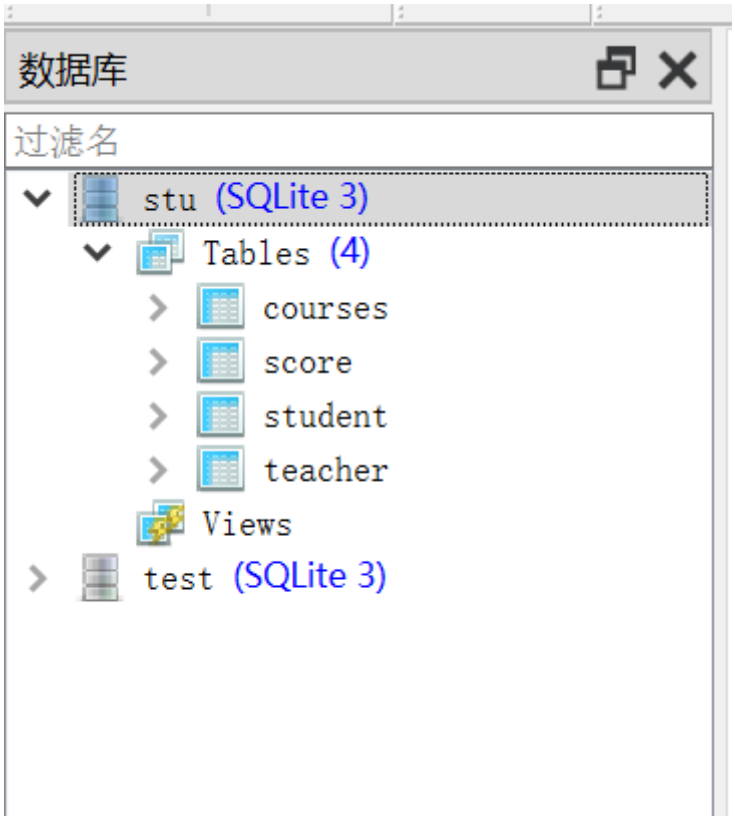
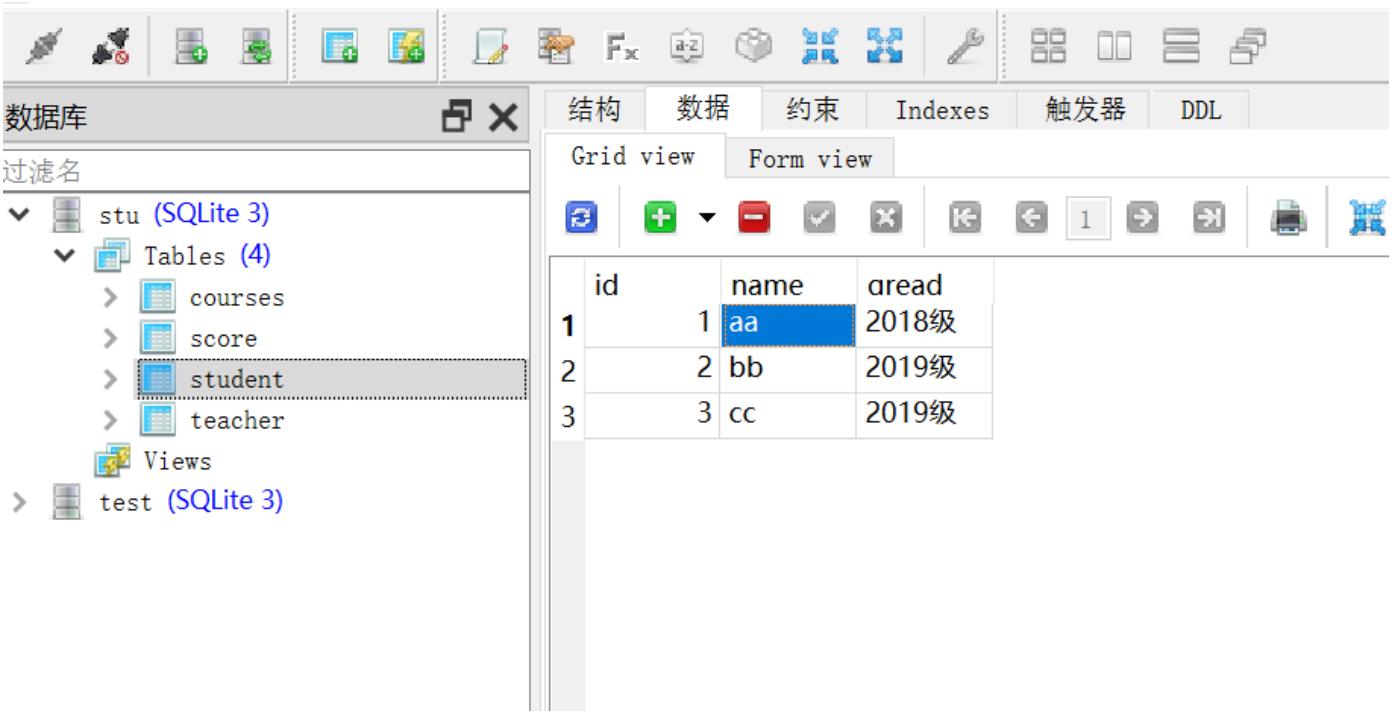
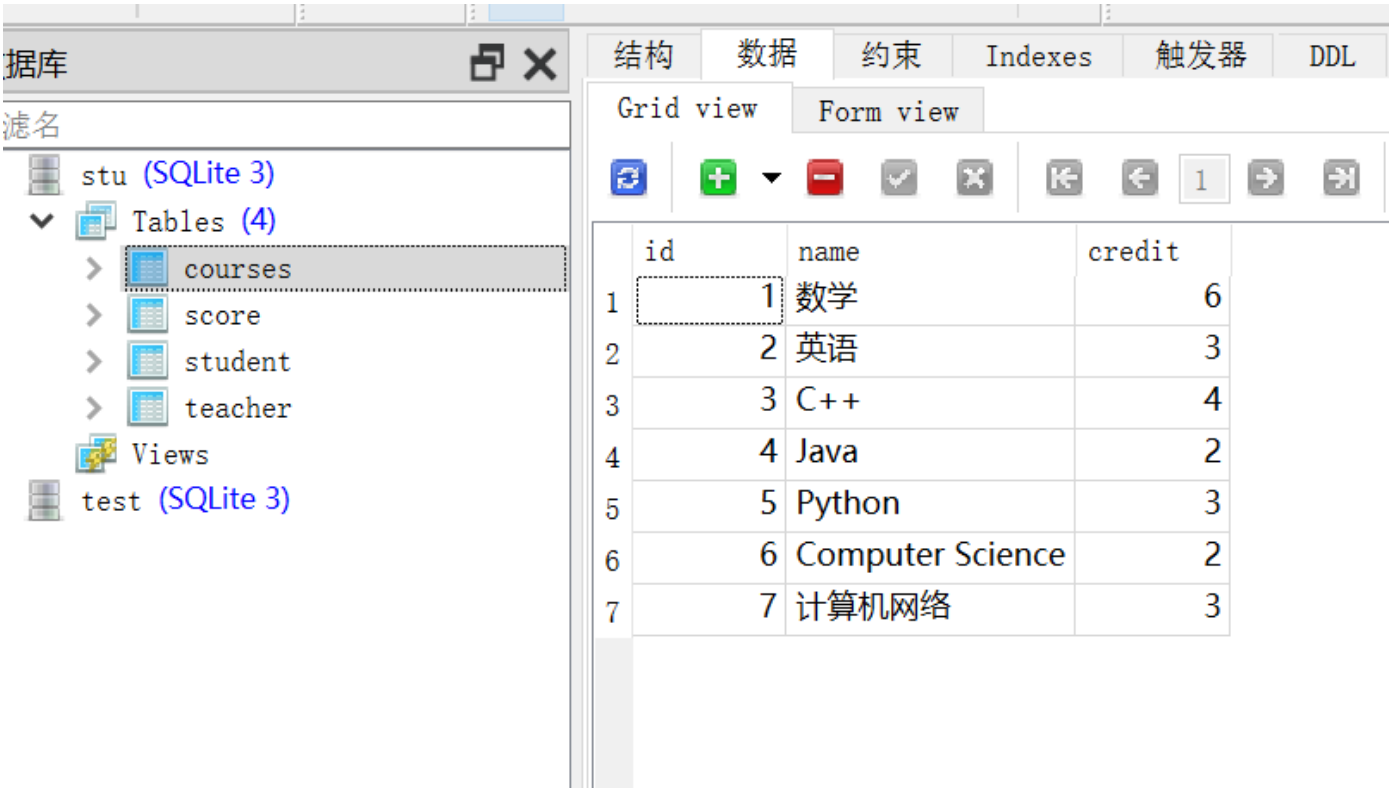
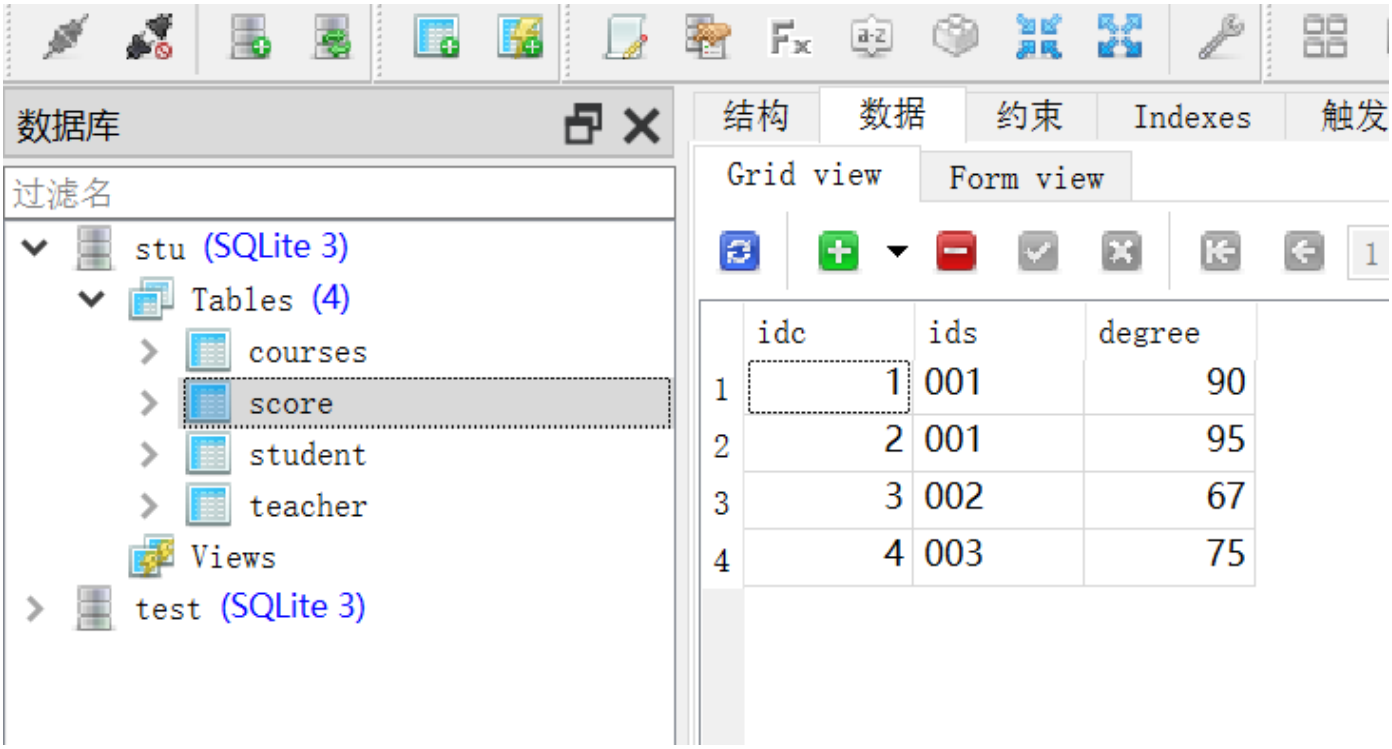
In [8]:

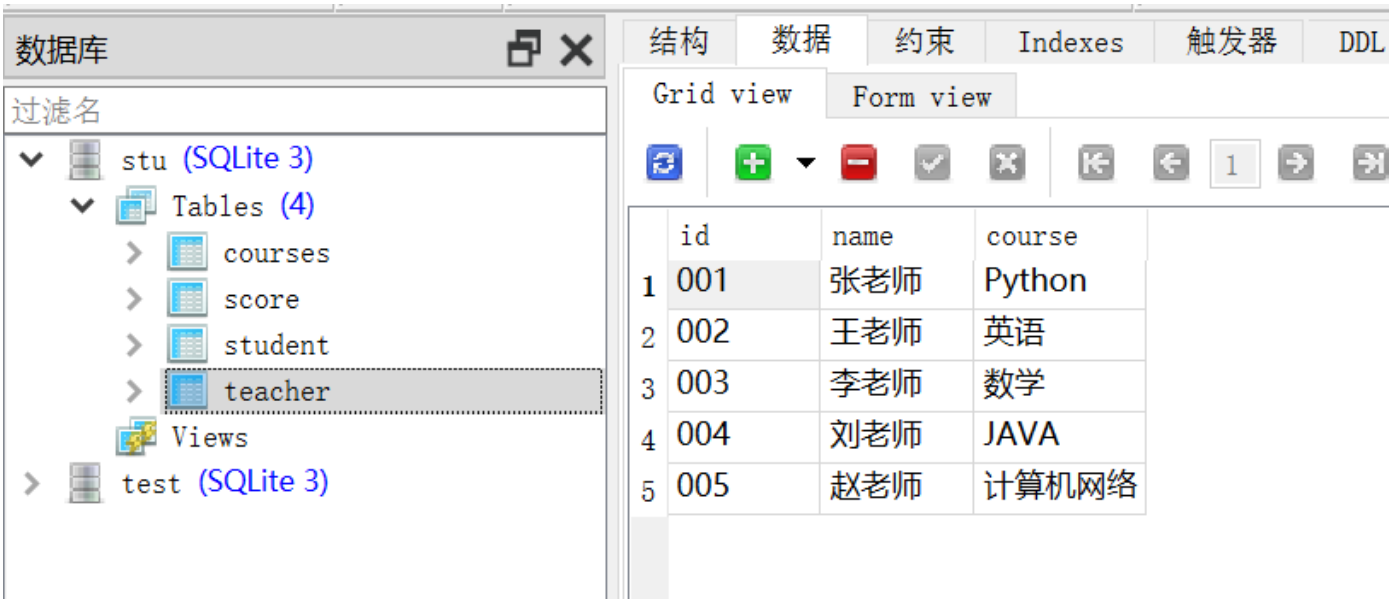
```
import sqlite3
conn = sqlite3.connect('stu.db')
cur = conn.cursor()
cur.execute("DROP TABLE IF EXISTS student")             #student
cur.execute("CREATE TABLE student(id TEXT, name TEXT, gread TEXT)")
cur.execute("insert into student values('001','aa','2018级')")
cur.execute("insert into student values('002','bb','2019级')")
cur.execute("insert into student values('003','cc','2019级')")

cur.execute("DROP TABLE IF EXISTS courses")            #course
cur.execute("CREATE TABLE courses(id INT, name TEXT, credit INT)")
courses = (
    (1, '数学', 6),
    (2, '英语', 3),
    (3, 'C++', 4),
    (4, 'Java', 2),
    (5, 'Python', 3),
    (6, 'Computer Science', 2),
    (7, '计算机网络', 3)
)
cur.executemany("INSERT INTO courses VALUES(?, ?, ?)", courses)

cur.execute("DROP TABLE IF EXISTS score")              #score
cur.execute("CREATE TABLE score(idc INT, ids TEXT, degree INT)")
degrees = (
    (1, '001', 90),
    (2, '001', 95),
    (3, '002', 67),
    (4, '003', 75),
)
cur.executemany("INSERT INTO score VALUES(?, ?, ?)", degrees)

cur.execute("DROP TABLE IF EXISTS teacher")             #teacher
cur.execute("CREATE TABLE teacher(id TEXT, name TEXT, course INT)")
teachers = (
    ('001', '张老师', 'Python'),
    ('002', '王老师','英语'),
    ('003', '李老师', '数学'),
    ('004', '刘老师', 'JAVA'),
    ('005', '赵老师', '计算机网络'),
)
cur.executemany("INSERT INTO teacher VALUES(?, ?, ?)", teachers)

conn.commit()
conn.close() # 关闭连接
```


数据库
过滤名
stu (SQLite 3)
Tables (4)
courses
score
student
teacher
Views
test (SQLite 3)


数据库
过滤名
stu (SQLite 3)
Tables (4)
courses
score
student
teacher
Views
test (SQLite 3)
结构
数据
约束
Indexes
触发器
DDL
Grid view
Form view
id
name
aread
1 1 aa 2018级
2 2 bb 2019级
3 3 cc 2019级


据库
滤名
stu (SQLite 3)
Tables (4)
courses
score
student
teacher
Views
test (SQLite 3)
结构
数据
约束
Indexes
触发器
DDL
Grid view
Form view
1
id name credit
1 1 数学 6
2 2 英语 3
3 3 C++ 4
4 4 Java 2
5 5 Python 3
6 6 Computer Science 2
7 7 计算机网络 3


数据库
过滤名
stu (SQLite 3)
Tables (4)
courses
score
student
teacher
Views
test (SQLite 3)
结构
数据
约束
Indexes
触发
Grid view
Form view
1
idc ids degree
1 1 001 90
2 2 001 95
3 3 002 67
4 4 003 75


数据库
过滤名
stu (SQLite 3)
Tables (4)
courses
score
student
teacher
Views
test (SQLite 3)
结构
数据
约束
Indexes
触发器
DDL
Grid view
Form view
1
id name course
1 001 张老师 Python
2 002 王老师 英语
3 003 李老师 数学
4 004 刘老师 JAVA
5 005 赵老师 计算机网络

In [38]:

```
import MySQLdb
DB_name = "stu"
stu_lst = (('0001', 'Tom', 20),
           ('0002', 'Lucy',19),
           ('0003', 'Lily',21),
           ('0004', 'Liu',20),
           ('0005', 'Mary',19))

cour_lst=(('001', 'JAVA', 20,3),
           ('002', 'Python',30,4),
           ('003', 'ENGLISH',60,5),
           ('004', 'CPP',18,2),
           ('005', 'OS',48,5),
           ('006', 'MATH',24,3))
sco_lst=(('0001', '001', 88),
           ('0002', '003',90),
           ('0003', '004',71),
           ('0004', '002',85),
           ('0004', '003',79),
           ('0005', '006',92))
tea_lst=(('1001', 'wang', 'IS'),
           ('1002', 'liu', 'CS'),
           ('1003', 'zhao','MA'),
           ('1004', 'yang','CS'),
           ('1005', 'zhu','CS'),
           ('1006', 'zhang','MA'),
           ('1007', 'hu','IS'))
job_lst=(('1001', '001', '2020'),
           ('1002', '003', '2019'),
           ('1003', '004','2019'),
           ('1004', '002','2020'),
           ('1005', '001','2018'),
           ('1006', '006','2019'),
           ('1007', '005','2020'))
try:
    # 打开数据库连接
    db = MySQLdb.connect(host="localhost", user="root",
                         passwd="123456", port=3306)

    cur = db.cursor()  # 使用cursor()方法获取操作游标
    cur.execute("create database if not exists stu")     # 创建数据库
    db.select_db(DB_name)  # 选择数据库

# 创建表student
    cur.execute("SET FOREIGN_KEY_CHECKS=0")
    cur.execute("drop table if exists student")
    cur.execute("SET FOREIGN_KEY_CHECKS=1")
    cur.execute("create table student (sid varchar(16) primary key,\
            sname varchar(16),\
            grade int)ENGINE=innoDB DEFAULT CHARSET=utf8")
    cur.executemany('insert into student values(%s, %s, %s)',stu_lst)
    db.commit()    # 提交修改,写入数据库
# 查询数据
    print("student表: ")
    cur.execute( "select * from student")
    print(cur.fetchall())

# 创建表course
    cur.execute("SET FOREIGN_KEY_CHECKS=0")
```

```
    cur.execute("drop table if exists course")
    cur.execute("SET FOREIGN_KEY_CHECKS=1")
    cur.execute("create table course (cid varchar(16) primary key,\
            cname varchar(16),\
            hours int,\
            credit int)ENGINE=innoDB DEFAULT CHARSET=utf8")
    cur.executemany('insert into course values(%s, %s, %s, %s)',cour_lst)
    db.commit()    # 提交修改,写入数据库
    # 查询数据
    print("course表：")
    cur.execute( "select * from course")
    print(cur.fetchall())

# 创建表score
    cur.execute("drop table if exists score")
    cur.execute("create table score (sid varchar(16),\
            cid varchar(16),\
            score int,\
            primary key(sid,cid),\
            FOREIGN KEY(sid) REFERENCES student(sid),\
            FOREIGN KEY(cid) REFERENCES course(cid)   )ENGINE=innoDB DEFAULT CHARSET=utf8")
    cur.executemany('insert into score values(%s, %s, %s)',sco_lst)
    db.commit()    # 提交修改,写入数据库
    # 查询数据
    print("score表：")
    cur.execute( "select * from score")
    print(cur.fetchall())

# 创建表teacher
    cur.execute("SET FOREIGN_KEY_CHECKS=0")
    cur.execute("drop table if exists teacher")
    cur.execute("SET FOREIGN_KEY_CHECKS=1")
    cur.execute("create table teacher (tid varchar(16) primary key,\
            tname varchar(16),\
            tmajor varchar(16))ENGINE=innoDB DEFAULT CHARSET=utf8")
    cur.executemany('insert into teacher values(%s, %s, %s)',tea_lst)
    db.commit()    # 提交修改,写入数据库
# 查询数据
    print("teacher表：")
    cur.execute( "select * from teacher")
    print(cur.fetchall())

# 创建表job
    cur.execute("drop table if exists job")
    cur.execute("create table job (tid varchar(16) ,\
            cid varchar(16),\
            semester varchar(16),\
            FOREIGN KEY(tid) REFERENCES teacher(tid),\
            FOREIGN KEY(cid) REFERENCES course(cid)   )ENGINE=innoDB DEFAULT CHARSET=utf8")
    cur.executemany('insert into job values(%s, %s, %s)',job_lst)
    db.commit()    # 提交修改,写入数据库
# 查询数据
    print("job表：")
    cur.execute( "select * from job")
    print(cur.fetchall())

#插入数据
    print("------------插入数据--------------")
    table_name=input("请输入要向哪个表插入数据：")
    if table_name=='student':
        sid_insert=input("请输入学生学号：")
```

```
        sname_insert=input("请输入学生姓名：")
        grade_insert=input("请输入学生年级：")
        insert_sql='insert into student values({x}, {y}, {z})'.format(x = sid_insert,y=sname_insert,
        cur.execute(insert_sql)
        db.commit()
    print("student表：")
    cur.execute( "select * from student")
    print(cur.fetchall())

#查询信息
    print("-------------查询信息---------------")
    id_search=input("请输入要查询的学生的id：")
    select_sql = "select * from student where sid={x}".format(x = id_search)
    cur.execute(select_sql)
    print("该学生信息为：")
    print(cur.fetchall())

# 修改数据
    print("-------------修改数据---------------")
    sid_insert=input("请输入学生学号：")
    sname_update=input("请输入学生姓名：")
    update_sql = 'update student set sname={x} where sid={y}'.format(x = sname_update,y=sid_insert)
    cur.execute(update_sql)
    print("student表：")
    cur.execute( "select * from student")
    print(cur.fetchall())

#删除信息
    print("-------------删除信息---------------")
    sid_del=input("请输入学生学号：")
    drop_sql = 'delete from student where sid={x}'.format(x = sid_del)
    cur.execute(drop_sql)
    print("student表：")
    cur.execute( "select * from student")
    print(cur.fetchall())
finally:
    if db:
        db.close()
```

```
student表：
(('0001', 'Tom', 20), ('0002', 'Lucy', 19), ('0003', 'Lily', 21), ('0004', 'Liu',
20), ('0005', 'Mary', 19))
course表：
(('001', 'JAVA', 20, 3), ('002', 'Python', 30, 4), ('003', 'ENGLISH', 60, 5), ('0
04', 'CPP', 18, 2), ('005', 'OS', 48, 5), ('006', 'MATH', 24, 3))
score表：
(('0001', '001', 88), ('0002', '003', 90), ('0003', '004', 71), ('0004', '002', 8
5), ('0004', '003', 79), ('0005', '006', 92))
teacher表：
(('1001', 'wang', 'IS'), ('1002', 'liu', 'CS'), ('1003', 'zhao', 'MA'), ('1004',
'yang', 'CS'), ('1005', 'zhu', 'CS'), ('1006', 'zhang', 'MA'), ('1007', 'hu', 'I
S'))
job表：
(('1001', '001', '2020'), ('1002', '003', '2019'), ('1003', '004', '2019'), ('100
4', '002', '2020'), ('1005', '001', '2018'), ('1006', '006', '2019'), ('1007', '0
05', '2020'))
-------------插入数据---------------
请输入要向哪个表插入数据：student
请输入学生学号：'0006'
请输入学生姓名：'ABCDE'
```

```
请输入学生年级：20
student表：
(('0001', 'Tom', 20), ('0002', 'Lucy', 19), ('0003', 'Lily', 21), ('0004', 'Liu',
20), ('0005', 'Mary', 19), ('0006', 'ABCDE', 20))
------------查询信息--------------
请输入要查询的学生的id：'0005'
该学生信息为：
(('0005', 'Mary', 19),)
------------修改数据--------------
请输入学生学号：'0002'
请输入学生姓名：'NEW'
student表：
(('0001', 'Tom', 20), ('0002', 'NEW', 19), ('0003', 'Lily', 21), ('0004', 'Liu',
20), ('0005', 'Mary', 19), ('0006', 'ABCDE', 20))
------------删除信息--------------
请输入学生学号：'0006'
student表：
(('0001', 'Tom', 20), ('0002', 'NEW', 19), ('0003', 'Lily', 21), ('0004', 'Liu',
20), ('0005', 'Mary', 19))
```

In [ ]: