

实验六：SQL 综合实验

一、实验目的

1. 熟练运用基础的 SQL 语句，包括数据库的创建，数据的插入、修改、删除、查询操作，数据库表的完整性，存储过程的创建和调用、数据库表索引的创建(*)；
2. 掌握 SQL 语句常见语法错误的调试方法。

二、实验要求说明

1. 完成下列 SQL 综合练习的所有小题内容（序号后标有*的小题是选做题，不强制要求完成），要求提交的实验报告中包含每个小题的题干和相应的实验结果截图；
2. 实验截图中应该包含有完整的 SQL 执行语句和输出的实验结果；
3. 数据库表的表名、字段名要求与提供的表格和附录一中的代码片段保持一致，表内的字段类型不做具体要求，合理即可；
4. 插入数据可以参考附录二中的代码片段，也可以自行定义，能够与后续题意相符即可。

三、实验内容

1. 建表（见附录一），表内字段的类型可以自行定义（合理即可），注意建表时不要忽略各表的主键约束和表间的外键约束；

表名:Book	
字段	说明
bno	书号
bname	书名
author	作者
price	单价

表名:Student	
字段	说明
sno	学号
sname	姓名
grade	年级

P. S. Borrow 表的 bstate 字段（用灰色标注的）是第 9 小题需要额外添加的字段，此处建表时可暂时忽略；

表名:Borrow	
字段	说明
sno	学号, 外键
bno	书号, 外键
rdate	还书截至日期，现实日期小于这个日期视为图书借阅未到期，现实日期大于这个日期视为图书借阅已过期
bstate	借阅状态

2. 插入样例数据（见附录二），当然样例数据也可自行定义，能够与后续题意相符即可；
3. 查询书名中包含”程序设计”的图书信息，输出所有信息（包括书名、书号、作者、单价），并按照单价降序排列；
4. 查询借阅了书名为”数据库原理及应用”的学生信息，输出该学生的学号、姓名和年级；

5. 统计每个学生借书信息，输出每个学生的学号、借书书名和还书日期；
6. 查询所有借阅已过期图书的信息，输出学生姓名、书名和还书日期；

P. S. 借阅已过期图书指的是“现实日期”大于还书截止日期字段的图书，“现实日期”可以自行指定，附录二中的数据样例将 11 月 8 日作为“现实日期”；

7. 查询没有借阅过书的学生信息，输出学生姓名和学号；
8. 查询借了“Java 程序设计”但没有借“数据库原理及应用”的读者信息，输出这些学生的学号，并按照学号升序排列；
9. 创建一个过程，使之能够实现如下功能：
 - a) 修改借阅表，增加字段“借阅状态”（字段名为“bstate”，数据类型可自行定义），字段含义为表示图书的借阅状态是否已经过期；
 - b) 并根据表中已有数据为该字段赋值（所赋的值与表定义时的数据类型保持一致即可，比如可以定义已到期图书的“借阅状态”为 True，未到期图书的“借阅状态”为 False），要求使用 if 语句进行条件判断；

P. S. MySQL 创建存储过程的语法可参考如下代码片段，其中存储过程的名称、是否带参数、参数的名称类型自行决定，合理即可；

```
# 设置分隔符为$$，这样存储过程中出现的分号(;)不会被当成分隔符
delimiter $$
```

```
# 创建存储过程
create procedure update_bstate()
begin
# 使用分隔符$$表示语句的结束
end$$
```

```
# 重新设置分隔符为分号(;)
delimiter ;
```

```
# 调用存储过程
call update_bstate();
```

10. (*)修改图书表，在 bname 列上增加唯一性索引 bname_index，并按 bname 降序排列；

附录

一、建表样例

```
create table Book (  
    bno varchar(10) primary key,  
    bname varchar(30),  
    author varchar(30),  
    price float  
);  
create table Student (  
    sno varchar(10) primary key,  
    sname varchar(30),  
    grade varchar(5)  
);  
create table Borrow (  
    sno varchar(10),  
    bno varchar(10),  
    rdate datetime,  
    primary key(sno, bno),  
    foreign key(sno) references Student(sno) on delete cascade,  
    foreign key(bno) references Book(bno) on delete cascade  
);
```

二、插入数据样例

```
insert into Book values  
    ('T1001', 'Java 程序设计', '李新民', 23.5),  
    ('T1002', '数据库原理及应用', '王珊', 27),  
    ('T1003', 'Java 高级编程', '陈海', 23.5),  
    ('T1004', '大学英语', '张宁', 18.5),  
    ('T1005', 'C++程序设计', '马品三', 33.5),  
    ('T1006', '数据结构', '刘子单', 35);  
insert into Student values  
    ('K001', '张三', '大一'),  
    ('K002', '李四', '大二'),  
    ('K003', '王五', '大三'),  
    ('K004', '赵六', '大四'),  
    ('K005', '刘七', '大四');  
insert into Borrow values  
    ('K001', 'T1006', '2022-10-9'),  
    ('K001', 'T1001', '2022-3-1'),  
    ('K001', 'T1004', '2022-5-7'),  
    ('K002', 'T1002', '2022-6-9'),  
    ('K002', 'T1003', '2022-12-5'),  
    ('K002', 'T1001', '2022-11-3'),  
    ('K003', 'T1005', '2022-9-4'),  
    ('K004', 'T1002', '2022-2-5');
```