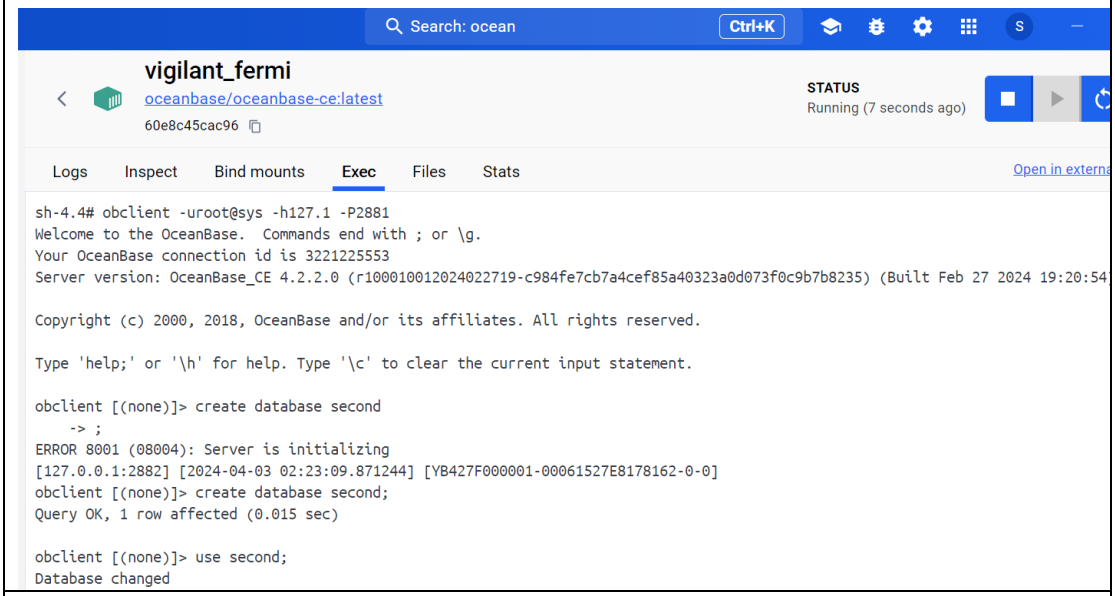
《数据库系统原理》实验报告（2）

题目：DDL 实验语言

学号		姓名		日期	2024.4.3

实验环境：



实验步骤及结果截图：

1. 参照下表中内容，在数据库中建立教学管理系统相关数据表。

表名：students		
字段	**类型**	**约束**
No	Integer	主键，非空
Name	Varchar	非空
Gender	Varchar	非空，只能为 Female 或者 Male
Age	Integer	非空
D_no	Integer	非空，外键

表名：depts		
字段	**类型**	**约束**
No	Integer	主键，非空
Name	Varchar	非空

表名：courses		
字段	**类型**	**约束**
No	Integer	主键，非空
Name	Varchar	非空
Credit	Integer	非空
D_no	Integer	非空，外键

表名：scores		
字段	**类型**	**约束**
S_no	Integer	非空，外键
C_no	Integer	非空，外键
Score	Integer	非空

解答：

```
obclient [second]> create table depts(
    -> no int not null,
    -> name varchar(20) not null,
    -> primary key (no)
    -> );
Query OK, 0 rows affected (0.037 sec)

obclient [second]> create table students(
    -> no int not null,
    -> name varchar(20) not null,
    -> gender varchar(20) not null,
    -> age int not null,
    -> d_no int not null,
    -> primary key (no),
    -> foreign key (d_no) references depts (no),
    -> check (gender in ('female','male'))
    -> );
Query OK, 0 rows affected (0.082 sec)

obclient [second]> create table courses(
    -> no int not null,
    -> name varchar(20) not null,
    -> credit int not null,
    -> d_no int not null,
    -> primary key (no),
    -> foreign key (d_no) references depts (no)
    -> );
Query OK, 0 rows affected (0.045 sec)

obclient [second]> create table scores(
    -> s_no int not null,
    -> c_no int not null,
    -> score int not null,
    -> foreign key (s_no) references students(no),
    -> foreign key (c_no) references courses(no)
    -> );
Query OK, 0 rows affected (0.096 sec)
```

一、 插入数据样例

```
insert into students values
    (202415120,'Michael','Male',21,3),
    (202415121,'Tommy','Male',20,1),
  (202415122,'Michael','Female',19,1),
  (202415123,'Ale','Female',18,2),
  (202415125,'Billy','Male',19,3);
insert into depts values
  (1,'Computer Science'),
  (2,'Mathematics'),
  (3,'English'),
  (4,'Management');
insert into courses values
  (1,'Database',5,1),
  (2,'Mathematics',2,2),
  (3,'Information System',1,4),
  (4,'Operating System',6,1),
  (5,'Data Structure',4,1),
  (6,'Data Processing',2,4),
  (7,'PASCAL',3,1);
insert into scores values
  (202415121,1,92),
  (202415121,2,85),
  (202415121,3,88),
  (202415122,2,90),
  (202415122,3,80);
```

```
obclient [second]> insert into depts values
    -> (1,'Computer Science'),
    -> (2,'Mathematics'),
    -> (3,'English'),
    -> (4,'Management');
Query OK, 4 rows affected (0.005 sec)
Records: 4  Duplicates: 0  Warnings: 0

obclient [second]> insert into students values
    ->  (202415120,'Michael','Male',21,3),
    ->  (202415121,'Tommy','Male',20,1),
    -> (202415122,'Michael','Female',19,1),
    -> (202415123,'Ale','Female',18,2),
    -> (202415125,'Billy','Male',19,3);
Query OK, 5 rows affected (0.008 sec)
Records: 5  Duplicates: 0  Warnings: 0

obclient [second]> insert into courses values
    -> (1,'Database',5,1),
    -> (2,'Mathematics',2,2),
    -> (3,'Information System',1,4),
    -> (4,'Operating System',6,1),
    -> (5,'Data Structure',4,1),
    -> (6,'Data Processing',2,4),
    -> (7,'PASCAL',3,1);
Query OK, 7 rows affected (0.007 sec)
Records: 7  Duplicates: 0  Warnings: 0
```

```
obclient [second]> insert into scores values
    -> (202415121,1,92),
    -> (202415121,2,85),
    -> (202415121,3,88),
    -> (202415122,2,90),
    -> (202415122,3,80);
Query OK, 5 rows affected (0.012 sec)
Records: 5  Duplicates: 0  Warnings: 0


obclient [second]> select *
    -> from students;
+-----------+---------+--------+-----+------+
| no        | name    | gender | age | d_no |
+-----------+---------+--------+-----+------+
| 202415120 | Michael | Male   |  21 |    3 |
| 202415121 | Tommy   | Male   |  20 |    1 |
| 202415122 | Michael | Female |  19 |    1 |
| 202415123 | Ale     | Female |  18 |    2 |
| 202415125 | Billy   | Male   |  19 |    3 |
+-----------+---------+--------+-----+------+
5 rows in set (0.003 sec)

obclient [second]> select * from depts;
+----+------------------+
| no | name             |
+----+------------------+
|  1 | Computer Science |
|  2 | Mathematics      |
|  3 | English          |
|  4 | Management       |
+----+------------------+
4 rows in set (0.002 sec)


obclient [second]> select * from courses;
+----+--------------------+--------+------+
| no | name               | credit | d_no |
+----+--------------------+--------+------+
|  1 | Database           |      5 |    1 |
|  2 | Mathematics        |      2 |    2 |
|  3 | Information System |      1 |    4 |
|  4 | Operating System   |      6 |    1 |
|  5 | Data Structure     |      4 |    1 |
|  6 | Data Processing    |      2 |    4 |
|  7 | PASCAL             |      3 |    1 |
+----+--------------------+--------+------+
7 rows in set (0.009 sec)
```

```
obclient [second]> select * from scores;
+-----------+------+-------+
| s_no      | c_no | score |
+-----------+------+-------+
| 202415121 |    1 |    92 |
| 202415121 |    2 |    85 |
| 202415121 |    3 |    88 |
| 202415122 |    2 |    90 |
| 202415122 |    3 |    80 |
+-----------+------+-------+
5 rows in set (0.005 sec)
```

2. 使用比较运算符查询所有年龄在 21 岁以下的学生姓名及其年龄

```
obclient [second]> select name,age
    -> from students
    -> where age<21;
+---------+-----+
| name    | age |
+---------+-----+
| Tommy   |  20 |
| Michael |  19 |
| Ale     |  18 |
| Billy   |  19 |
+---------+-----+
4 rows in set (0.002 sec)
```

3. 查询选 2 号课程（S_no='2'）且成绩在 80--90 的学号

```
obclient [second]> select s_no
    -> from scores
    -> where c_no='2' and score between 80 and 90;
+-----------+
| s_no      |
+-----------+
| 202415121 |
| 202415122 |
+-----------+
2 rows in set (0.005 sec)
```

4. 查询姓名第二个字母是'e'的学生姓名

```
obclient [second]> select name
    -> from students
    -> where name like '_e%';
Empty set (0.002 sec)
```

5. 查询全体男学生的学号、系、年龄结果按所在的系升序排列，同一系中的学生按年龄降序排列

```
obclient [second]> select no,d_no,age
    -> from students
    -> where gender='male'
    -> order by d_no asc,age desc;
+-----------+------+-----+
| no        | d_no | age |
+-----------+------+-----+
| 202415121 |    1 |  20 |
| 202415120 |    3 |  21 |
| 202415125 |    3 |  19 |
+-----------+------+-----+
3 rows in set (0.003 sec)
```

6. 查询女学生的总人数和平均年龄

```
obclient [second]> select count(distinct no),avg(age)
    -> from students
    -> where gender='female';
+--------------------+----------+
| count(distinct no) | avg(age) |
+--------------------+----------+
|                  2 |  18.5000 |
+--------------------+----------+
1 row in set (0.009 sec)
```

7. 查询选修 2 号课程并及格（分数大于 60）的学生的最高分、最低分及总分

```
obclient [second]> select max(score),min(score),sum(score)
    -> from scores
    -> where c_no='2' and score>60;
+------------+------------+------------+
| max(score) | min(score) | sum(score) |
+------------+------------+------------+
|         90 |         85 |        175 |
+------------+------------+------------+
1 row in set (0.002 sec)
```

8. 向 scores 表中插入一条记录（200266666,1,85）

```
obclient [second]> insert into students values(200266666,'mike','male',21,2);
Query OK, 1 row affected (0.005 sec)

obclient [second]> insert into scores(s_no,c_no,score) value(200266666,1,85);
Query OK, 1 row affected (0.002 sec)
```

```
obclient [second]> select *
    -> from scores;
+-----------+------+-------+
| s_no      | c_no | score |
+-----------+------+-------+
| 202415121 |    1 |    92 |
| 202415121 |    2 |    85 |
| 202415121 |    3 |    88 |
| 202415122 |    2 |    90 |
| 202415122 |    3 |    80 |
| 200266666 |    1 |    85 |
+-----------+------+-------+
6 rows in set (0.001 sec)
```

9. 求每个学生（号）的平均成绩，并将其超过 70 分的按学号输出

```
obclient [second]> select s_no,avg(score)
    -> from scores
    -> group by s_no
    -> having avg(score)>70
    -> order by s_no;
+-----------+------------+
| s_no      | avg(score) |
+-----------+------------+
| 200266666 |    85.0000 |
| 202415121 |    88.3333 |
| 202415122 |    85.0000 |
+-----------+------------+
3 rows in set (0.003 sec)
```

10. 查询选修了课程 1 或者选修了课程 3 的学生姓名

```
obclient [second]> select *
    -> from students;
+-----------+---------+--------+-----+------+
| no        | name    | gender | age | d_no |
+-----------+---------+--------+-----+------+
| 200266666 | mike    | male   |  21 |    2 |
| 202415120 | Michael | Male   |  21 |    3 |
| 202415121 | Tommy   | Male   |  20 |    1 |
| 202415122 | Michael | Female |  19 |    1 |
| 202415123 | Ale     | Female |  18 |    2 |
| 202415125 | Billy   | Male   |  19 |    3 |
+-----------+---------+--------+-----+------+
6 rows in set (0.001 sec)

obclient [second]> select * from scores;
+-----------+------+-------+
| s_no      | c_no | score |
+-----------+------+-------+
| 202415121 |    1 |    92 |
| 202415121 |    2 |    85 |
| 202415121 |    3 |    88 |
| 202415122 |    2 |    90 |
| 202415122 |    3 |    80 |
| 200266666 |    1 |    85 |
+-----------+------+-------+
6 rows in set (0.001 sec)


obclient [second]> (select distinct name
    -> from scores,students
    -> where c_no='1' and scores.s_no=students.no)
    -> union
    -> (select distinct name
    -> from scores,students
    -> where c_no='3' and scores.s_no=students.no);
+---------+
| name    |
+---------+
| mike    |
| Tommy   |
| Michael |
+---------+
3 rows in set (0.000 sec)
```

11. 查询既选修了课程 1 又选修了课程 2 的学生姓名

```
obclient [second]> select *
    -> from students;
+-----------+---------+--------+-----+------+
| no        | name    | gender | age | d_no |
+-----------+---------+--------+-----+------+
| 200266666 | mike    | male   |  21 |    2 |
| 202415120 | Michael | Male   |  21 |    3 |
| 202415121 | Tommy   | Male   |  20 |    1 |
| 202415122 | Michael | Female |  19 |    1 |
| 202415123 | Ale     | Female |  18 |    2 |
| 202415125 | Billy   | Male   |  19 |    3 |
+-----------+---------+--------+-----+------+
6 rows in set (0.001 sec)

obclient [second]> select * from scores;
+-----------+------+-------+
| s_no      | c_no | score |
+-----------+------+-------+
| 202415121 |    1 |    92 |
| 202415121 |    2 |    85 |
| 202415121 |    3 |    88 |
| 202415122 |    2 |    90 |
| 202415122 |    3 |    80 |
| 200266666 |    1 |    85 |
+-----------+------+-------+
6 rows in set (0.001 sec)


obclient [second]> (select distinct name
    -> from scores,students
    -> where c_no='1' and scores.s_no=students.no)
    -> intersect
    -> (select distinct name
    -> from scores,students
    -> where c_no='2' and scores.s_no=students.no);
+-------+
| name  |
+-------+
| Tommy |
+-------+
1 row in set (0.006 sec)
```

12. 查询选修 Database 这门课最高分学生所在的系名

```
obclient [second]> select depts.name
    -> from scores,students,depts
    -> where scores.s_no=students.no
    -> and students.d_no=depts.no
    -> and score=(select max(score) from scores where c_no=(select no from courses where name ='database'));
+------------------+
| name             |
+------------------+
| Computer Science |
+------------------+
1 row in set (0.008 sec)
```

13. 建立一个包含学生学号，姓名，年龄，以及所在系名的视图（赋予列名为 sno,sname,sage,deptname）【create view】

```
obclient [second]> create view stu(sno,sname,sage,deptname)
    -> as(select students.no,students.name,students.age,depts.name
    -> from students,depts
    -> where students.d_no=depts.no);
Query OK, 0 rows affected (0.062 sec)

obclient [second]> select * from stu;
+-----------+---------+------+------------------+
| sno       | sname   | sage | deptname         |
+-----------+---------+------+------------------+
| 200266666 | mike    |   21 | Mathematics      |
| 202415120 | Michael |   21 | English          |
| 202415121 | Tommy   |   20 | Computer Science |
| 202415122 | Michael |   19 | Computer Science |
| 202415123 | Ale     |   18 | Mathematics      |
| 202415125 | Billy   |   19 | English          |
+-----------+---------+------+------------------+
6 rows in set (0.010 sec)
```

出现的问题：（问题与下面的解决方案每条对应）

1. 插入数据时首先插入了 students 的数据，插入失败

```
obclient [second]> insert into students values
    -> (202415120,'michael','male',21,3),
    -> (202415121,'Tommy','Male',20,1),
    -> (202415122,'Michael','Female',19,1),
    -> (202415123,'Ale','Female',18,2),
    -> (202415125,'Billy','Male',19,3);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails
```

2. 在做插入的小题时，又忽略了插入顺序的问题，欲直接插入题目所需数据

```
obclient [second]> insert into scores(s_no,c_no,score) value(200266666,1,85);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails
```

解决方案：

1. 原因还是外码的问题，因为 students 的 d_no 是参考 depts 的外码，所以需要先把 depts 中对应的数据插入进去才能在 students 插入相应的数据

2. 原因同上，想要在 scores 里面插入一个之前不存在的学号，需要先在 students 里创建该学号对应的信息。

```
obclient [second]> insert into scores(s_no,c_no,score) value(200266666,1,85);
ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails
obclient [second]> insert into students values(200266666,'mike','male',21,2);
Query OK, 1 row affected (0.005 sec)

obclient [second]> insert into scores(s_no,c_no,score) value(200266666,1,85);
Query OK, 1 row affected (0.002 sec)
```