：** 赵志宾

**完成时间：** 2020.11.29

**填表日期：** 2020 年 11 月 29 日

目录

[题目内容 3](#_Toc58445411)

[数据库的创建 4](#_Toc58445412)

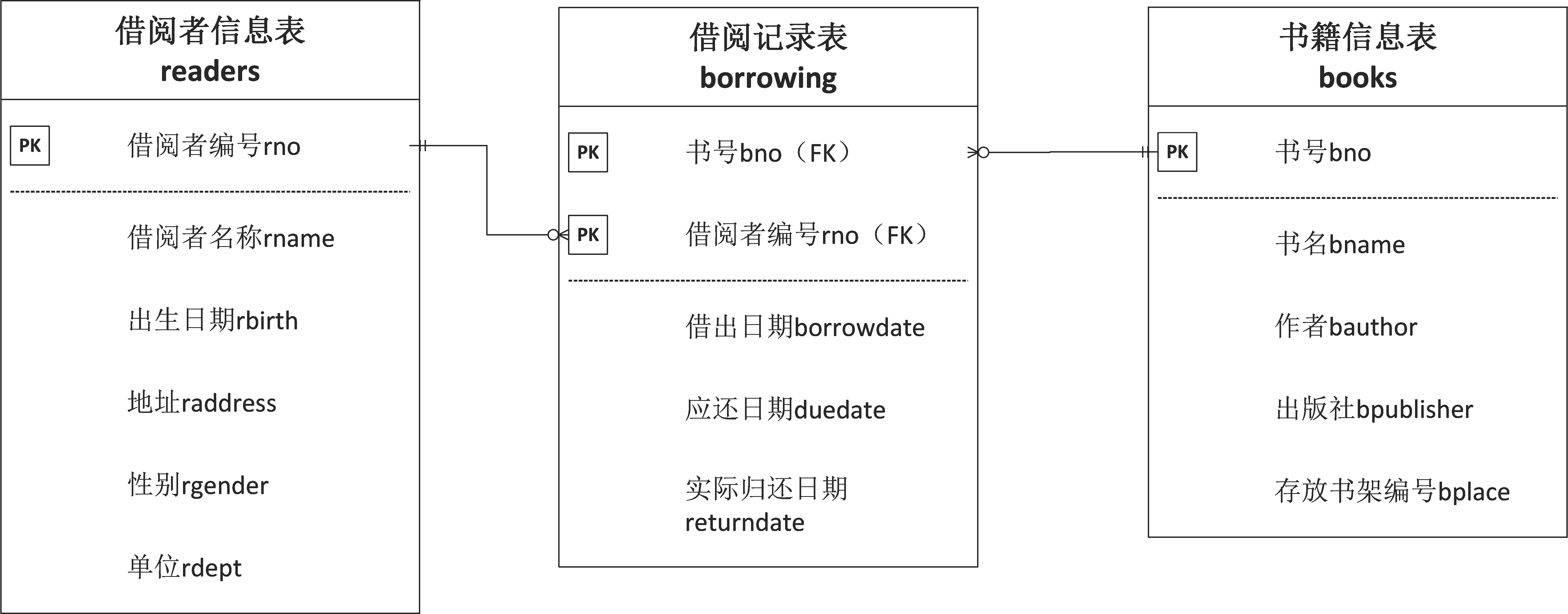
[表的创建 5](#_Toc58445413)

[解答代码 6](#_Toc58445414)

[总结 13](#_Toc58445415)

# 题目内容

* 图书馆数据库对每个借阅者存有读者记录，包括借阅者号、姓名、地址、性别、年龄、单位。
* 对每本书存有书号、书名、作者、出版社、库存位置编号。
* 每个借阅者允许借阅多本书籍，对每本被借出的书存有书号、借阅者号、借出日期、应还日期和实际归还日期。



1. **基于SQL SERVER软件，使用T-SQL语言在D盘根目录下创建一个名为liberary\_db的数据库，包含主文件和log文件，文件大小都为5MB，请写出相关语句。**
2. **根据表单关系图，使用SQL语言在liberary\_db数据库中建立表单，请写出相关语句，要求设置表单主键和外键约束。**
3. **根据liberary\_db数据库中数据表的信息和查询要求，写出对应的SQL语句。**
   1. 指向liberary\_db数据库。
   2. 查看图书馆全部书籍的信息。
   3. 查看所有借阅过“数据库原理”相关书籍的同学，并按照学号升序和姓名降序排列。
   4. 查看所有借阅书籍数量超过10本的同学的信息。
   5. 查看借书最多的五位同学的读者号、姓名、所借书籍的总数。
   6. 查看当天归还的书籍超期的同学的信息。
   7. 查看被借阅最多的10本书籍的信息。
   8. 查询借阅过全部书籍的同学。
   9. 查询借阅了“数据库原理”相关书籍，没有借阅“面向对象编程”相关书籍的读者的编号、姓名、单位。
   10. 查看超期书籍最多的同学的信息。

# 数据库的创建

**一、基于SQL SERVER软件，使用T-SQL语言在D盘根目录下创建一个名为liberary\_db的数据库，包含主文件和log文件，文件大小都为5MB，请写出相关语句。**

CREATE DATABASE liberary\_db

ON

(NAME = liberary\_Data,--主文件名字

FILENAME = 'D:\liberary\_Data.mdf',--存储路径

SIZE = 5 MB,--文件大小

MAXSIZE=100MB,

FILEGROWTH=15%

)

LOG ON(

NAME = liberary\_Log,--日志文件名字

FILENAME = 'D:\liberary\_Log.ldf',--存储路径

size = 5 MB,

MAXSIZE=100MB,

FILEGROWTH=15%

)

GO

# 

# 表的创建

**二、根据表单关系图，使用SQL语言在liberary\_db数据库中建立表单，请写出相关语句，要求设置表单主键和外键约束。**

--借阅者信息表

CREATE TABLE Readers

(Rno char(9) NOT NULL PRIMARY KEY,

Rname char(10) NOT NULL,

Rbirth date,

Raddress char(20) ,

Rgender char(2) NOT NULL,

Rdept char(20) NOT NULL

)

--书籍信息表

CREATE TABLE Books

(Bno char(10) NOT NULL PRIMARY KEY,--主键Bno

Bname char(20) Not NULL,

Bauthor char(15),

Bpublisher char(20),

Bplace char(10) Not NULL

)

--借阅记录表

CREATE TABLE Borrowing

(Bno char(10) NOT NULL,

Rno char(9) NOT NULL,--不能为空

Borrowdate date,

duedate date,

returndate date,--date数据类型

PRIMARY KEY(Bno,Rno),--主键由Rno和Bno组成

FOREIGN KEY(Rno) REFERENCES Readers(Rno), --外键约束，被参照表是Readers,列是Rno

FOREIGN KEY(Bno) REFERENCES Books(Bno)--外键约束，参照书籍信息表中的Bno.1

)

--原先创建表单时忘记设置唯一值约束条件，-用语句设置-

alter table Books ADD UNIQUE(Bno)

alter table Readers ADD UNique(Rno)

# 解答代码

**三、根据liberary\_db数据库中数据表的信息和查询要求，写出对应的SQL语句。**

1、指向liberary\_db数据库。

USE liberary\_db

GO

2、查看图书馆全部书籍的信息。

SELECT \*

FROM Books

3、查看所有借阅过“数据库原理”相关书籍的同学，并按照学号升序和姓名降序排列。

SELECT Readers.Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers,Borrowing,Books

where Readers.Rno=Borrowing.Rno

and Borrowing.Bno=Books.Bno

and Books.Bname LIKE '%数据库原理%'

ORDER BY Rname DESC --按姓名降序排列

--分析:把三个表通过各自的外键连接起来,再通过Bname属性找出与“数据库原理”相关的行，即筛选出了借阅此书的同学，再用ORDER BY DESC or ASC分别按照学号升序和姓名降序排列。

SELECT Readers.Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers JOIN Borrowing s1 ON(Readers.Rno=s1.Rno),

Books JOIN Borrowing s2 ON(Books.Bno=s2.Bno)

where Books.Bname LIKE '%数据库原理%' and s1.Rno=s2.Rno

and s1.Bno=s2.Bno

ORDER BY Readers.Rno --按学号升序(默认)排列

1. 查看所有借阅书籍数量超过10本的同学的信息。

**方案一:**

SELECT Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers

where Rno IN

(SELECT Rno

from Borrowing

GROUP BY Rno --每个不同的借阅者(编号)为一组

HAVING COUNT(\*)>10 --每一组里面超过本书即为所借书籍超过本

)

**方案二:**

SELECT Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers

where EXISTS

(SELECT Rno

from Borrowing

GROUP BY Rno --每个不同的借阅者(编号)为一组

HAVING COUNT(\*)>10 and Readers.Rno=Borrowing.Rno --每一组里面超过本书即为所借书籍超过本

)

**方案三:**

SELECT Readers. Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers,Borrowing

where Readers.Rno = Borrowing.Rno --连接表

group by Readers.Rno,Rname,Rbirth,Raddress,Rgender,Rdept

HAVING COUNT(\*) > 10

选第一个比较合理，第一个查询是先查询内层表，选出借书数量超过10本的借阅者编号Rno,在从Readers表中查询这些Rno一致的借阅者，并选择；方案二和方案一差不多；方案三要先建立连接，再把借阅者信息放进一个组里，因为这些信息是关于Rno不变的，分组后每组有几行记录就表示借了多少本书。

5、查看借书最多的五位同学的读者号、姓名、所借书籍的总数。

-查看借书最多的五位同学的读者号,姓名,所借书籍的总数

SELECT top 5 Readers.Rno 编号,Rname 姓名,COUNT(\*) as 所借书籍总数

from Readers,Borrowing

where Readers.Rno = Borrowing.Rno--连接两个表

GROUP BY Readers.Rno,Rname--把连接后的表中的姓名和读者编号放进分组里

ORDER BY 所借书籍总数 DESC--表里的行数即为所借书籍的数量

分析:因为没有涉及到Books表的数据，只需要把前面两个表连接，按照读者编号分组(即同一编号里面的书籍就是一个人借的所有书籍),进而得出书籍的总数，再用ORDER BY 对COUNT(每组的数量)进行排序。

6、查看当天归还的书籍超期的同学的信息。

--查看当天归还的书籍超期的同学信息

**方案一:**

SELECT Readers.Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers JOIN Borrowing ON(Readers.Rno=Borrowing.Rno)--连接两个表

where convert(varchar(8),getdate(),112)=convert(varchar(8),returndate,112)

and convert(varchar(8),returndate,112) > convert(varchar(8),duedate,112)

--convert函数转换为相应四位日期的格式

**方案二:**

select distinct Readers.Rno 编号,Rname 姓名,Rbirth 出生日期,Raddress 地址,Rgender 性别,Rdept 单位

from Readers JOIN Borrowing ON(Readers.Rno=Borrowing.Rno)

where Readers.Rno in --判断Rno是否在今天归还书籍的人群当中

(select Readers.Rno

from Readers,Borrowing

where Readers.Rno=Borrowing.Rno and DATEDIFF(DAY,GETDATE(),returndate)=0

and DATEDIFF(DAY,returndate,duedate)<0

)

对比:方案一是直接把Readers和Borrowing两个表连接起来，用“当日归还”和“超期归还”两个条件并列进行判断；而方案二是先筛选出“当天归还的读者人群”，再在外层再次连接两表根据“超期归还”这一条件找出当天归还超期的同学。对比来看方案一更简洁且最为简单，方案二看似是做了多余的步骤，只是步骤和思想不同。

7、查看被借阅最多的10本书籍的信息。

-查看被借阅最多的10本书籍的信息

--因为一名借阅者不会借同一本书，所以书号Bno+读者号Rno不会重复

SELECT top 10 Books.Bno 书号,Bname 书名,Bauthor 作者,Bpublisher 出版社,Bplace 书架

from Books JOIN Borrowing ON(Books.Bno=Borrowing.Bno)--连接两个表

GROUP BY Books.Bno,Bname,Bauthor,Bpublisher,Bplace

ORDER BY COUNT(\*) DESC

分析：--因为两个表连接了，所以Books.Bno也可换成Borrowing.Bno

降序排列,所以前10个记录即为被借阅最多的10本书

8、查询借阅过全部书籍的同学。

--查看借阅过全部书籍的同学--没有一本书是他不借的

**方案一:**

SELECT \*

from readers

where NOT EXISTS

(SELECT \*

from Books

WHERE NOT EXISTS

(SELECT \*

from Borrowing

WHERE Bno=Books.Bno and Rno=Readers.Rno

)

)

代码解释:

第三层查询判断每一个书号是否可以输出，当没有一个书号可以输出时，这个元组才可以输出

,至此表示这个借阅者借阅了所有书籍

外层是选择一行读者信息，然后进入第二层选择一行借阅记录，进入第三次查看是否三个

表可以连接(表示读者借了这本书),如果每个书号都能连接成功，则第三次条件值为假，当所以书号使得其都为假时,这时第二层无书号无元组可输出,即此时第二层条件为空,最外层为真,这时可以输出。

**方案二:**

select Rname 名称

from Readers,Books,Borrowing

where Readers.Rno=Borrowing.Rno

and Books.Bno=Borrowing.Bno

group by Rname

HAVING COUNT(Books.Bno)=

(select COUNT(\*) from Books)

代码解释：先把三个表构建连接，再根据Bno进行分组，统计各组的数量并和Books的总数量对比，如果相等即借阅了所有的书籍。

对比:我觉得方案二的方法更好，也更易理解，直接拿某个读者的借阅表和所有书籍表的数量(即书的总数)比较看是否相等，相等即借阅了所有书籍，反之则没有；而方案一每次选取Readers一组后都要进行两次NOT EXISTS空与否的判断，不仅不易理解，而且代码复杂度也较高，综合考虑还是选择方案二。

9、查询借阅了“数据库原理”相关书籍，没有借阅“面向对象编程”相关书籍的读者的编号、姓名、单位。

**方案一:**

SELECT Readers.Rno 编号,Rname 姓名,Rdept 单位

from Readers

where EXISTS

(select \* from

Borrowing,Books

where Readers.Rno=Borrowing.Rno and Borrowing.Bno=Books.Bno

and Bname LIKE '%数据库原理%'

)

AND NOT EXISTS

(select \* from

Borrowing,Books

where Readers.Rno=Borrowing.Rno and Borrowing.Bno=Books.Bno

and Bname LIKE '%面向对象编程%'

)

方案一是从Readers逐次选出一行，进入EXISTS子句判断是否符合条件符合即为真值，再结合第二个条件NOT EXISTS子句若选中的Readers某行不符合条件则为真就可以输入了，即条件借阅了“数据库原理”且没有借阅“面向对象编程”的读者记录被输出。

**方案二:**

SELECT Readers.Rno 编号,Rname 姓名,Rdept 单位

from Readers,Borrowing,Books

where Readers.Rno=Borrowing.Rno and Borrowing.Bno=Books.Bno

and Bname LIKE '%数据库原理%' and

Readers.Rno not in(

SELECT Readers.Rno

from Readers,Borrowing,Books

where Readers.Rno=Borrowing.Rno and Borrowing.Bno=Books.Bno

and Bname LIKE '%面向对象编程%'

)

方案二先判断Readers中的一行记录借阅了“数据库原理”的读者，再判断Rno是否在未选修“面向对象编程”的读者集合里面。

对比：方案二较好，且较为简单地理解，更符合常理；方案一单从EXISTS和NOT EXISTS来思考不是那么容易看出，而且每次都要进行一次EXISTS内的语句判断。但是方案二直接把没有借阅“面向对象编程”的读者集合起来，只需要判断Rno读者编号是否在里面就可以了。

10、查看超期书籍最多的同学的信息。

**方案一:**

SELECT \*

from Readers

where Rno =

(SELECT top 1 Rno

from Borrowing

where convert(varchar(8),returndate,112) > convert(varchar(8),duedate,112)--超期归还

--把超期归还的行数留下，去除不符合超期条件的行数

--因为where在group by 之前进行筛选

GROUP BY(Rno)--同一编号放进一组中

ORDER BY COUNT(\*) DESC --按降序排列,那么第一行即为超期书籍数最多的那位

)

**方案二:**

select \*from readers where rno =

(select rno from

borrowing

where DATEDIFF(day,returndate,duedate)<0—超期判断

group by rno

having COUNT(bno)

=(select max(s.a) –选出最大的超期数量

from (select COUNT(bno) as a from borrowing

where

DATEDIFF(DAY,returndate,duedate)<0 group by rno)s ));

对比：方案一是先把超期的记录筛选出来，然后根据Rno分组求出每个读者超期书籍的数量，

再降序排列第一个就是超期数量最多的读者；而方案二是先在内层找出超期数量最多的那条

记录中的数量MAX,再根据分组统计数量判断COUNT是否相等，把相等的Rno记录，最后在外层筛选Rno；我觉得方案一最好，它直接把所有读者超期书籍数量信息按顺序排列下来，一下子就能理解和找出。

# 总结

这次大作业是在第一天晚上写完的，因为开始并没有测试数据，所以有些代码运行起来还是有问题的，后来几次修改并且在添加数据后，及时更正了错误，在原有基础上也添加了方法。所有案例都是在数据测试下完成的，尽管还有些许瑕疵。

通过这次小作业，我体会到了数据库的强大和魅力，虽然只是一些查询语句，却能变幻万千，产生很多不同的结果和过程。在一些关键词EXISTS、NOT EXISTS、SELECT、GROUP BY、

ORDER BY、COUNT、AVG等的变换和组合下，我们可以随意获得我们想要的数据结构和信息排列组合，一行一行记录不断的层叠最终形成了具有强大检索能力的数据库天地。