****

数据库课程设计

**姓 名：胡澳治**

**学 号：201900800510**

**题 目：学校图书借阅管理系统**

**专 业：计算机科学与技术**

**年 级：19级**

**2021年 5 月 3 日**

# 选题说明

图书借阅系统是图书馆借书遇到的系统，非常典型，是学习过程实践的好例子。

# 2．需求介绍

**需求**：

1.实现图书信息、类别、出版社等信息的管理；

2.实现读者信息、借阅信息的管理；

3.实现图书的借阅、续借，归还管理；

4.实现超期罚款管理；

5.创建触发器，分别实现借书和还书时自动更新图书信息的在册数量；

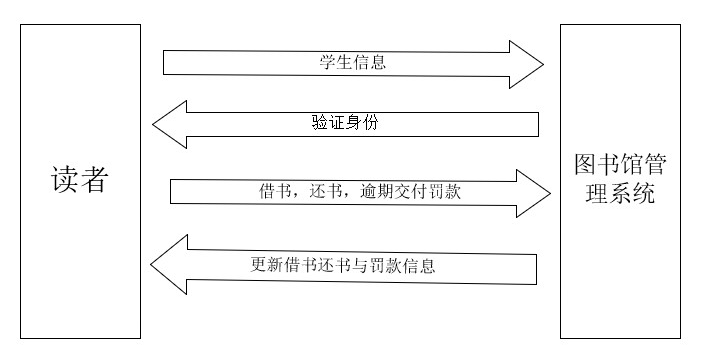
6.创建视图查询各种类型图书的书号、书名、总数和在册数；

7.创建存储过程查询指定读者借阅图书的情况；

8.建立数据库相关表之间的参照完整性约束。

即能查询全部图书和当前在册图书，能确定某本书是否借出，能查询借书的学生的信息，借书本书以及对应书籍的信息，能查询管理员信息，能处理借书，还书，续借，罚款操作等一般图书馆管理需要的基本功能

环境图



# 2．需求介绍

**需求**：

1.实现图书信息、类别、出版社等信息的管理；

2.实现读者信息、借阅信息的管理；

3.实现图书的借阅、续借，归还管理；

4.实现超期罚款管理；

5.创建触发器，分别实现借书和还书时自动更新图书信息的在册数量；

6.创建视图查询各种类型图书的书号、书名、总数和在册数；

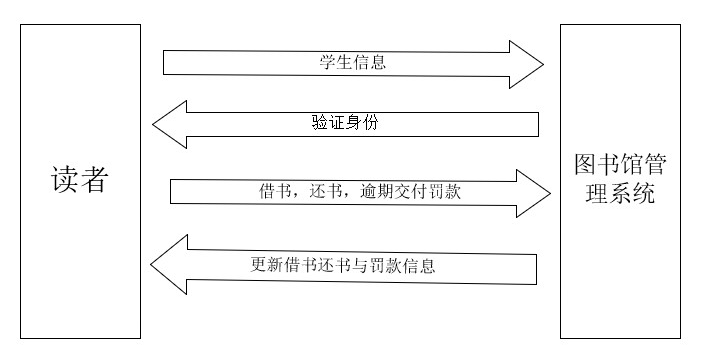
7.创建存储过程查询指定读者借阅图书的情况；

8.建立数据库相关表之间的参照完整性约束。

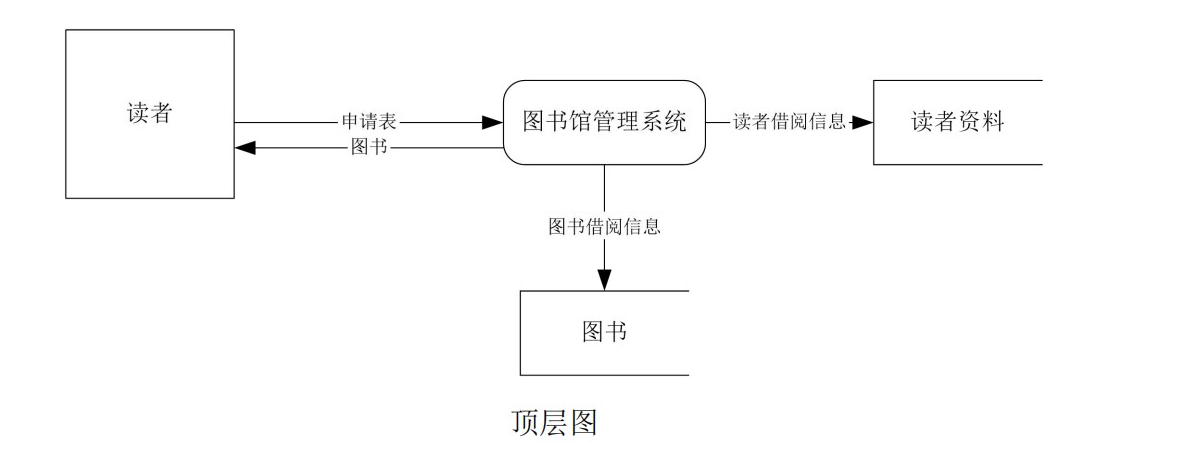
即能查询全部图书和当前在册图书，能确定某本书是否借出，能查询借书的学生的信息，借书本书以及对应书籍的信息，能查询管理员信息，能处理借书，还书，续借，罚款操作等一般图书馆管理需要的基本功能

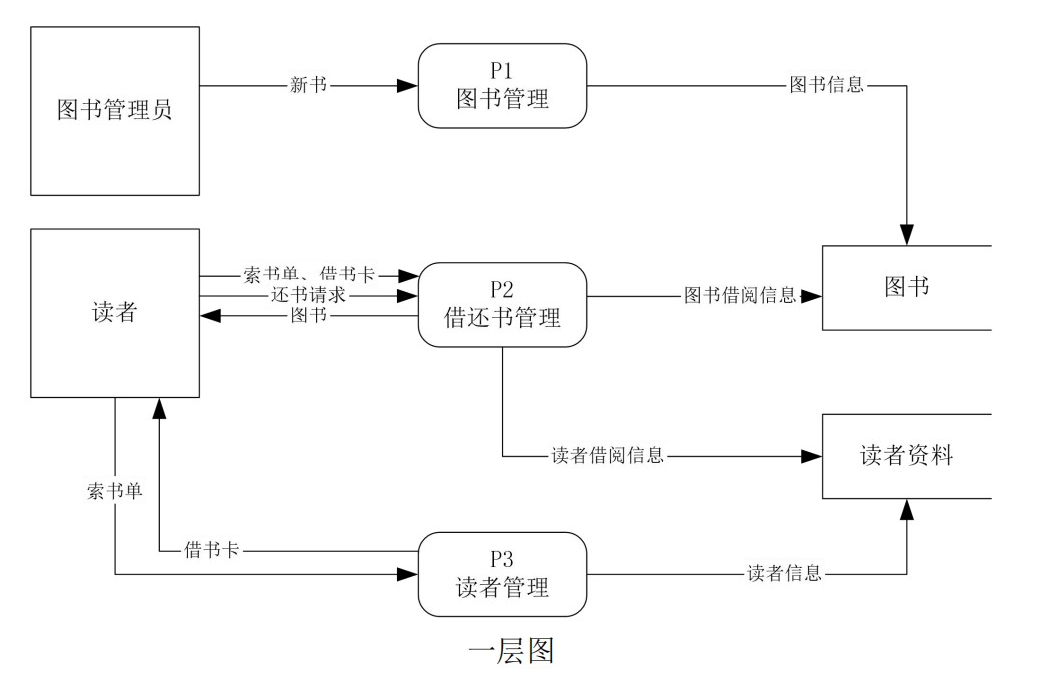
# .3 数据流图与数据字典

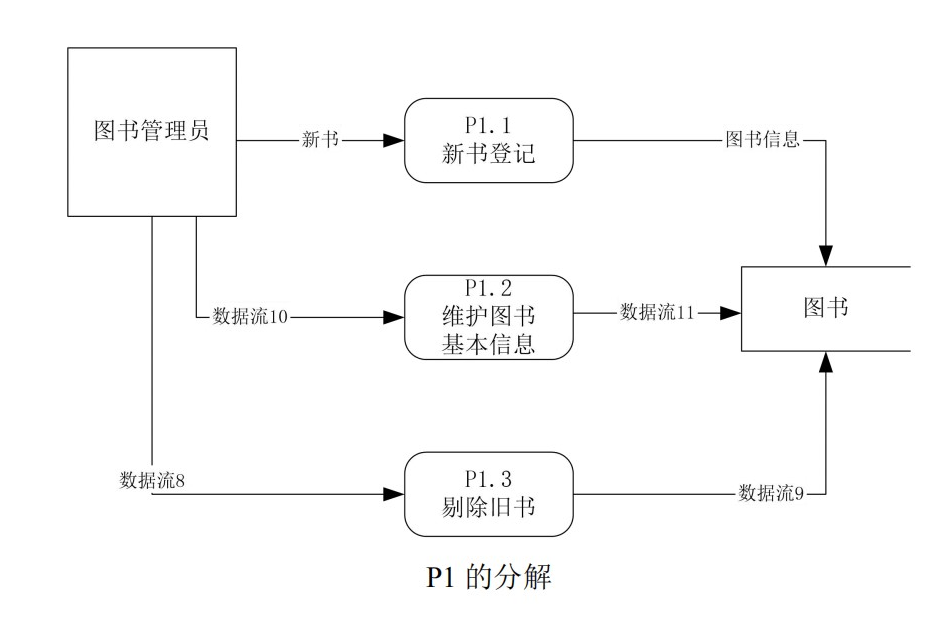
**环境图：**

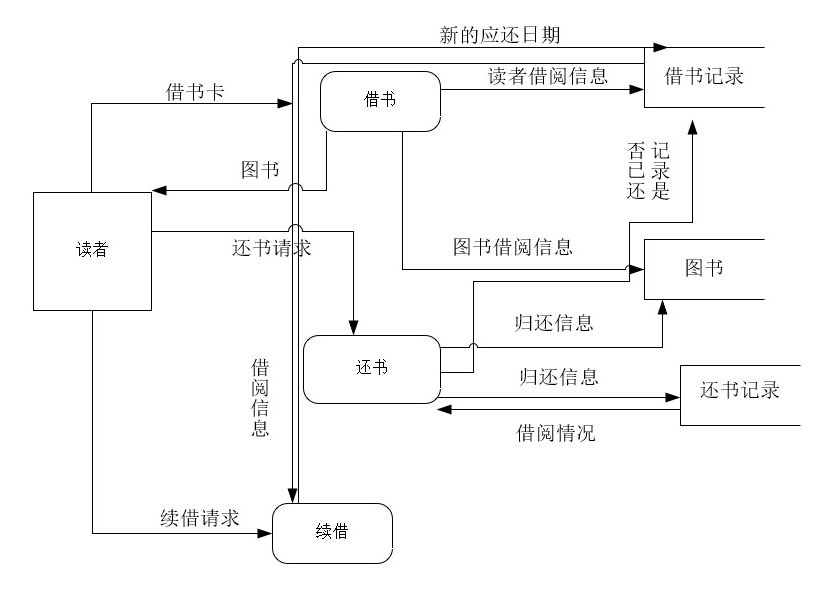


**数据流图：**

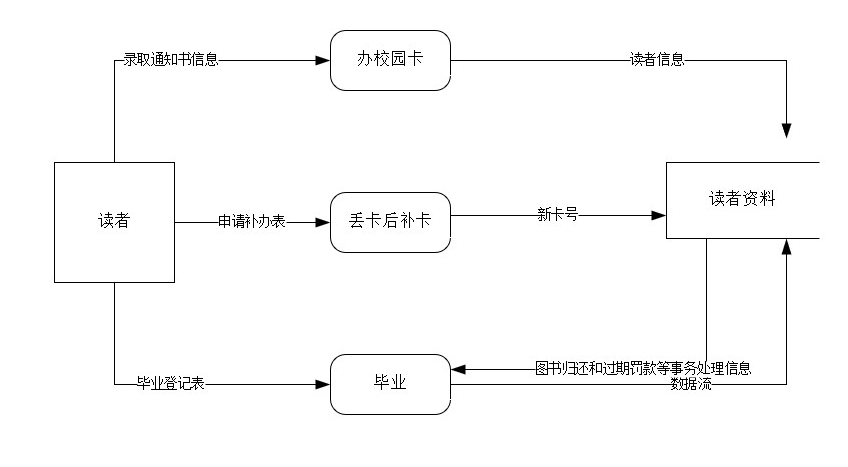








**P2的分解**



**P3的分解**

**数据项**：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据项 | 别名 | 类型 | 长度 | 取值范围 | 取值含义 | 含义说明 |
| 学号 | 学生编号 | Char | 12 | 2017000000  到2020000000 | 学生级数+院系编号+专业编号+编号 | 唯一标识一名学生 |
| 书号 | 书本编号 | Char | 10 | 0到书本总数 | 书的编号 | 唯一标识一本书 |

**数据结构**：

|  |  |  |
| --- | --- | --- |
| 数据结构 | 含义说明 | 组成 |
| 学生 | 借阅书籍的的主体数据结构，定义了一个学生的有关信息 | 学号，姓名，性别，院系，年级，专业 |
| 书籍 | 被借阅的主体数据结构，定义了一本书的有关信息 | 书号，书名，类型，出版社，作者，价格 |

**数据流**：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据流 | 含义说明 | 数据流来源 | 数据流去向 | 组成 |
| 借阅图书信息流 | 学生从图书馆借出书籍 | 学生 | 图书借还管理 | 学生及所借书籍信息组成的借书关系 |
| 还书信息流 | 学生归还图书 | 学生 | 图书借还管理 | 学生及所还图书信息组成的还书关系 |

**数据存储**：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据存储 | 含义说明 | 流入的数据流 | 流出的数据流 | 组成 | 存取方式 | 存取量 |
| 学生信息 | 学生读者的基本信息 | 新生的信息 | 查询的学生的信息 | 学号，姓名，性别，院系，年级，专业 | 顺序存取 | 每年新增招生数 |
| 图书馆在册图书 | 所有在册的图书的记录 | 图书馆新增图书 | 图书馆借出的图书 | 书号 | 随机存取 | 新增的图书量和借出的图书量的差 |
| 书籍信息 | 每本书的基本信息 | 新增图书的信息 | 查询的图书的信息 | 书号，书名，类型，出版社，作者，价格 | 顺序存取 | 新增的图书量 |
| 读者借书信息 | 读者借书记录 | 某读者借某本书 | 查询读者借书情况 | 读者学号，书本编号，借书日期，应还日期 | 顺序存取 | 读者借书记录数量 |

**处理过程**：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 处理过程 | 含义说明 | 输入数据流 | 输出数据流 | 处理 |
| 借书操作处理 | 根据对借书操作对在册图书，借书记录更新 | 借书的读者信息与书本信息等 | 查询借书记录信息 | 借书后在借书记录中添加借书记录，减少在册图书 |
| 还书操作处理 | 根据对还书操作对在册图书，借书记录更新 | 还书的读者信息与书本信息等 | 查询借书信息 | 删除借书记录，增加在册图书 |
| 新增图书处理 | 图书馆新增图书进行信息更新 | 新增的图书信息 | 可查询到该新图书 | 在书籍表中加入该书信息 |

**数据项**：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据项 | 别名 | 类型 | 长度 | 取值范围 | 取值含义 | 含义说明 |
| 学号 | 学生编号 | Char | 12 | 2017000000  到2020000000 | 学生级数+院系编号+专业编号+编号 | 唯一标识一名学生 |
| 书号 | 书本编号 | Char | 10 | 0到书本总数 | 书的编号 | 唯一标识一本书 |

**数据结构**：

|  |  |  |
| --- | --- | --- |
| 数据结构 | 含义说明 | 组成 |
| 学生 | 借阅书籍的的主体数据结构，定义了一个学生的有关信息 | 学号，姓名，性别，院系，年级，专业 |
| 书籍 | 被借阅的主体数据结构，定义了一本书的有关信息 | 书号，书名，类型，出版社，作者，价格 |

**数据流**：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据流 | 含义说明 | 数据流来源 | 数据流去向 | 组成 |
| 借阅图书信息流 | 学生从图书馆借出书籍 | 学生 | 图书借还管理 | 学生及所借书籍信息组成的借书关系 |
| 还书信息流 | 学生归还图书 | 学生 | 图书借还管理 | 学生及所还图书信息组成的还书关系 |

**数据存储**：

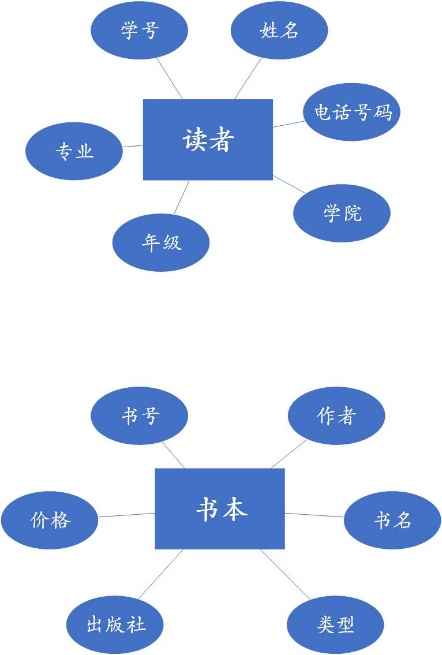
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据存储 | 含义说明 | 流入的数据流 | 流出的数据流 | 组成 | 存取方式 | 存取量 |
| 学生信息 | 学生读者的基本信息 | 新生的信息 | 查询的学生的信息 | 学号，姓名，性别，院系，年级，专业 | 顺序存取 | 每年新增招生数 |
| 图书馆在册图书 | 所有在册的图书的记录 | 图书馆新增图书 | 图书馆借出的图书 | 书号 | 随机存取 | 新增的图书量和借出的图书量的差 |
| 书籍信息 | 每本书的基本信息 | 新增图书的信息 | 查询的图书的信息 | 书号，书名，类型，出版社，作者，价格 | 顺序存取 | 新增的图书量 |
| 读者借书信息 | 读者借书记录 | 某读者借某本书 | 查询读者借书情况 | 读者学号，书本编号，借书日期，应还日期 | 顺序存取 | 读者借书记录数量 |

**处理过程**：

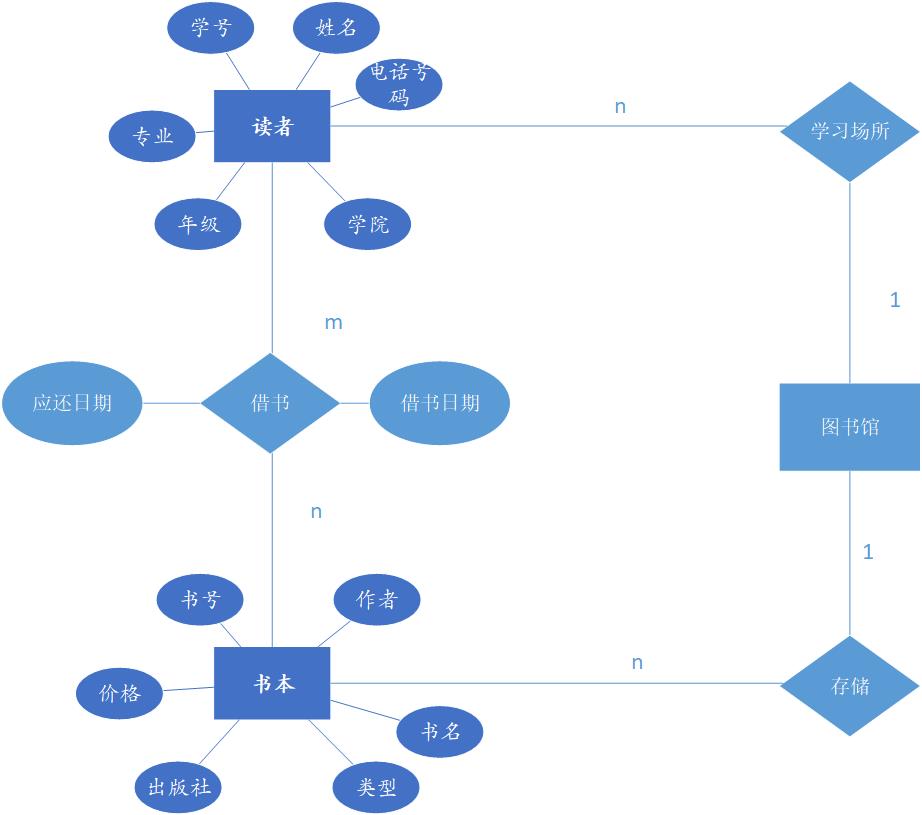
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 处理过程 | 含义说明 | 输入数据流 | 输出数据流 | 处理 |
| 借书操作处理 | 根据对借书操作对在册图书，借书记录更新 | 借书的读者信息与书本信息等 | 查询借书记录信息 | 借书后在借书记录中添加借书记录，减少在册图书 |
| 还书操作处理 | 根据对还书操作对在册图书，借书记录更新 | 还书的读者信息与书本信息等 | 查询借书信息 | 删除借书记录，增加在册图书 |

# 3．概念结构设计

## 3．1 局部E-R图



## 3．2 全局E-R图



# 4．逻辑结构设计

学生（学号，姓名，专业，学院，电话号码）；

图书馆管理员（管理员号，性别，姓名，电话号码）

书籍表（书号，书名，封面图片地址，作者，类型，出版社，价格）；

借书记录（学号，书号，借书日期，管理员号，应还日期，是否已经续借）；

还书记录（学号，书号，还书日期，管理员号）

罚款表（学号，书号，还书日期，逾期天数，罚款金额）

每一个非主属性既不部分依赖于码也不传递依赖于码，是第三范式；

# 5．物理结构设计

**书籍表**



**学生表**

****

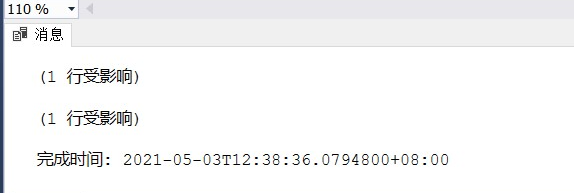
**图书管理员表**

****

**下面进行部分操作测试：**

**1进行借书操作：**

exec 借书 '201900800510','2021042201','21001'; --两行受影响

****

**借书记录表：（产生一条借书记录，并记录应还日期等信息）**

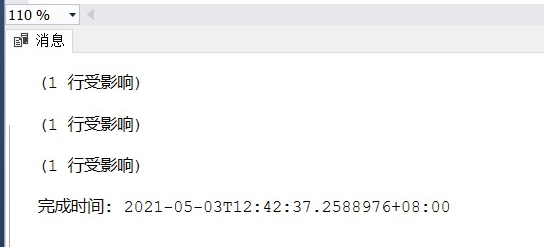
****

**书籍表：（对应的书已是借出状态）**

****

**2 进行还书操作：**

exec 还书 '2021042201','21001'; --三行受影响

****

**书籍表（被还书在册）**

****

**借书记录表（已归还）**

****

**还书记录表**

****

**代码清单**

--创建俩数据库

use master;

create database 图书馆管理数据库

ON PRIMARY

(

name = 'library',

filename = 'D:\201900800510\library\_data.mdf',

size = 5mb,

maxsize = 500mb,

filegrowth = 10%

)

LOG ON

(

name = 'library\_log',

filename = 'D:\201900800510\library\_log.ldf',

size = 3mb,

maxsize = unlimited,

filegrowth = 1mb

)

--创建表

use 图书馆管理数据库;

create table 学生表

(

学号 varchar(20) primary key,

姓名 varchar(10),

学院 varchar(20),

专业 varchar(20),

电话号码 char(14),

)

create unique index 学号索引

on 学生表(学号 asc)

create table 图书馆管理员表

(

管理员号 varchar(5) primary key,

性别 char(2),

姓名 varchar(20),

电话号码 char(14),

)

create unique index 管理员号索引

on 图书馆管理员表(管理员号 asc)

create table 书籍表

(

书号 varchar(10) primary key,

书名 varchar(20) not null,

封面图片地址 varchar(50),

作者 varchar(20) not null,

类型 varchar(20) not null,

出版社 varchar(20) not null,

价格 float(3),

在册或借出 char(4) default '在册' not null,

)

create unique index 书索引

on 书籍表(书号 asc,书名)

create table 借书记录表

(

学号 varchar(20) not null,

书号 varchar(10) not null,

管理员号 varchar(5),

借书日期 date not null,

应还日期 date,

已续借 char(2),

已归还 char(2) check(已归还 in ('是','否')),

primary key(学号,书号,借书日期)

)

create unique index 借书日期索引

on 借书记录表(借书日期)

create table 还书记录表

(

学号 varchar(20) not null,

书号 varchar(10) not null,

管理员号 varchar(5),

还书日期 date not null,

primary key(学号,书号,还书日期)

)

create unique index 还书日期索引

on 还书记录表(还书日期)

create table 罚款表

(

学号 varchar(20) not null,

书号 varchar(10) not null,

还书日期 date not null,

逾期天数 int,

罚款金额 float,

primary key(学号,书号,还书日期)

)

create unique index 罚款索引

on 罚款表(学号,罚款金额)

--外码约束

alter table 借书记录表

add constraint frgn\_key\_lend\_1

foreign key(学号) references 学生表(学号);

alter table 借书记录表

add constraint frgn\_key\_lend\_2

foreign key(书号) references 书籍表(书号);

alter table 借书记录表

add constraint frgn\_key\_lend\_3

foreign key(管理员号) references 图书馆管理员表(管理员号);

alter table 还书记录表

add constraint frgn\_key\_return\_1

foreign key(学号) references 学生表(学号);

alter table 还书记录表

add constraint frgn\_key\_return\_2

foreign key(书号) references 书籍表(书号);

alter table 还书记录表

add constraint frgn\_key\_return\_3

foreign key(管理员号) references 图书馆管理员表(管理员号);

alter table 罚款表

add constraint frgn\_key\_fine\_1

foreign key(学号) references 学生表(学号);

alter table 罚款表

add constraint frgn\_key\_fine\_2

foreign key(书号) references 书籍表(书号);

--自定义的约束

alter table 图书馆管理员表

add constraint zdy\_1 check(性别 in ('男','女'))

alter table 借书记录表

add constraint zdy\_2 check(已续借 in('是','否'))

alter table 借书记录表

add constraint zdy\_3 default getdate() for 借书日期 --借书日期是date型

alter table 借书记录表

add constraint zdy\_4 default dateadd(day, 30, getdate()) for 应还日期

alter table 借书记录表

add constraint zdy\_5 default '否' for 已续借

alter table 还书记录表

add constraint zdy\_6 default getdate() for 还书日期

alter table 学生表

add constraint zdy\_7 check(len(学号) >= 12) --学号大于等于12位

alter table 借书记录表

add constraint zdy\_8 default '否' for 已归还

-- 因为可能同一个学生可能借同一本书多次，必须记录是否归还

--create assertion asse\_1 --断言，每个人任何时刻不能同时借书超过60本

--check((select max(count(\*)) from 借书记录表 group by 学号) <= 60);

--网上说sql没有assertion关键字，应该通过触发器设置断言功能

--向书籍表，学生表，图书管理员表这三个表手动导入数据

--设置触发器

go

create trigger limited\_book --保证每个人任何时刻不能同时借书超过60本

on 借书记录表

for insert

as

begin

declare @book\_num int;

select top 1 @book\_num = count(\*)

from 借书记录表

group by 学号

order by count(\*) desc;

if(@book\_num > 60)

begin

print('该同学借书已达上限60本，无法再借，请先还部分书籍');

rollback;

end

end

go

create trigger return\_book --还书时如果逾期则将罚款自动记录到罚款表,

--同时将书籍表中所还书标记为在册

on 还书记录表

for insert

as

begin

declare @学号 varchar(20);

declare @书号 varchar(10);

declare @应还日期 date;

select @学号 = 学号,@书号 = 书号

from inserted;

update 书籍表 --更新书籍表

set 在册或借出 = '在册'

where 书号 = @书号;

select @应还日期 = 应还日期

from 借书记录表

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否';

declare @逾期天数 int;

set @逾期天数 = DATEDIFF(day, @应还日期, getdate())

if(@逾期天数 > 0)

begin

insert into 罚款表--逾期每天罚款0.15元

values(@学号,@书号,getdate(),@逾期天数,cast(@逾期天数\*0.15 as float))

end

update 借书记录表 --更新借书记录表

set 已归还 = '是'

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否';

end

go

--建立视图

create view 东野圭吾的书

as

select \*

from 书籍表

where 作者 = '东野圭吾'

go

create view 机械工业出版社的书

as

select distinct(书名)

from 书籍表

where 出版社 = '机械工业出版社'

go

create view 价格大于30元的书

as

select distinct(书名)

from 书籍表

where 价格 > 30

go

create view 库存小于10本的书

as

select \*

from (select 书名,count(\*)

from 书籍表

group by 书名) as lsb(书名,本数)

where lsb.本数 < 10;

go

create view 管理员彭于晏处理的借书记录

as

select \*

from 借书记录表

where 管理员号 = (select 管理员号 from 图书馆管理员表 where 姓名 = '彭于晏')

go

create view 小说类型的书 --小说类型的书的全部书号，书名，以及小说类型的总数和在册数

as

select count(\*) 总数,count(case 在册或借出

when '在册'

then 1 else 0

end)在册数

from 书籍表

where 类型 = '小说'

go

--建立存储过程

--查询指定读者的当前借阅图书的情况

create proc 读者借阅情况 @学号 varchar(20)

as

select 学生表.学号,学生表.姓名,借书日期,应还日期,已续借

from 借书记录表 join 学生表 on 借书记录表.学号 = 学生表.学号

where 借书记录表.学号 = @学号

and 已归还 = '否'

go

--查询指定读者的全部借阅历史

create proc 读者全部借阅历史 @学号 varchar(20)

as

select 学生表.学号,学生表.姓名,借书日期,应还日期,已续借

from 借书记录表 join 学生表 on 借书记录表.学号 = 学生表.学号

where 借书记录表.学号 = @学号

go

--借书

create proc 借书 @学号 varchar(20), @书号 varchar(10), @管理员号 varchar(5)

as

insert into 借书记录表

values(@学号,@书号,@管理员号,getdate(),DATEADD(day,30,getdate()),'否','否')

if(@@ROWCOUNT = 1) --没有因超过60本而被触发器rollback，则继续后面的步骤

begin

update 书籍表

set 在册或借出 = '借出'

where 书号 = @书号;

end

go

--续借

create proc 续借 @学号 varchar(20), @书号 varchar(10)

as

declare @已续借 char(2);

select @已续借 = 已续借

from 借书记录表

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否'

if(@已续借 = '是')

print('只能续借一次，续借失败！')

else

begin

declare @应还日期 date;

select @应还日期 = 应还日期

from 借书记录表

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否'

if(DATEDIFF(day,getdate(),@应还日期) < 1) -- 在最后借期一天才可续借，并只能续借一次，加30天

begin

update 借书记录表

set 应还日期 = dateadd(day,30,@应还日期)

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否';

update 借书记录表

set 已续借 = '是'

where 学号 = @学号 and 书号 = @书号 and 已归还 = '否'

end

end

go

--还书

create proc 还书 @书号 varchar(10),@管理员号 varchar(5) --还书可以不需要学号

as

begin

declare @学号 varchar(20);

select @学号 = 学号

from 借书记录表

where 书号 = @书号 and 已归还 = '否'

insert into 还书记录表 --具体细节操作触发前面写的触发器

values(@学号,@书号,@管理员号,getdate())

end

exec 借书 '201900800510','2021042201','21001'; --两行受影响

exec 还书 '2021042201','21001'; --三行受影响

--经检验功能无误

# 6．总结

设计的数据库能够满足需求，并遵循了数据库建立的科学过程，具有实体完整性和参照完整性等，建立了视图，方便了信息查询，并用触发器和存储过程等处理使得数据库成为各表信息关联的统一的整体，并能较高效的处理和查询信息；

但是系统还比较初级与简单，应该还有细节可以优化，这个小项目自己不仅复习了数据库的知识，巩固了知识，还学习了新的函数以及获得了很多新的经验；

# 7．参考资料

菜鸟教程，博客园，CSDN等网站；

<https://blog.csdn.net/rongtaoup/article/details/82183743>

<https://blog.csdn.net/qq_24188927/article/details/83273944>

<https://blog.csdn.net/weixin_43808478/article/details/105709686?utm_medium=distribute.pc_relevant.none-task-blog-baidujs_title-0&spm=1001.2101.3001.4242>

<https://www.runoob.com/sql/sql-dates.html>

书籍：数据库系统概论

致谢：感谢老师的悉心指导和解答疑惑和网上的博主；