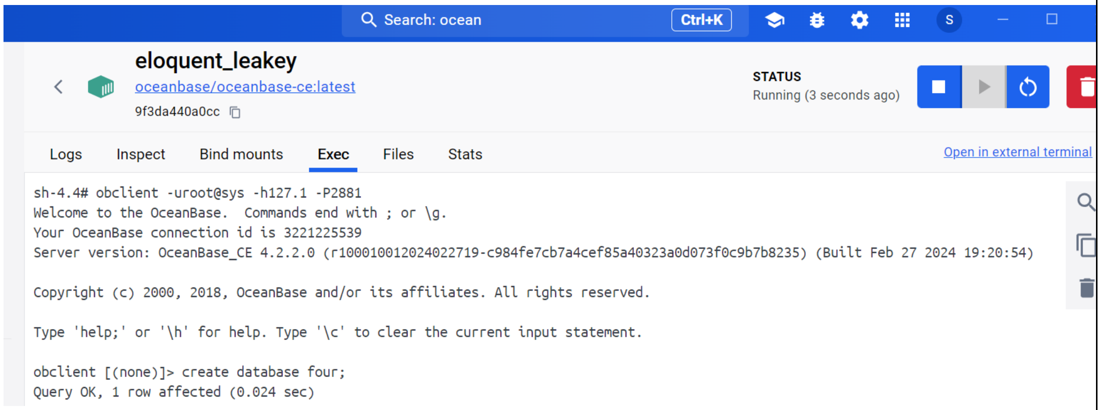
《数据库系统原理》实验报告（4）

题目： 数据库安全性

学号		姓名		日期	2024.4.16

实验环境：



```
sh-4.4# obclient -uroot@sys -h127.1 -P2881
Welcome to the OceanBase.  Commands end with ; or \g.
Your OceanBase connection id is 3221225539
Server version: OceanBase_CE 4.2.2.0 (r100010012024022719-c984fe7cb7a4cef85a40323a0d073f0c9b7b8235) (Built Feb 27 2024 19:20:54)

Copyright (c) 2000, 2018, OceanBase and/or its affiliates. All rights reserved.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

obclient [(none)]> create database four;
Query OK, 1 row affected (0.024 sec)
```

实验步骤及结果截图：

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；

```
obclient [(none)]> use four;
Database changed
obclient [four]> create table Book (
    ->     Book_no varchar(10) primary key,
    ->     Book_name varchar(30),
    ->     Author varchar(30),
    ->     Price float
    -> );
Query OK, 0 rows affected (0.049 sec)

obclient [four]> create table Student (
    ->     Student_no varchar(10) primary key,
    ->     Student_name varchar(30),
    ->     Grade varchar(10)
    -> );
Query OK, 0 rows affected (0.035 sec)

obclient [four]> create table Borrow (
    ->     Student_no varchar(10),
    ->     Book_no varchar(10),
    ->     Return_date datetime,
    ->     primary key(Student_no, Book_no),
    ->     foreign key(Student_no) references Student(Student_no) on delete cascade,
    ->     foreign key(Book_no) references Book(Book_no) on delete cascade
    -> );
Query OK, 0 rows affected (0.088 sec)
```

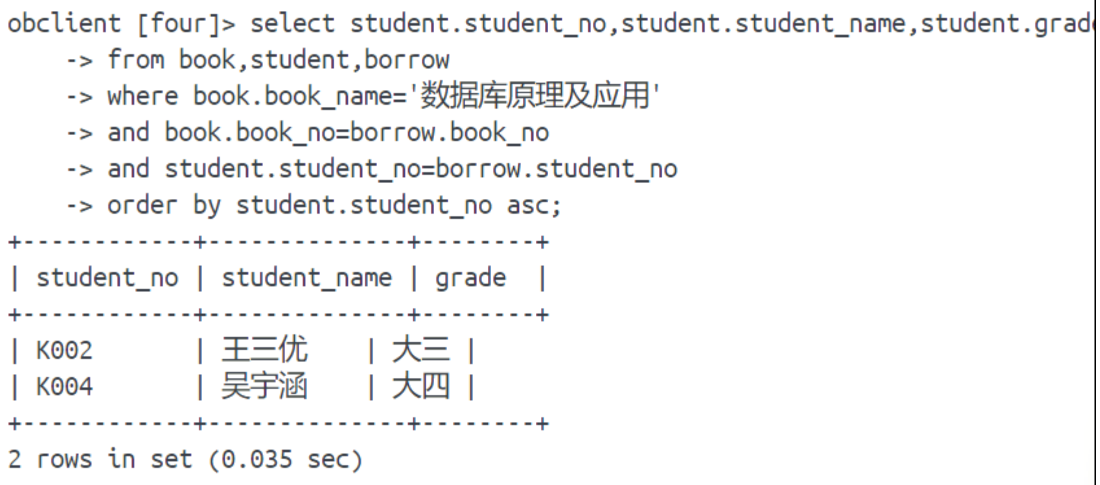
2. 插入样例数据（见附录二）；

```
obclient [four]> insert into Book values
    ->     ('T1001','Java程序设计','李新珊',89.5),
    ->     ('T1002','数据库原理及应用','王敏',39),
    ->     ('T1003','Java高级编程','陈明海',63.5),
    ->     ('T1004','专业英语','张倪宁',23.1),
    ->     ('T1005','C++程序设计','马天颖',83.2),
    ->     ('T1006','编译原理','王鑫单',65);
Query OK, 6 rows affected (0.010 sec)
Records: 6  Duplicates: 0  Warnings: 0

obclient [four]> insert into Student values
    ->     ('K005','张鑫翁','大一'),
    ->     ('K003','徐晨皓','大二'),
    ->     ('K002','王三优','大三'),
    ->     ('K001','刘孔阴','大三'),
    ->     ('K004','吴宇涵','大四');
Query OK, 5 rows affected (0.006 sec)
Records: 5  Duplicates: 0  Warnings: 0

obclient [four]> insert into Borrow values
    ->     ('K001','T1006','2023-10-9'),
    ->     ('K001','T1001','2024-3-1'),
    ->     ('K002','T1002','2023-10-9'),
    ->     ('K002','T1003','2024-4-5'),
    ->     ('K002','T1001','2023-11-3'),
    ->     ('K003','T1005','2024-1-4'),
    ->     ('K004','T1002','2024-2-5');
Query OK, 7 rows affected (0.007 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

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；

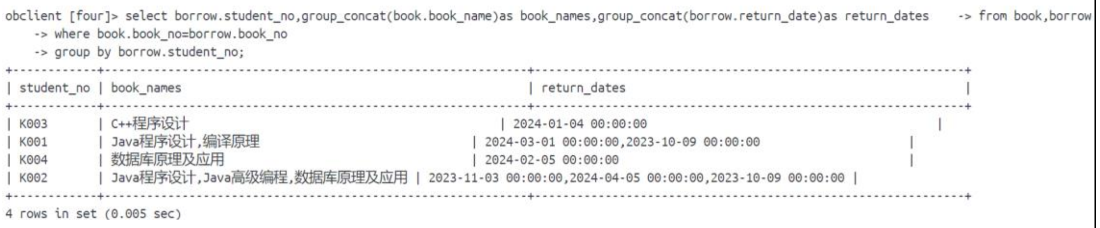
```
obclient [four]> select *
    -> from book
    -> where book_name like '%程序设计%'
    -> order by price desc;
+---------+------------------+------------+-------+
| Book_no | Book_name        | Author     | Price |
+---------+------------------+------------+-------+
| T1001   | Java程序设计 | 李新珊 |  89.5 |
| T1005   | C++程序设计  | 马天颖 |  83.2 |
+---------+------------------+------------+-------+
2 rows in set (0.009 sec)
```

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，并按照学号升序排列；



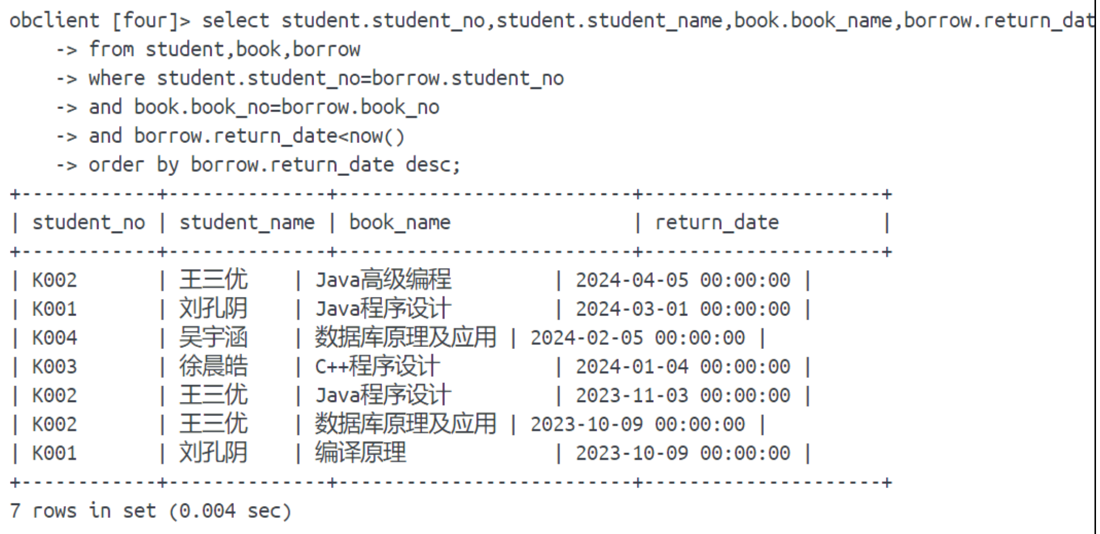
```
obclient [four]> select student.student_no,student.student_name,student.grad
    -> from book,student,borrow
    -> where book.book_name='数据库原理及应用'
    -> and book.book_no=borrow.book_no
    -> and student.student_no=borrow.student_no
    -> order by student.student_no asc;
+------------+--------------+--------+
| student_no | student_name | grade  |
+------------+--------------+--------+
| K002       | 王三优       | 大三 |
| K004       | 吴宇涵       | 大四 |
+------------+--------------+--------+
2 rows in set (0.035 sec)
```

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；



```
obclient [four]> select borrow.student_no,group_concat(book.book_name)as book_names,group_concat(borrow.return_date)as return_dates    -> from book,borrow
    -> where book.book_no=borrow.book_no
    -> group by borrow.student_no;
+------------+-----------------------------------------------------------+----------------------------------------------------------------+
| student_no | book_names                                                | return_dates                                                   |
+------------+-----------------------------------------------------------+----------------------------------------------------------------+
| K003       | C++程序设计                                    | 2024-01-04 00:00:00                                            |
| K001       | Java程序设计,编译原理                      | 2024-03-01 00:00:00,2023-10-09 00:00:00                        |
| K004       | 数据库原理及应用                           | 2024-02-05 00:00:00                                            |
| K002       | Java程序设计,Java高级编程,数据库原理及应用 | 2023-11-03 00:00:00,2024-04-05 00:00:00,2023-10-09 00:00:00 |
+------------+-----------------------------------------------------------+----------------------------------------------------------------+
4 rows in set (0.005 sec)
```

6. 查询所有借阅已过期图书的信息，输出学生学号、姓名、书名和还书日期，并按还书日期降序排列；



```
obclient [four]> select student.student_no,student.student_name,book.book_name,borrow.return_dat
    -> from student,book,borrow
    -> where student.student_no=borrow.student_no
    -> and book.book_no=borrow.book_no
    -> and borrow.return_date<now()
    -> order by borrow.return_date desc;
+------------+--------------+--------------------------+---------------------+
| student_no | student_name | book_name                | return_date         |
+------------+--------------+--------------------------+---------------------+
| K002       | 王三优       | Java高级编程         | 2024-04-05 00:00:00 |
| K001       | 刘孔阴       | Java程序设计         | 2024-03-01 00:00:00 |
| K004       | 吴宇涵       | 数据库原理及应用 | 2024-02-05 00:00:00 |
| K003       | 徐晨皓       | C++程序设计          | 2024-01-04 00:00:00 |
| K002       | 王三优       | Java程序设计         | 2023-11-03 00:00:00 |
| K002       | 王三优       | 数据库原理及应用 | 2023-10-09 00:00:00 |
| K001       | 刘孔阴       | 编译原理             | 2023-10-09 00:00:00 |
+------------+--------------+--------------------------+---------------------+
7 rows in set (0.004 sec)
```

7. 查询没有借阅过书的学生信息，输出学生姓名和学号；

left join 保证了即使在借书表中找不到对应的记录，学生表中的所有学生信息也会被包含在结果中。

```
obclient [four]> select student.student_name,student.student_no
    -> from student
    -> left join borrow on student.student_no=borrow.student_no
    -> where borrow.student_no is null;
+--------------+------------+
| student_name | student_no |
+--------------+------------+
| 张鑫翁       | K005       |
+--------------+------------+
1 row in set (0.007 sec)
```

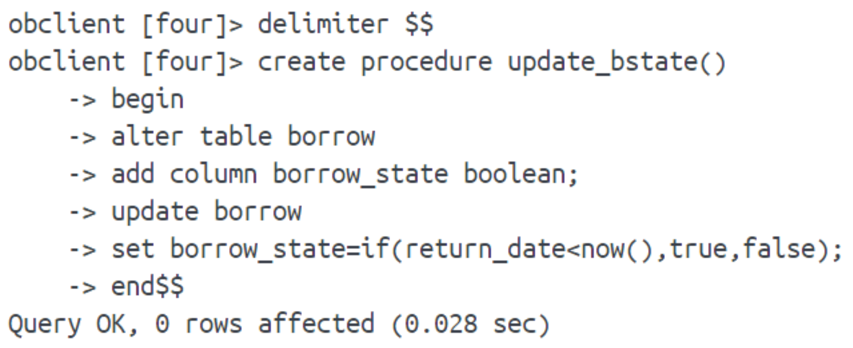
8. 查询借了"Java 程序设计"但没有借"数据库原理及应用"的读者信息，输出这些学生的学号，并按照学号升序排列；

```
obclient [four]> select student_no
    -> from borrow
    -> where book_no=
    -> (select book_no from book where book_name='Java程序设计')
    -> and student_no not in
    -> (select student_no from borrow where book_no=
    -> (select book_no from book where book_name='数据库原理及应用')
    -> )
    -> order by student_no asc;
+------------+
| student_no |
+------------+
| K001       |
+------------+
1 row in set (0.020 sec)
```

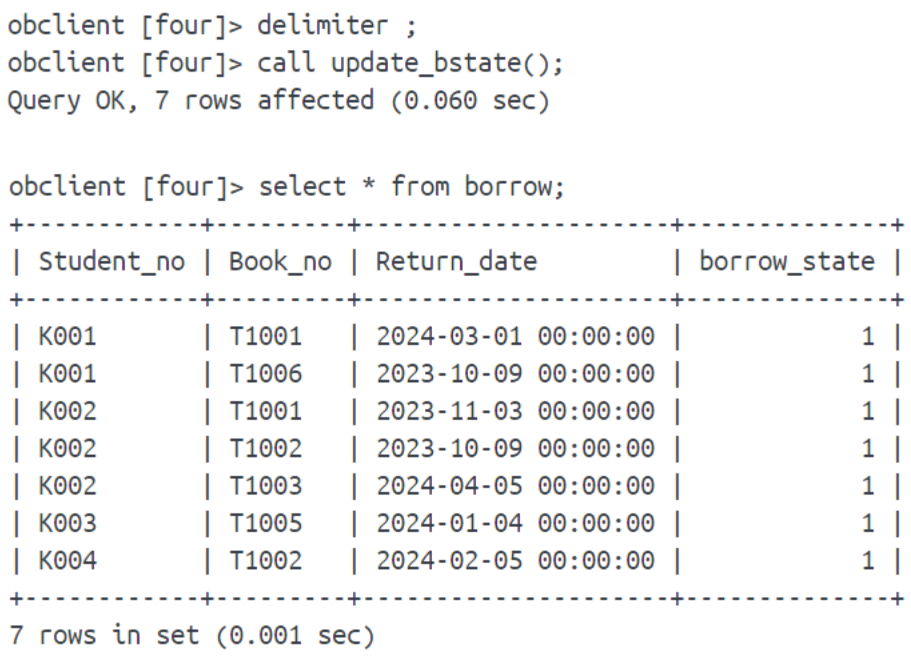
9. 创建一个过程，使之能够实现如下功能：

10. 修改借阅表，增加字段"借阅状态"（字段名为"Borrow_state"），字段含义为表示图书的借阅状态是否已经过期；

11. 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；

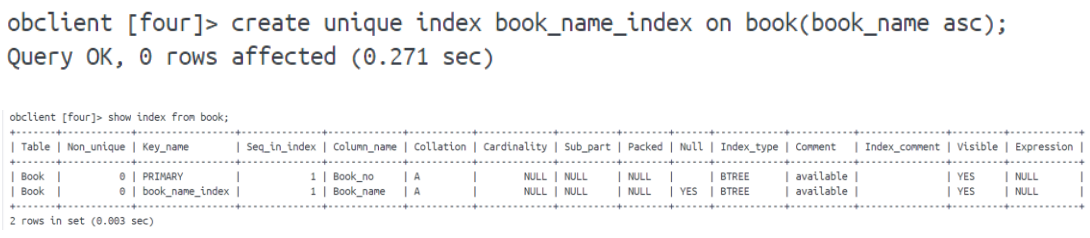


```
obclient [four]> delimiter $$
obclient [four]> create procedure update_bstate()
    -> begin
    -> alter table borrow
    -> add column borrow_state boolean;
    -> update borrow
    -> set borrow_state=if(return_date<now(),true,false);
    -> end$$
Query OK, 0 rows affected (0.028 sec)
```



```
obclient [four]> delimiter ;
obclient [four]> call update_bstate();
Query OK, 7 rows affected (0.060 sec)

obclient [four]> select * from borrow;
+------------+---------+---------------------+--------------+
| Student_no | Book_no | Return_date         | borrow_state |
+------------+---------+---------------------+--------------+
| K001       | T1001   | 2024-03-01 00:00:00 |            1 |
| K001       | T1006   | 2023-10-09 00:00:00 |            1 |
| K002       | T1001   | 2023-11-03 00:00:00 |            1 |
| K002       | T1002   | 2023-10-09 00:00:00 |            1 |
| K002       | T1003   | 2024-04-05 00:00:00 |            1 |
| K003       | T1005   | 2024-01-04 00:00:00 |            1 |
| K004       | T1002   | 2024-02-05 00:00:00 |            1 |
+------------+---------+---------------------+--------------+
7 rows in set (0.001 sec)
```

12. (*)修改图书表，在 Book_name 列上增加唯一性索引 Book_name_index，并按 Book_name 降序排列；



```
obclient [four]> create unique index book_name_index on book(book_name asc);
Query OK, 0 rows affected (0.271 sec)

obclient [four]> show index from book;
+-------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+-----------+---------------+---------+------------+
| Table | Non_unique | Key_name        | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment   | Index_comment | Visible | Expression |
+-------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+-----------+---------------+---------+------------+
| Book  |          0 | PRIMARY         |            1 | Book_no     | A         |        NULL | NULL     | NULL   |      | BTREE      | available |               | YES     | NULL       |
| Book  |          0 | book_name_index |            1 | Book_name   | A         |        NULL | NULL     | NULL   | YES  | BTREE      | available |               | YES     | NULL       |
+-------+------------+-----------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+-----------+---------------+---------+------------+
2 rows in set (0.003 sec)
```

出现的问题：

1. 如何归并不同行的信息

例如第 **5** 题

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；

需要把同一个同学的多行借书信息汇总输出

2. 如何比较时间

解决方案：

1. 利用 **group_concat** 函数

select borrow.student_no , group_concat (book.book_name) as book_names , group_concat (borrow.return_date) as return_dates

2.利用 **now()**函数可以比较记录时间和当前时间