目录

[Sql语句总结 1](#_Toc24623134)

[一.创建数据库和表 1](#_Toc24623135)

[二.插入数据(增加数据) 2](#_Toc24623136)

[三.更改数据和删除数据 2](#_Toc24623137)

[四.删除(谨慎操作) 3](#_Toc24623138)

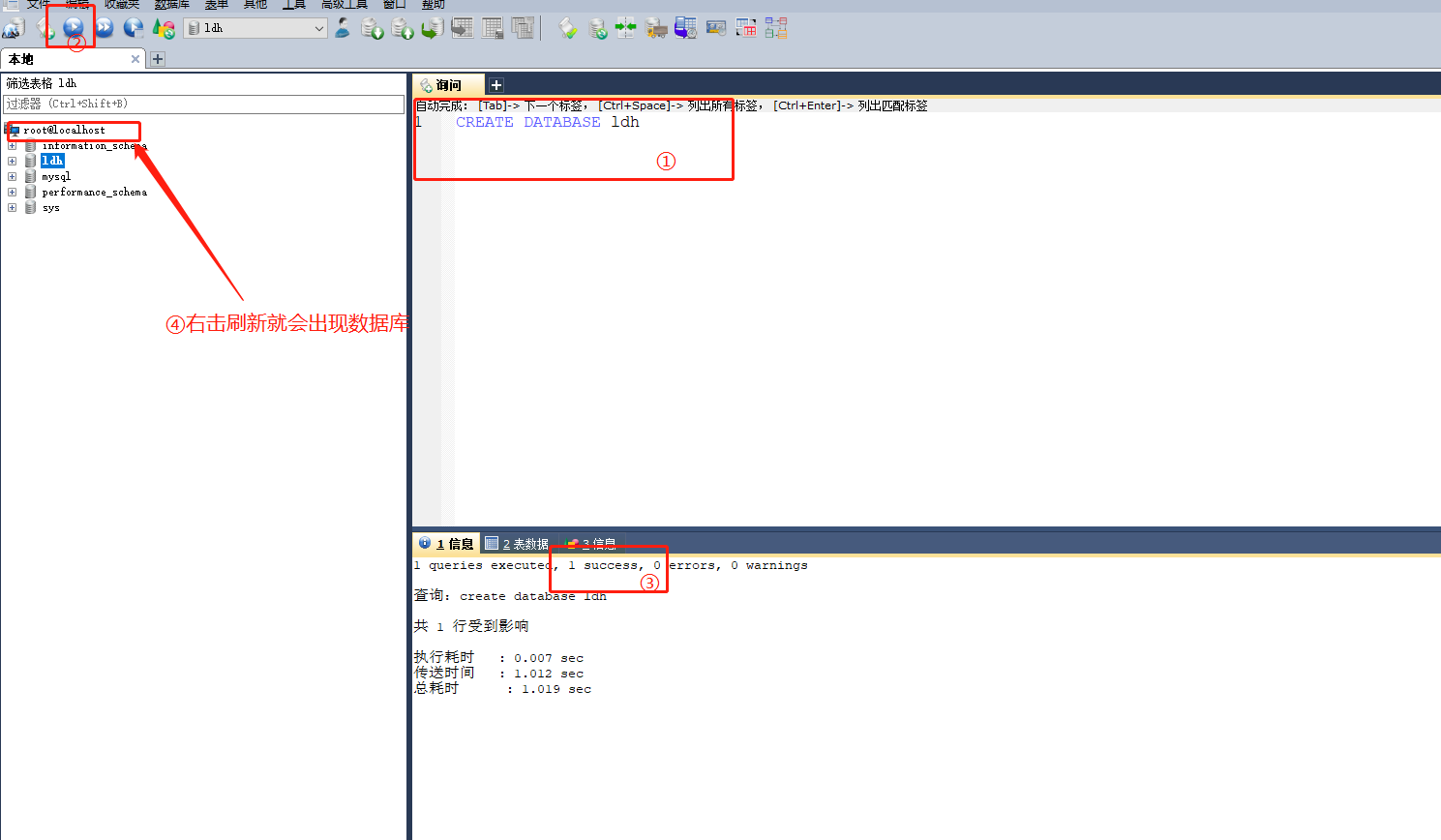
[五.查询和所用到的函数以及条件关键字 3](#_Toc24623139)

[考核练习题： 14](#_Toc24623140)

Sql语句总结

# 一.创建数据库和表

1. 创建数据库 最好用这个sql（create database 库名 charset utf8）



注：库名尽量不要为关键字或者保留字(database,table,select……),自行查询

1. 查看数据库

Show databases

1. 删除数据库

Drop database ldh(库名)

1. 创建表

CREATE TABLE student(

NAME VARCHAR(50),

age SMALLINT,

sex VARCHAR(25)

)

1. 查看表

Show tables

1. 修改表名

RENAME TABLE student(老表名) TO stu(新表名)

1. 删除表

Drop table 表名

# 二.插入数据(增加数据)

1. INSERT INTO stu (namee,age,sex) VALUES ('李四',4,'男')

注：值中为非数值时用单引号，数值不用。

插入多行数据，用多行insert，每行后有分号。

1. 如需对表新增字段

ALTER TABLE stu ADD address VARCHAR(50) AFTER sex

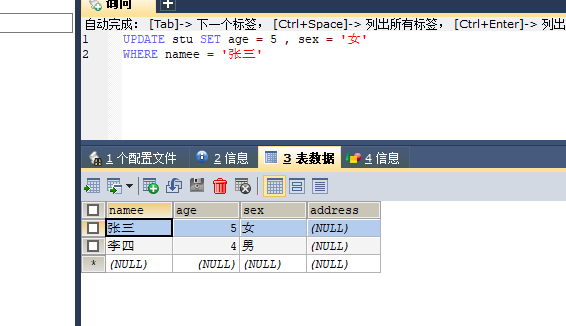
1. 重命名字段

ALTER TABLE stu CHANGE NAME(老字段) na(新字段) VARCHAR(50)(新字段类型)

# 三.更改数据和删除数据

1. Update(更新已存在的表数据)

格式：update + 表名 + set + 要改变的字段+ where+ 条件

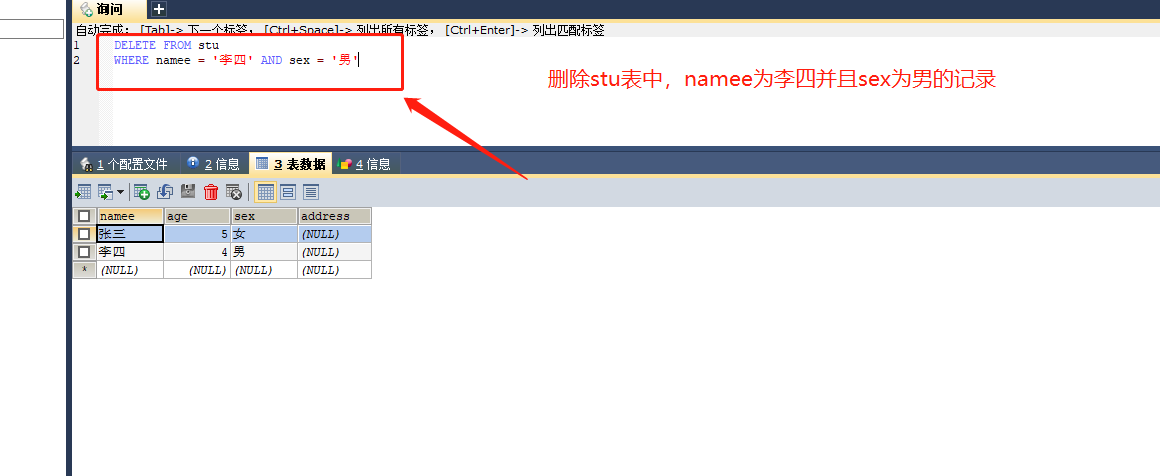


1. alter （修改列，同二中的例子）

# 四.删除(谨慎操作)

1. Delete（删除表中的记录）

delete from + 表名 + where + 条件



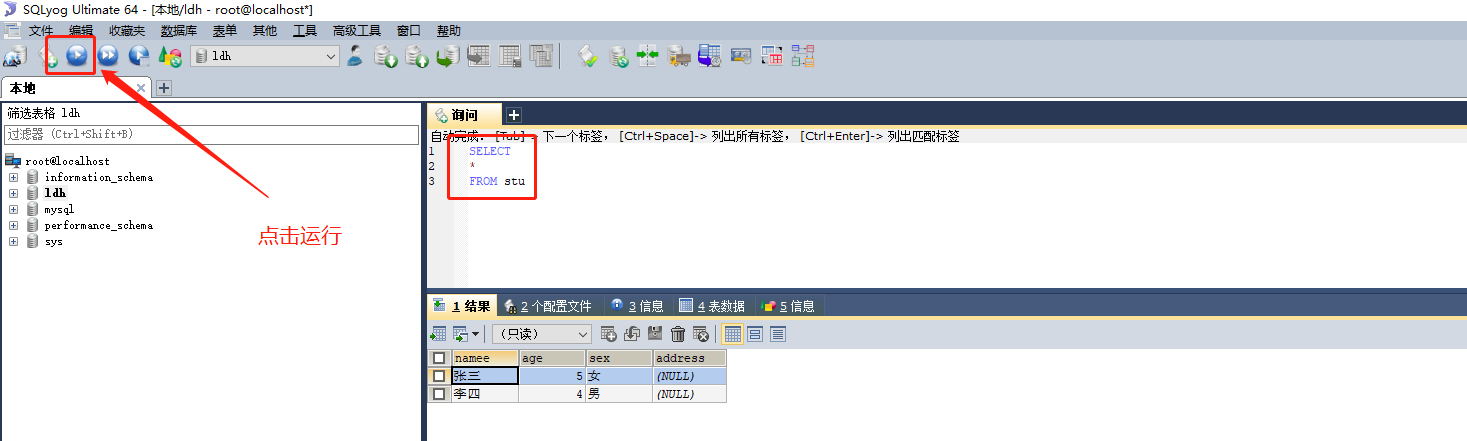
1. drop（删除表、数据库）

drop TABLE + 表名

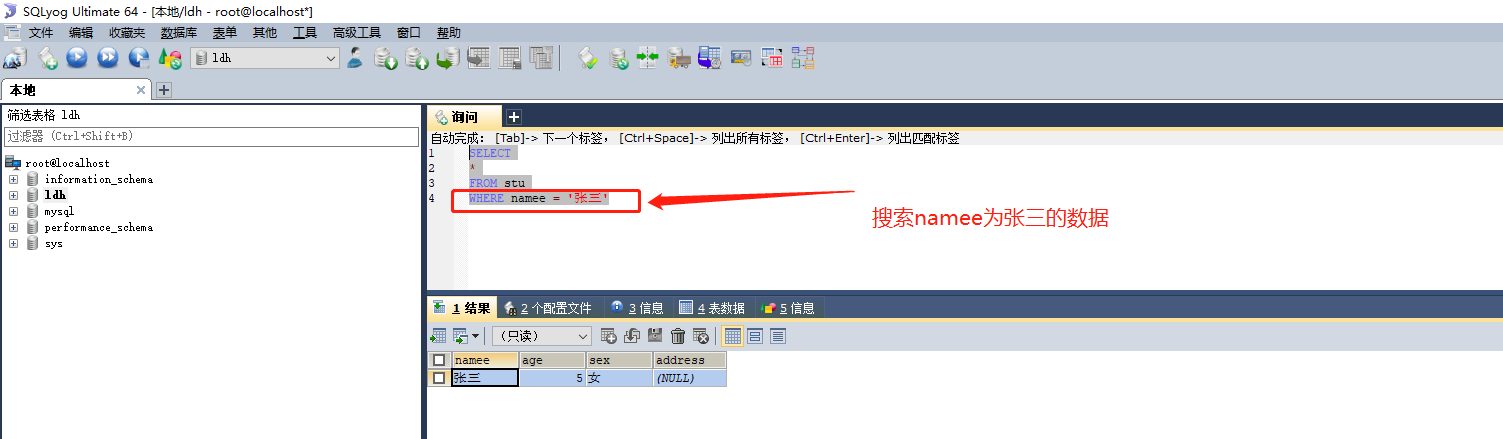
drop DATABASE + 库名

# 五.查询和所用到的函数以及条件关键字

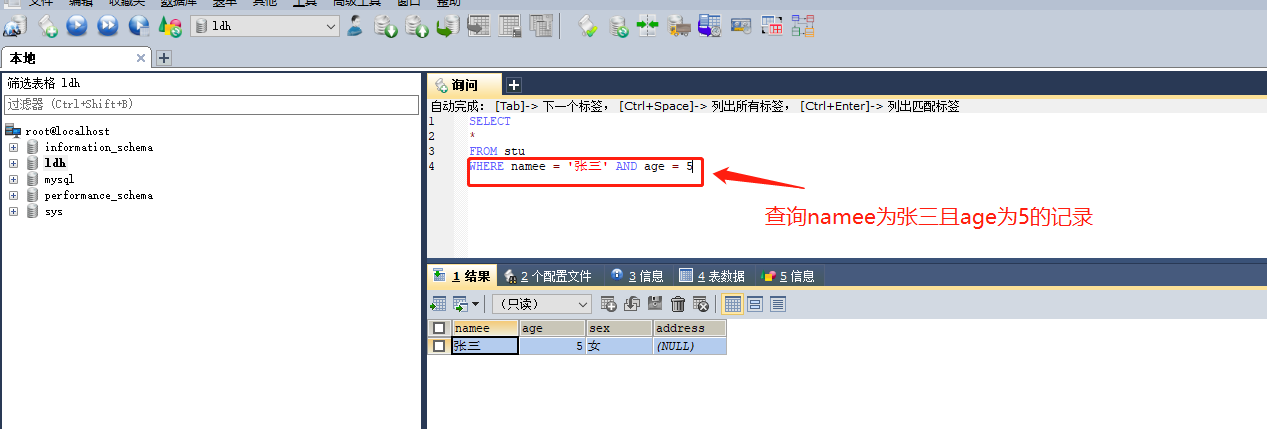
1. 简单查询
   1. Select \* from + 表名(\*表示这个表中的所有字段和数据)



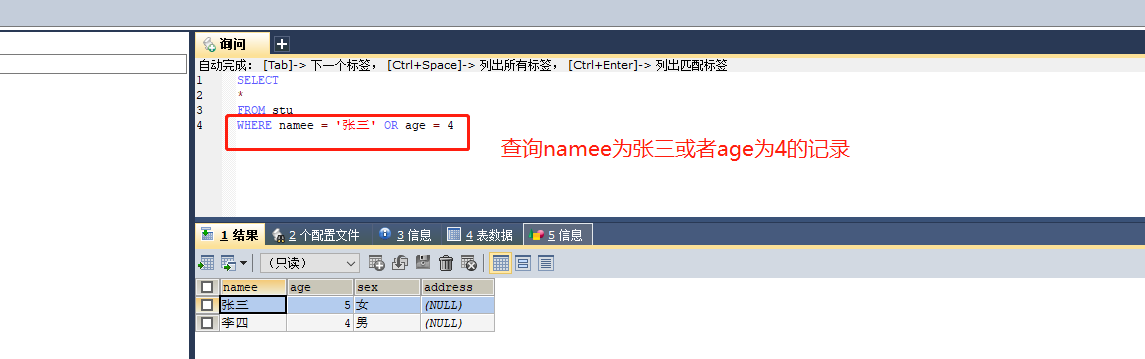
* 1. Select \* from + 表名 + where + 条件



* 1. Select \* from + 表名 + where + 条件1 and 条件2



* 1. Select \* from + 表名 + where 条件1 or 条件2



* 1. Select \* from+ 表名 + where 条件1 and (条件2 or 条件3)

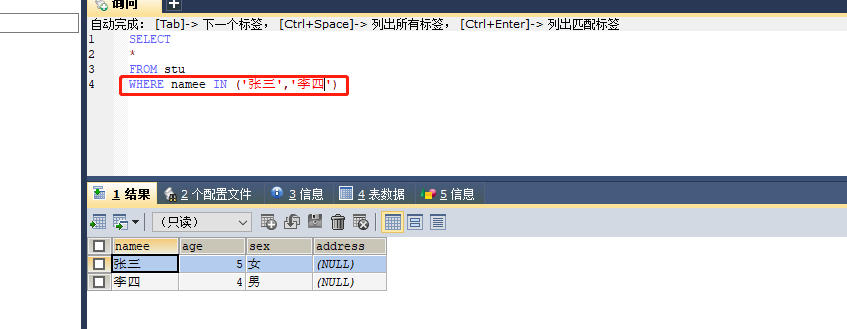
**说明**：查询满足条件1 且 满足条件2 或者 条件3的记录

(查询满足条件1 且 条件2 和 查询满足条件1 且 条件3的记录 )

* 1. IN、Between和LIKE

1)IN为相同字段的规定多个值

例如：查询namee为张三和李四的记录

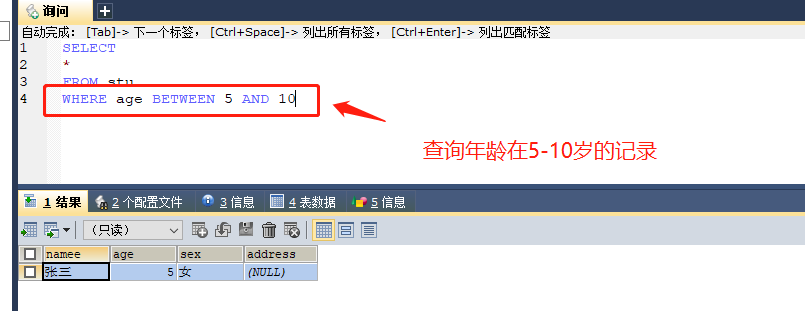


等同于：

Select \* from stu + where namee = ‘张三’ or name = ‘李四’

2)Between …… and ……在什么之间

例如：查询年龄在5—10岁的记录

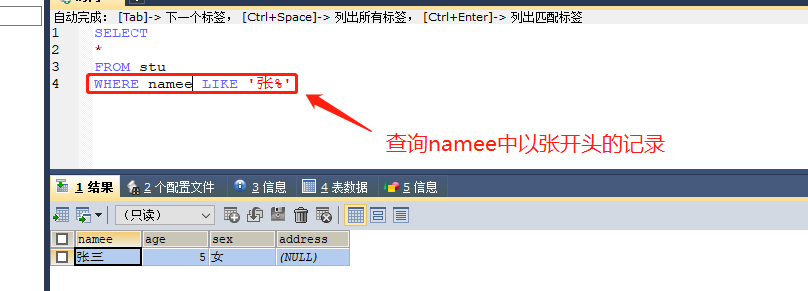


3)Like是模糊查询

A%：条件为指定字段中以‘A’开头的所有记录

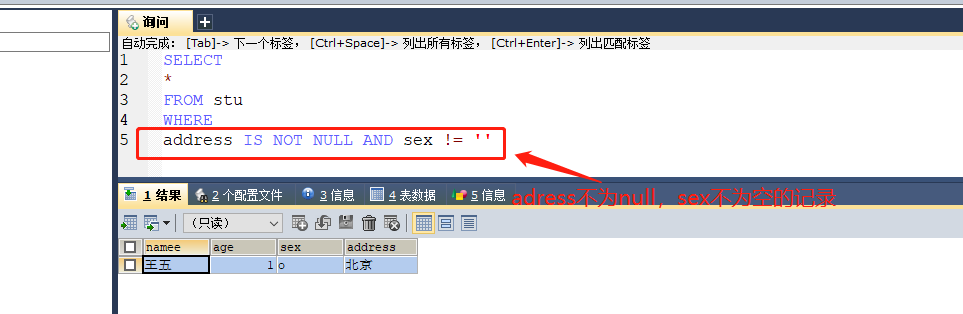
%A：条件为指定字段中以‘A’结尾的所有记录

%A%:条件为指定字段中含A的所有记录

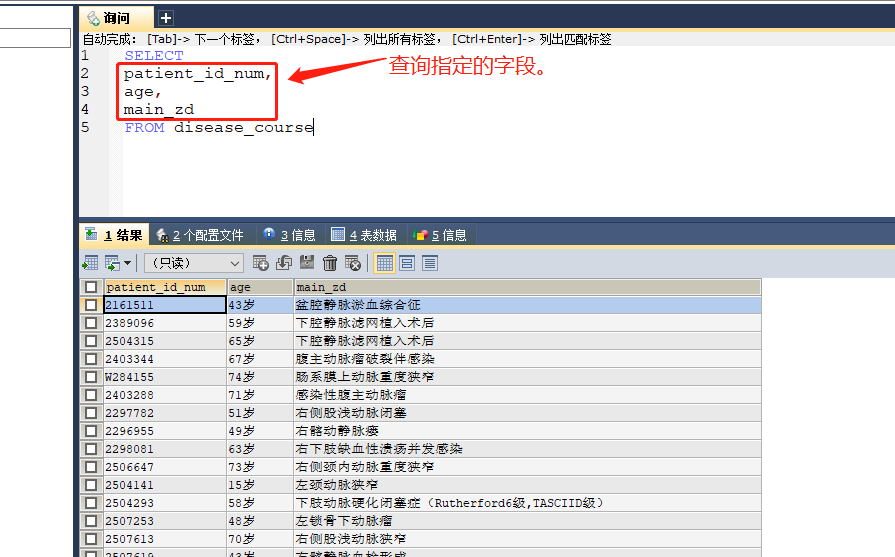


* 1. Where条件中不为空的表达方式

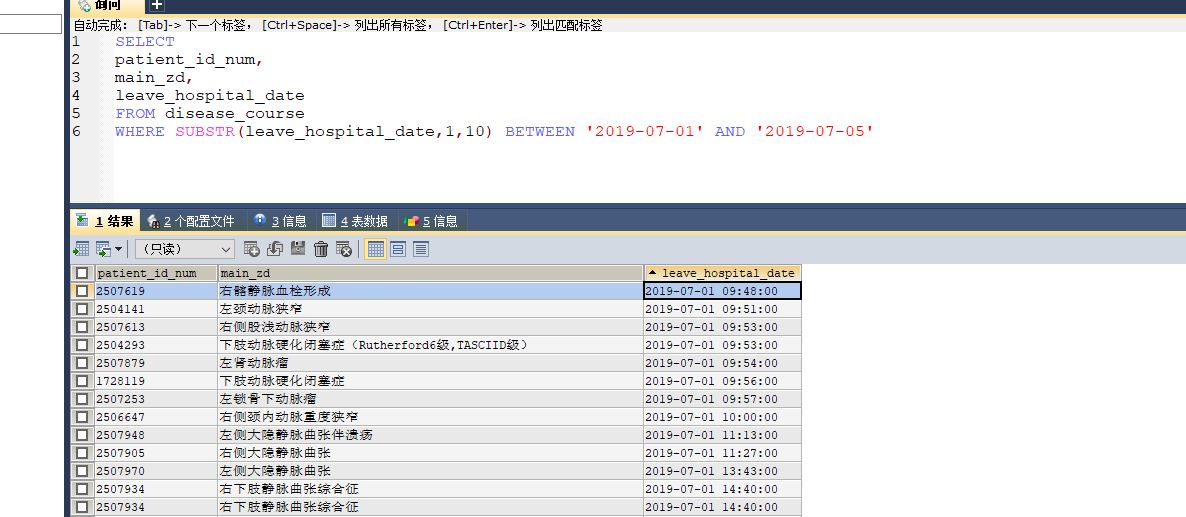
Select \* from stu where namee is not null and sex != ‘’



1. 复杂查询
   1. 查询disease\_course表中的病案号、年龄和主要诊断

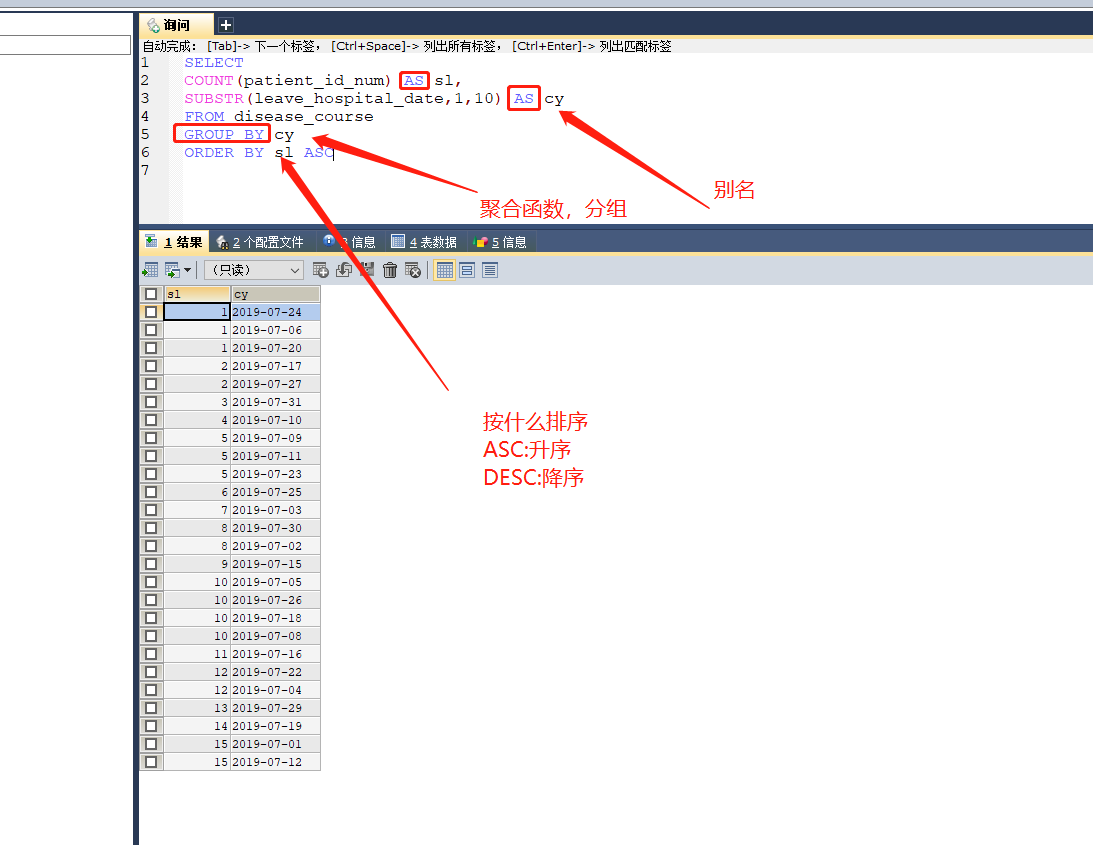


* 1. 查询disease\_course表中出院时间在2019-07-01和2019-07-05之间的病案号和主要诊断和出院时间。

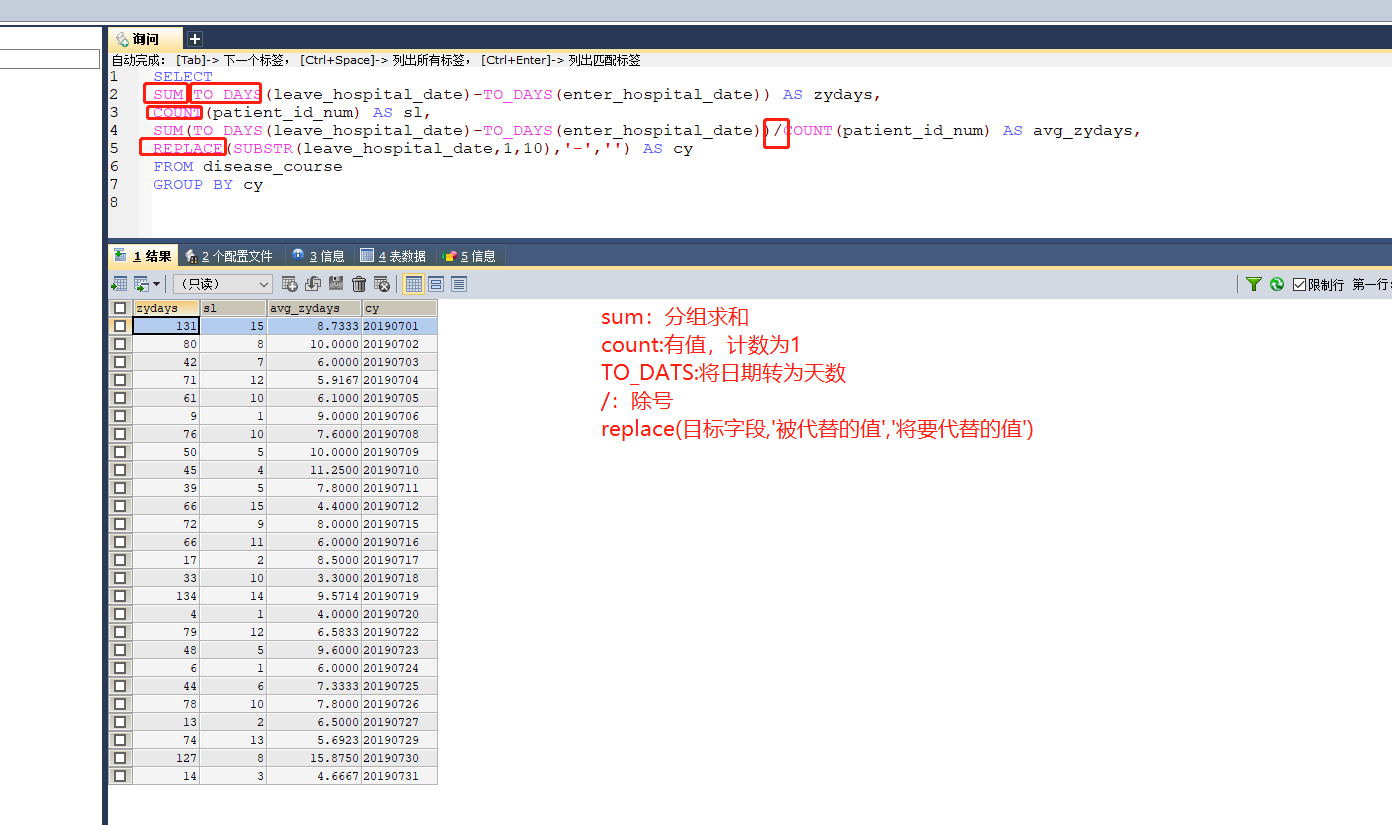


**Substr函数**是截取，格式：substr(目标字段，起始位置，截取的个数) 从1开始

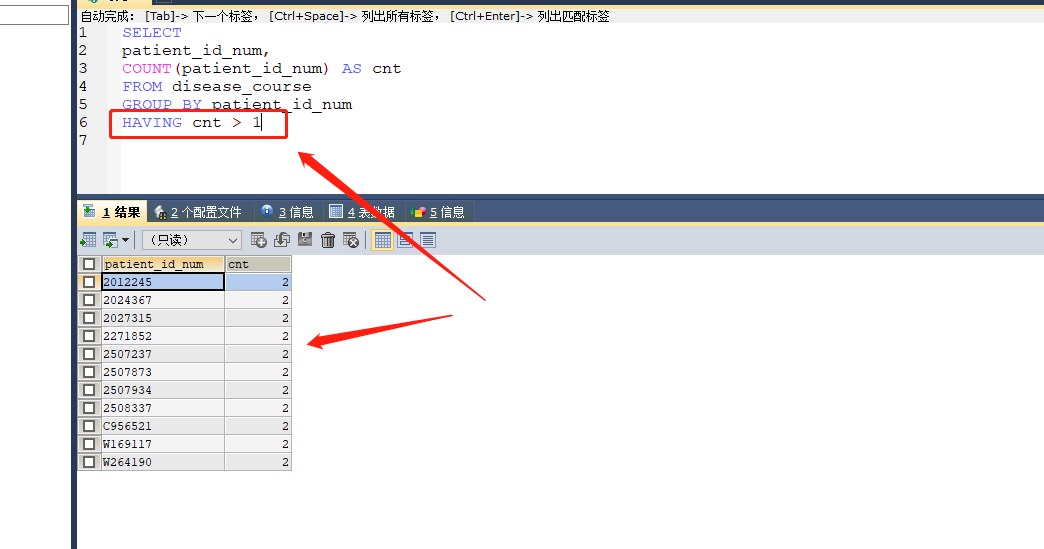
* 1. 查询disease\_course表中的2019年7月每天的患者数量并患者数按升序排列

④在disease\_course表中查询平均住院日(按天计算)

公式平均住院日(天)=出院时间的每天住院天数总和/每天的出院患者数



⑤在disease\_course表中查询病案号的重复值

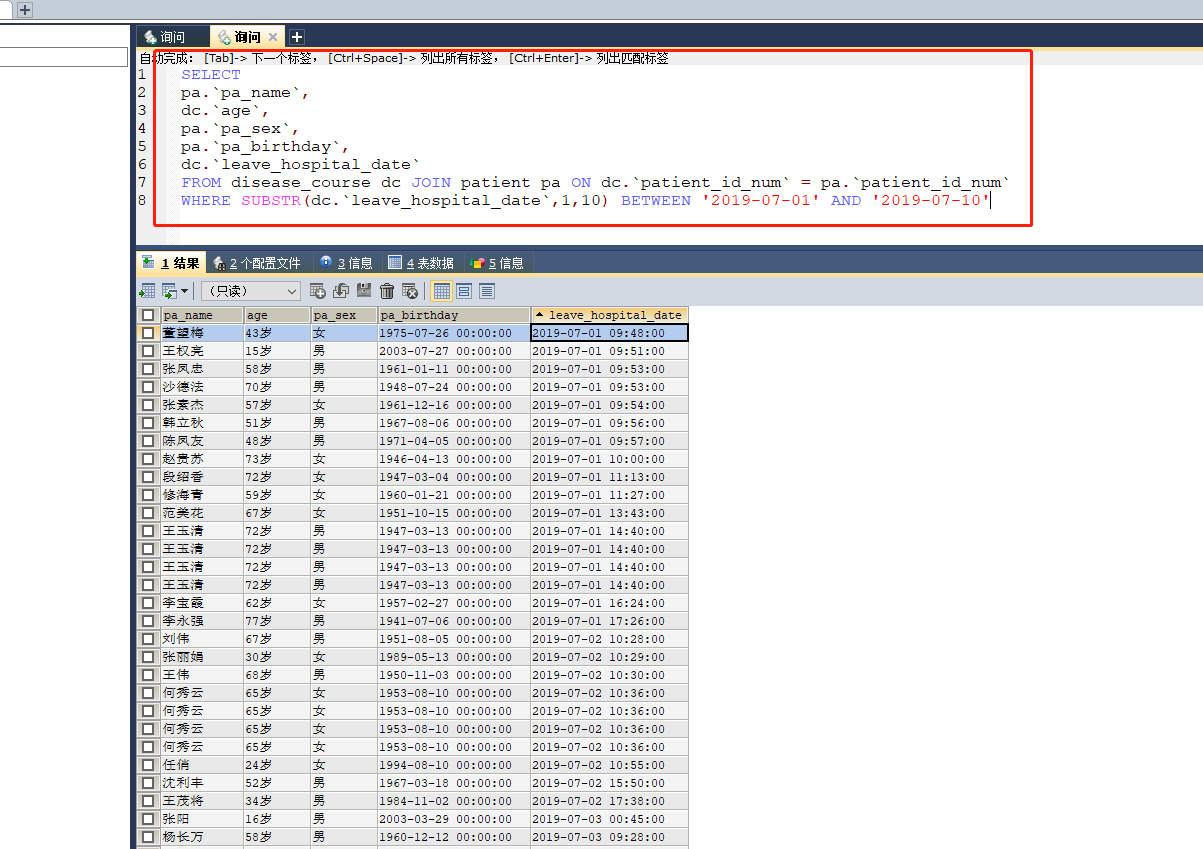


**HAVING：** 子句可以让我们筛选分组后的各组数据。通常与group by结合

⑥多表关联(join)

1）join(inner join)：两个表中的共同值条件时

例：查询患者的姓名,年龄,性别，出生日期，出院时间在2019-07-01至2019-07-10



2）LEFT JOIN(左连接) 关键字从左表（table1）返回所有的行，即使右表（table2）中没有匹配。如果右表中没有匹配，则结果为 NULL。(库里没有，这个写考试题的第4题)

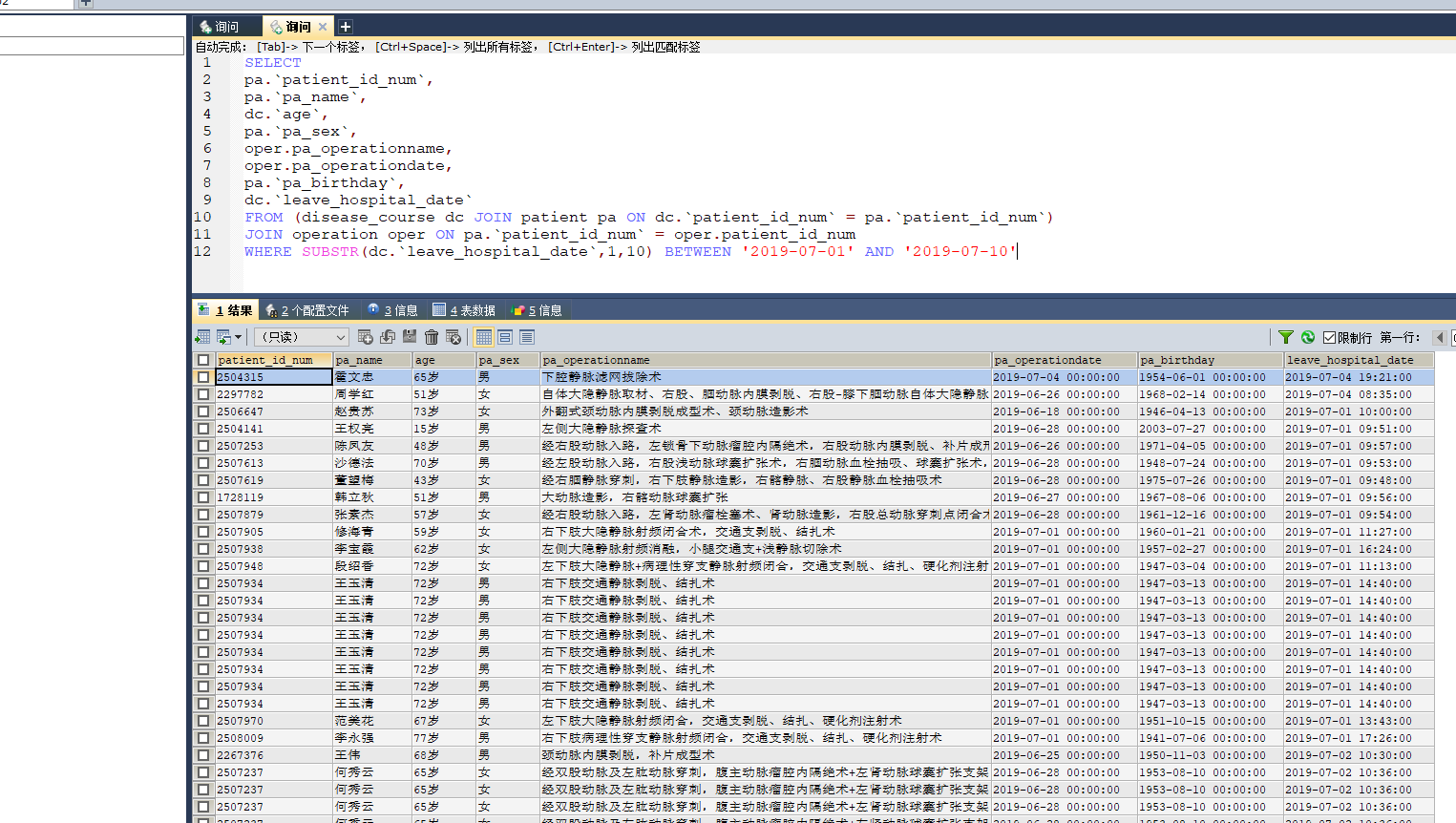


3) RIGHT JOIN(右关联) 关键字从右表（table2）返回所有的行，即使左表（table1）中没有匹配。如果左表中没有匹配，则结果为 NULL。(这个库里没有不用写)



4)三表关联

查询患者的病案号，姓名，年龄，性别，手术名称，手术时间，出生日期和出院时间，出院时间在2019-07-01至2019-07-10



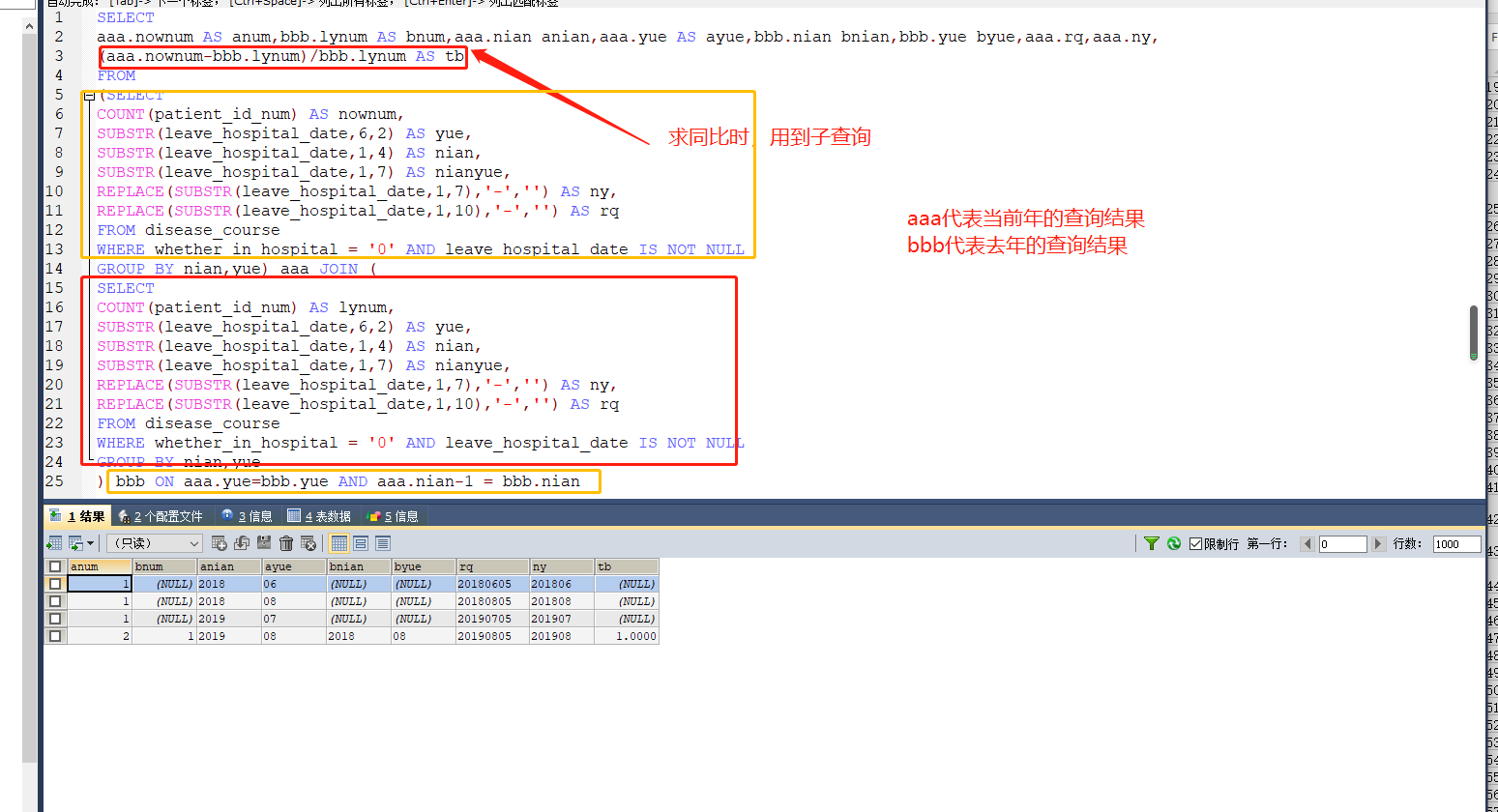
5) UNION 操作符用于合并两个或多个 SELECT 语句的结果集。

请注意，UNION 内部的每个 SELECT 语句必须拥有相同数量的列。列也必须拥有相似的数据类型。同时，每个 SELECT 语句中的列的顺序必须相同。

UNION ALL 可以允许有重复值



6)子查询(求同比)



7）Case函数

简单Case函数

CASE sex

WHEN '1' THEN '男'

WHEN '2' THEN '女'

ELSE '其他' END

--Case搜索函数

CASE WHEN sex = '1' THEN '男'

WHEN sex = '2' THEN '女'

ELSE '其他' END    
   这种方式，可以实现相同的功能。简单Case函数的写法相对比较简洁，但是和Case搜索函数相比，功能方面会有些限制，比如写判断式。还有一个需要注意的问题，Case函数只返回第一个符合条件的值，剩下的Case部分将会被自动忽略。

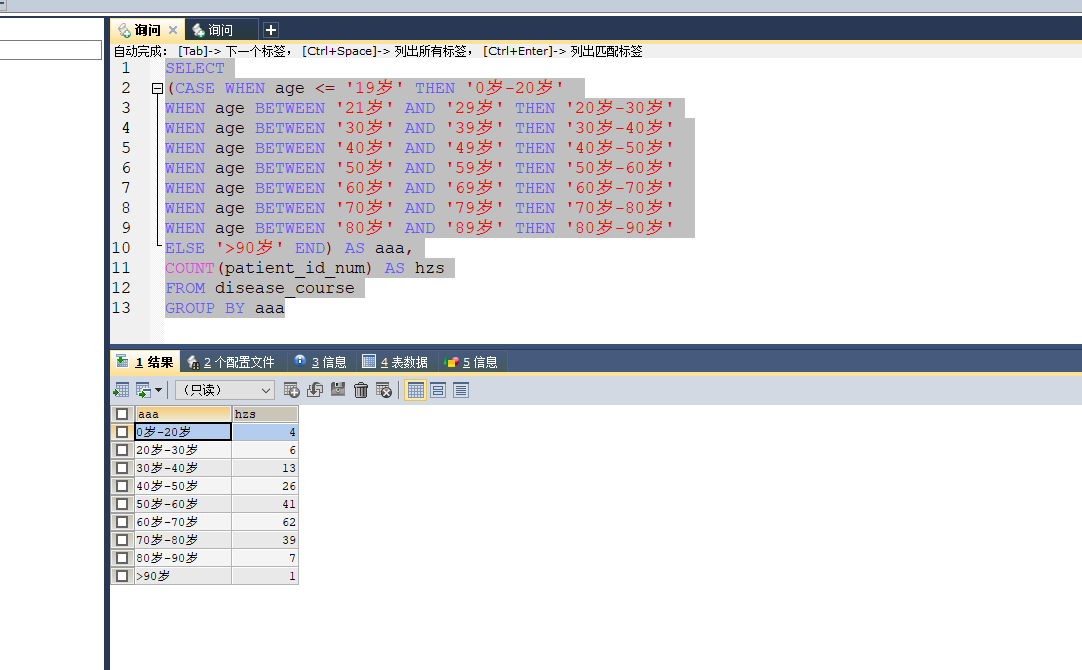
--比如说，下面这段SQL，你永远无法得到“第二类”这个结果

CASE WHEN col\_1 IN ( 'a', 'b') THEN '第一类'

WHEN col\_1 IN ('a')  THEN '第二类'

ELSE'其他' END

**例子：**



# 考核练习题：

**IP: 10.39.3.228 端口号：3306 账号密码：root**

**(1-6题)数据库：jiuqi表：course、score、student、teacher**

**(7,8题)数据库：ldh 表：disease\_course、operation、patient**

1.向student表中插入3个学生的信息，要求其中1个学生没有性别

2.查询姓王的学生名单

3.查询含李的学生名单

4. 查询所有学生的学号、姓名、选课数、总成绩

5.查询男生、女生的人数

6. 查询不及格的课程并按课程号从大到小排列

7.计算7月份每天的手术量

8. 在disease\_course表中查询术前平均住院日(按天计算)

手术时间-入院时间=术前住院时间，所有术前住院时间总和/人数(注：手术时间要对应上入院时间)