

杭州电子科技大学

数据库原理
实 验 报 告

学 院	_____ 网络空间安全学院
专 业	_____
班 级	_____
学 号	_____
学生姓名	_____
教师姓名	_____
完成日期	_____
成 绩	

实验二 T-SQL 语句操作（1）

一、 实验目的、

熟练掌握 T-SQL 语句的使用

二、 实验内容

- （1）利用 T-SQL 定义与维护数据表结构
- （2）利用 T-SQL 创建与管理数据表的索引
- （3）利用 T-SQL 维护数据表记录内容
- （4）SELECT 语句的数据查询功能

三、 实验环境

BIOS 信息：

SMBIOSBIOSVersion : 1.19.1
Manufacturer : Dell Inc.
Name : 1.19.1
SerialNumber : C5NY4N3
Version : DELL-2

操作系统版本信息：

BuildNumber : 19045
BuildType : Multiprocessor Free
OSType : 18
ServicePackMajorVersion : 0
ServicePackMinorVersion : 0

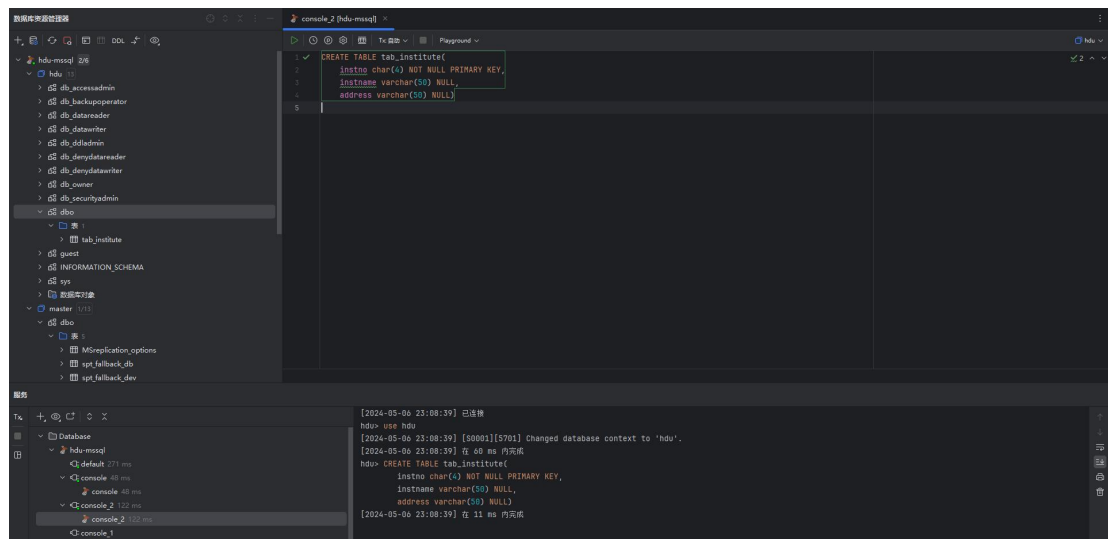
使用软件信息：

Docker+MySQL+DataGrip

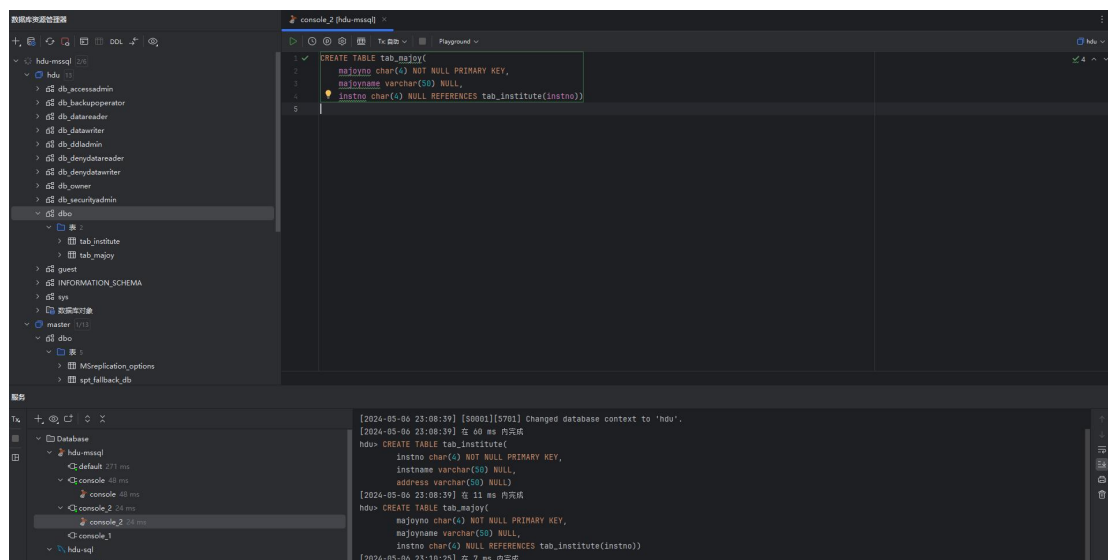
四、 主要操作步骤及实验结果记录

- （一）使用 T-SQL 定义和维护数据表结构
- 1、利用 CREATE TABLE 语句定义数据表结构

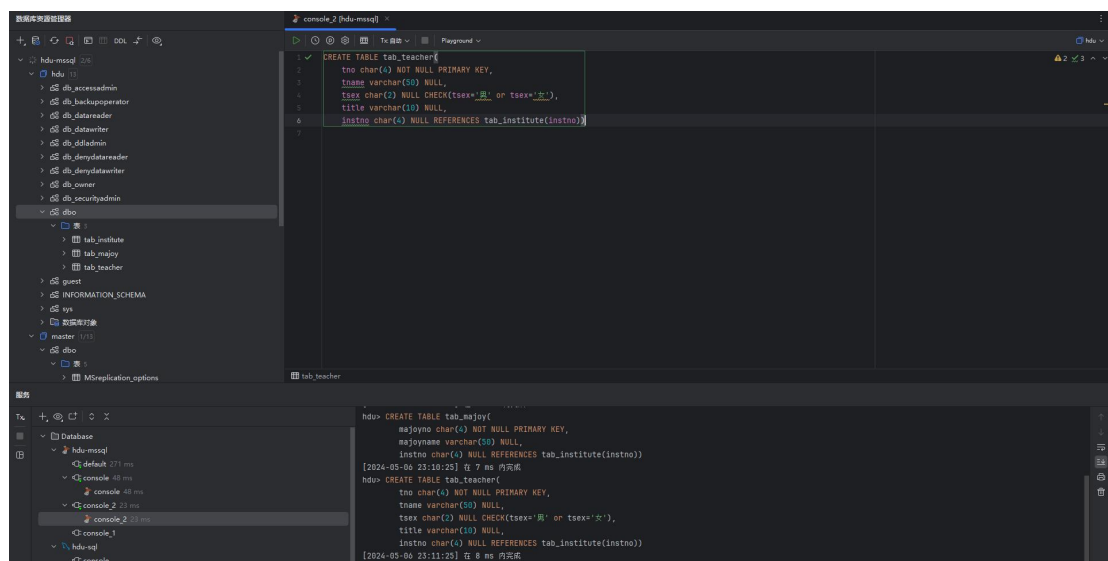
定义学院表 tab_institute



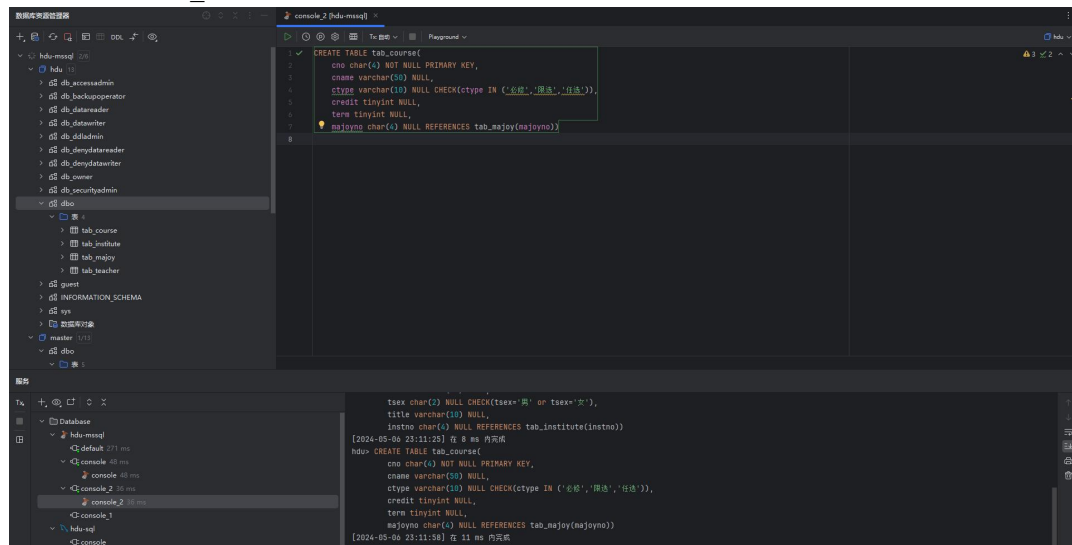
定义专业表 tab_major



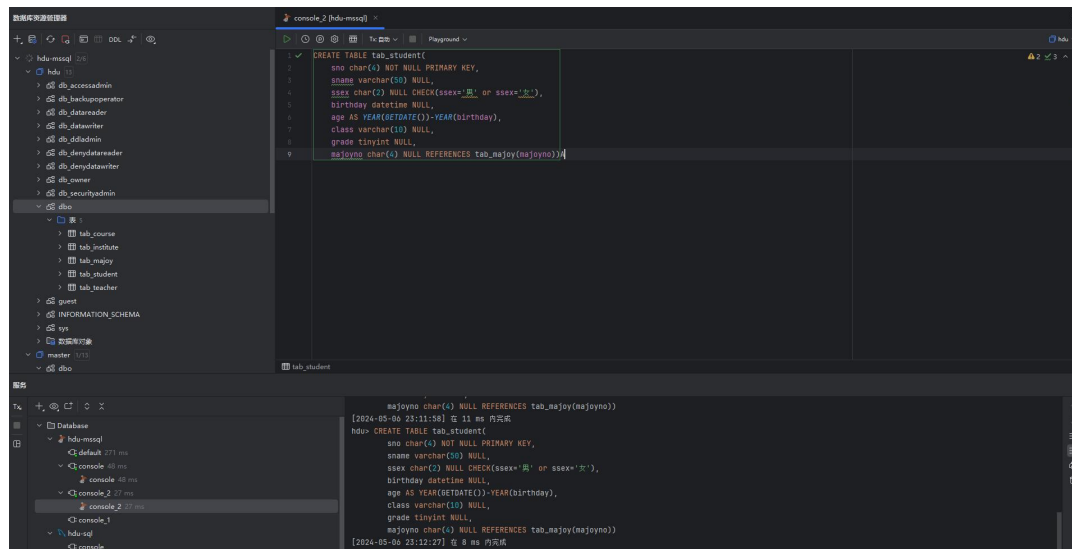
定义教师表 tab_teacher



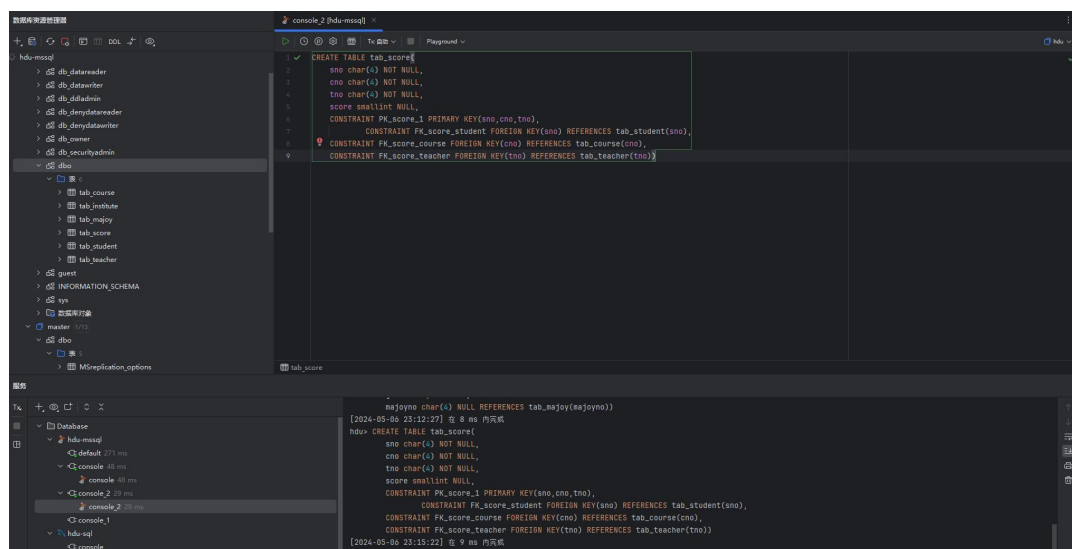
定义课程表 tab_courses



定义学生表 tab_students

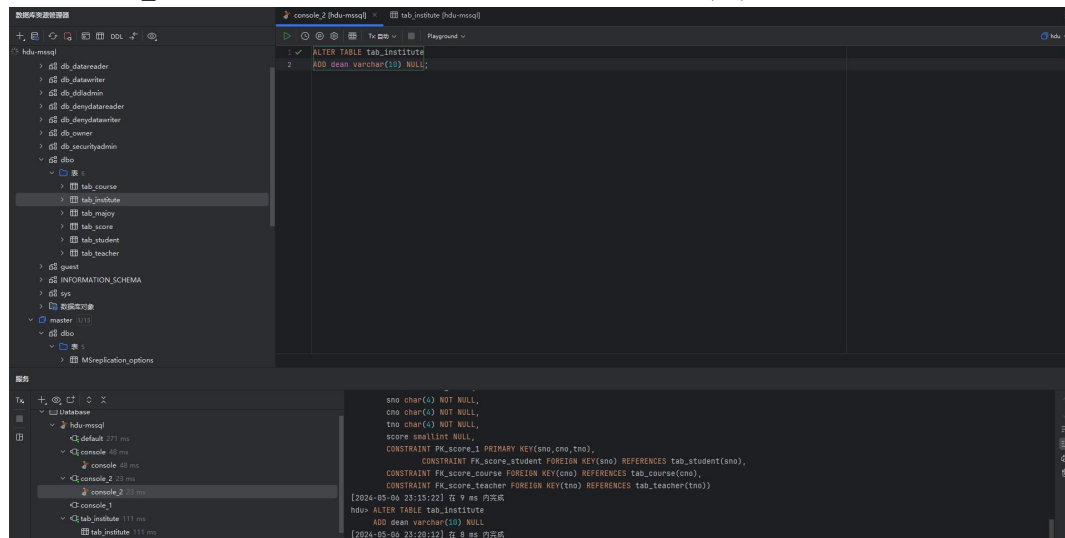


定义选修（成绩）表 tab_score

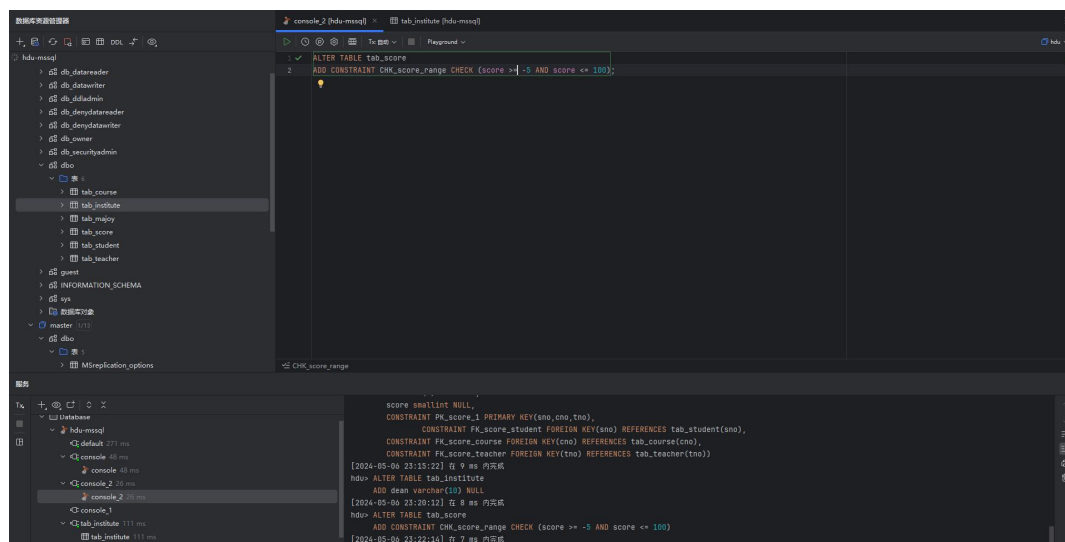


2、利用 ALTER TABLE 语句修改数据表结构

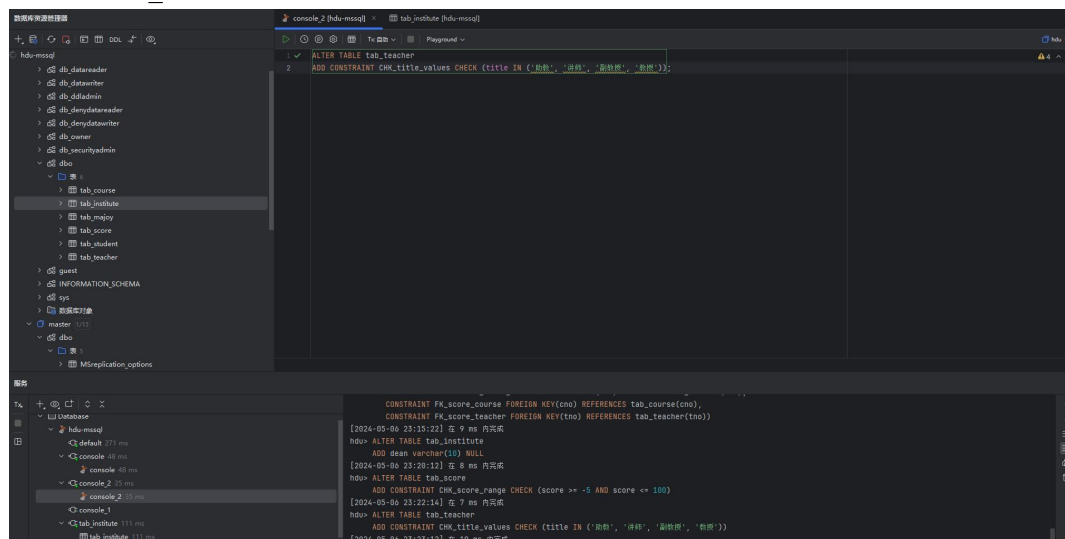
(1) 在 tab_institute 表中加一个“院长”字段 (dean,varchar(10),NULL)。



(2) 在 tab_score 表中对成绩字段 score 加一个 CHECK 约束 (-5~100)，其中-1~5 分别代表五级制成绩“优秀、良好、正等、及格、不及格”，0~100 代表百分制成绩

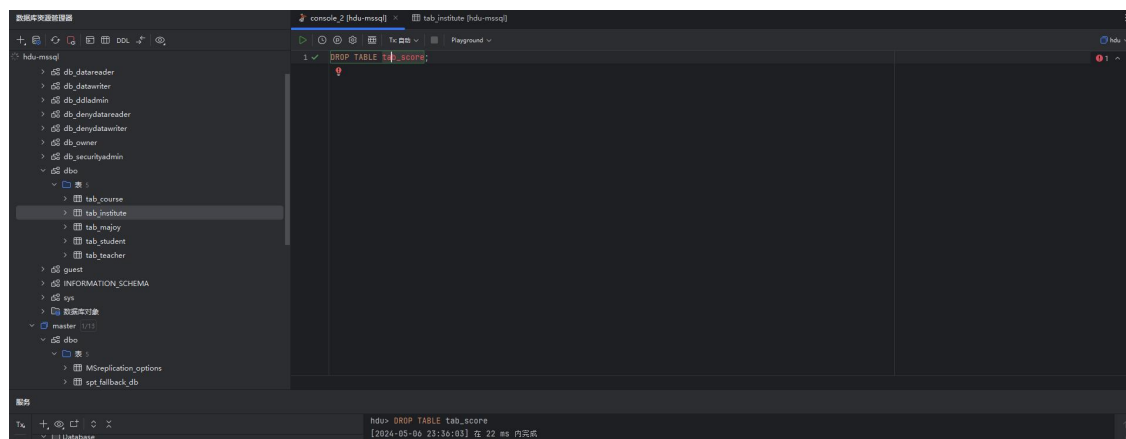


(3) 在 tab_teacher 表中对职称字段 title 加一个 CHECK 约束（助教，讲师，副教授，教授）。



3、利用 DROP TABLE 语句删除数据表结构

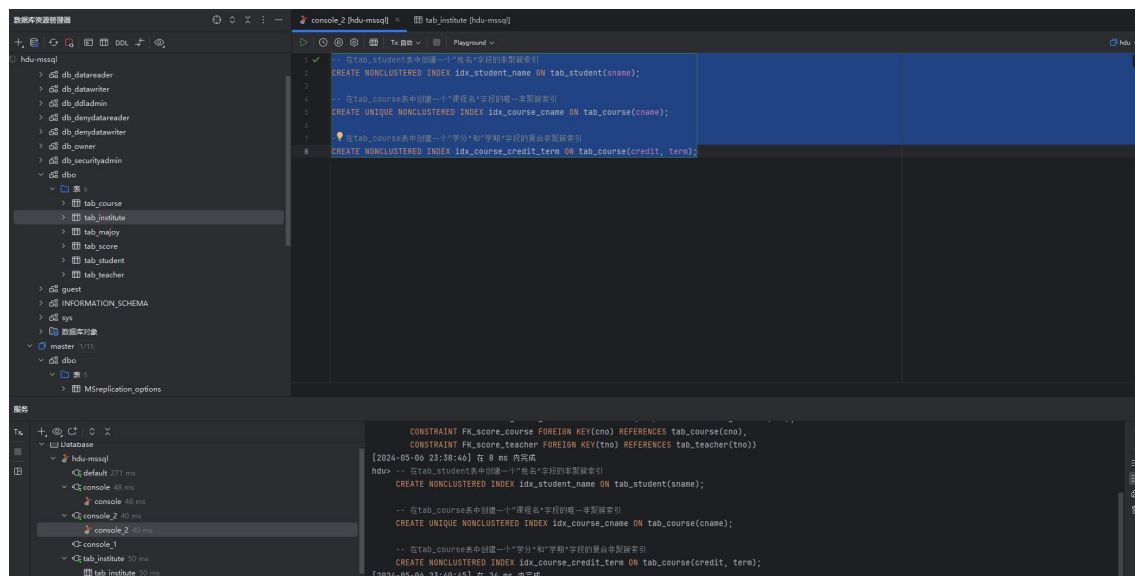
这里以删除 tab_score 表为例（之后又重新创建）：



4、利用 CREATE INDEX 语句创建索引

- (1) 在 tab_student 表中创建一个“姓名”字段的非聚簇索引。
- (2) 在 tab_course 表中创建一个“课程名”字段的唯一非聚簇索引。
- (3) 在 tab_course 表中创建一个“学分”和“学期”字段的复合非聚簇索引。

将以上三条要求对应的 T-SQL 语句一起运行：



5、利用 DROP INDEX 语句删除索引

这里删除“在 tab_student 表中创建一个‘姓名’字段的非聚簇索引”。

(二) 利用 T-SQL 维护数据表记录内容

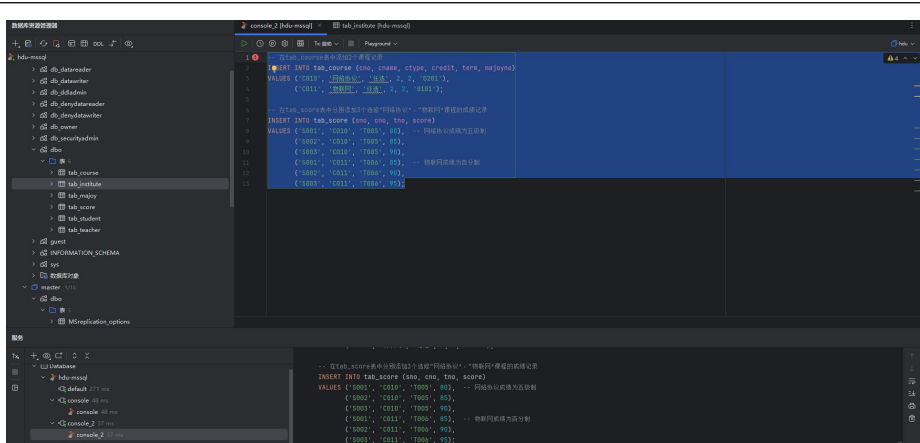
1、利用 INSERT 语句添加记录

- (1) 在 tab_course 表中添加 2 个课程记录，记录内容如下：

C010, 网络协议, 任选, 2, 2, 0201

C011, 物联网, 任选, 2, 2, 0101

- (2) 在 tab_score 表中分别添加 3 个选修“网络协议”、“物联网”课程的成绩记录（学生可跨专业，任课教师分别为 T005 和 T006，“网络协议”成绩为五级制，“物联网”成绩为百分制）。



2、利用 UPDATE 语句修改记录内容、

(1) 将 tab_score 表中 “网络协议” 成绩改成百分制，转换规则如下：
优秀-95，良好-85，正等-75，及格-65，不及格-55

如：UPDATE tab_score SET score=95 where score= -1 AND cno= ‘C010’

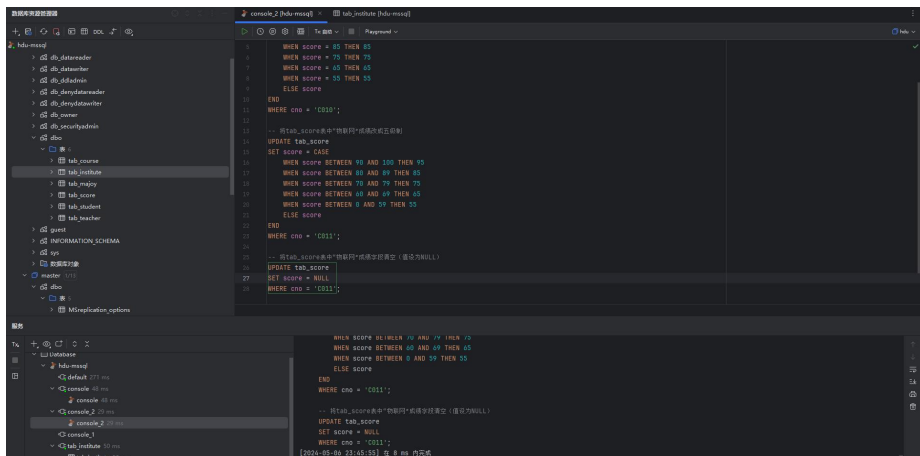
(2) 将 tab_score 表中 “物联网” 成绩改成五级制，转换规则如下：

优秀：100-90，良好：89-80，正等：79-70，及格：69-60，不及格：59-0

如：UPDATE tab_score SET score=-1

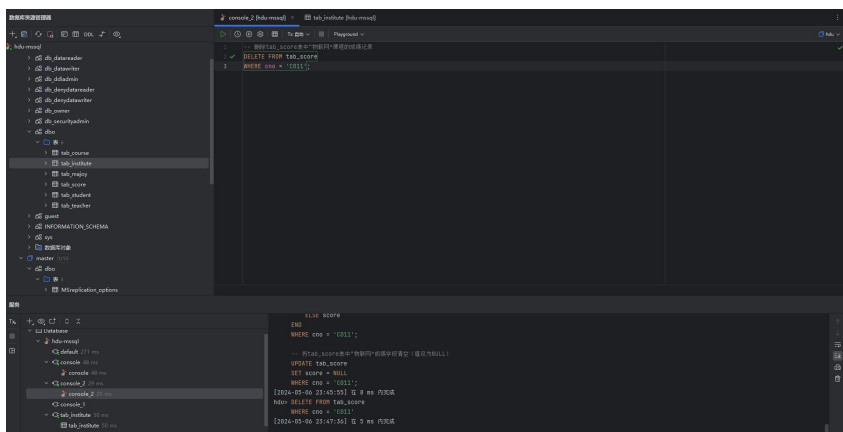
WHERE score BETWEEN 90 AND 100 AND cno= ‘C011’

(3) 将 tab_score 表中 “物联网” 成绩字段清空（值 NULL）。



3、利用 DELETE 语句删除记录

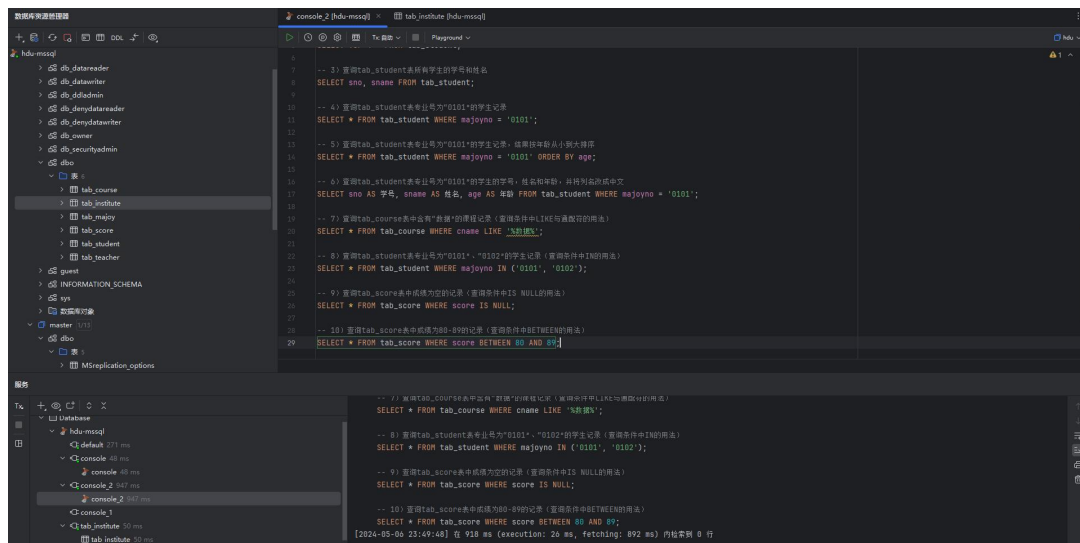
(1) 将 tab_score 表中 “物联网” 成绩记录删除。



（三）SELECT 语句的数据查询功能

1、单表查询——查询内容和查询条件都在同一个表中

- 1) 查询 tab_student 表所有学生的记录
- 2) 查询 tab_student 表中前 4 个学生的记录
- 3) 查询 tab_student 表所有学生的学号和姓名
- 4) 查询 tab_student 表专业号为“0101”的学生记录
- 5) 查询 tab_student 表专业号为“0101”的学生记录，结果按年龄从小到大排序。
- 6) 查询 tab_student 表专业号为“0101”的学生的学号，姓名和年龄，并将列名改成中文。
- 7) 查询 tab_couse 表中含有“数据”的课程记录。（查询条件中 LIKE 与通配符的用法）
- 8) 查询 tab_student 表专业号为“0101”、“0102”的学生记录（查询条件中 IN 的用法）
- 9) 查询 tab_score 表中成绩为空的记录（查询条件中 IS NULL 的用法）
- 10) 查询 tab_score 表中成绩为 80-89 的记录（查询条件中 BETWEEN 的用法）



2、连接查询的操作练习

- 1) 查询全校的教师的工号、姓名、所在学院名
- 2) 查询授课教师的工号、姓名、讲授课程名
- 3) 查询“数据结构”课程的学生的学号、姓名、课程名、成绩、任课教师名
- 4) 查询“计算机学院”所有学生的学号、姓名、课程名、成绩、任课教师名
- 5) 查询所有授课的“教授”的工号、姓名和授课课程名



3、嵌套查询的操作练习

```
sql

-- 1) 查询“计算机学院”的教师记录
SELECT * FROM 教师表 WHERE 学院名 = '计算机学院';

-- 2) 查询“计算机学院”的课程记录
SELECT * FROM 课程表 WHERE 学院名 = '计算机学院';

-- 3) 查询讲授“数据库”课程的教师记录
SELECT * FROM 教师表 WHERE 课程名 = '数据库';

-- 4) 查询选修“数据结构”课程的学生记录
SELECT * FROM 学生表 WHERE 课程名 = '数据结构';

-- 5) 查询“C001”课程成绩第一名的学生的学号与姓名
SELECT 学号, 姓名 FROM 成绩表 WHERE 课程号 = 'C001' ORDER BY 成绩 DESC LIMIT 1;

-- 6) 查询“C001”课程成绩高于“李四”的学生的学号与成绩
SELECT 学号, 成绩 FROM 成绩表 WHERE 课程号 = 'C001' AND 成绩 > (SELECT 成绩 FROM 学生表 WHERE 姓名 = '李四');

-- 7) 查询所有授课的“教授”的工号、姓名和授课课程
SELECT 工号, 姓名, 授课课程 FROM 教师表;
```

五、 实验分析总结及心得

在本次实验中，通过对数据库中的教师、学生和课程等表进行查询操作，加深了对 SQL 语言的理解和应用。观察到通过不同的条件筛选和连接操作，可以准确地获取所需的数据信息，如查询特定学院的教师记录、课程记录以及特定课程的教师信息等。在实验过程中，遇到了一些挑战，特别是在处理复杂的条件筛选和连接查询时可能会出现语法错误或逻辑错误。通过对 SQL 语句的原理进行分析和调试，能够有效解决这些问题，并获得正确的查询结果。

通过这次实验，我不仅加深了对 SQL 查询语句的理解，还学会了如何灵活运用 SQL 语言进行数据查询和分析，为今后的数据库应用和开发提供了更多的技术支持。