

《数据库系统原理》实验

GaussDB实验报告书

|  |  |
| --- | --- |
| 学号 | 22373180 |
| 姓名 | 王睿风 |

2024 年 10 月 20 日

#### 任务一：建表并完善约束

1. 写出**建表时**每个字段的类型和建表时所使用的约束。

Student表

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 约束 |
| Sid | CHAR(8) | PRIMARY KEY |
| Sname | VARCHAR(20) | NOT NULL |
| Sgender | VARCHAR(2) | NOT NULL |
| Sage | INT | NOT NULL |

Teacher表

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 约束 |
| Tid | CHAR(9) | PRIMARY KEY |
| Tname | VARCHAR(20) | NOT NULL |
| Tgender | VARCHAR(2) | NOT NULL |
| Tage | INT | NOT NULL |
| Temail | VARCHAR(127) |  |
| Tphone | VARCHAR(30) | UNIQUE |

Course表

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 约束 |
| Cid | CHAR(8) | PRIMARY KEY |
| Cname | VARCHAR(30) | NOT NULL |
| Ctype | VARCHAR(10) | NOT NULL |
| Ccredit | DECIMAL | NOT NULL |
| Tid | CHAR(9) | NOT NULL |

SC表

|  |  |  |
| --- | --- | --- |
| 字段名 | 类型 | 约束 |
| Sid | CHAR(8) | NOT NULL |
| Cid | CHAR**(**8**)** | NOT NULL |
| Score | DECIMAL | NULL |

1. 列出表中应该为外键的字段。

Course表中的Tid, SC表中的Sid和Cid.

1. 请提交建立约束的sql语句。

|  |
| --- |
| ALTER TABLE Student  ADD CONSTRAINT CHK\_Student\_Sid CHECK (Sid REGEXP '^[0-9]{8}$'),  ADD CONSTRAINT CHK\_Student\_Sage CHECK (Sage <= 200 AND Sage >= 0);  ALTER TABLE Teacher  ADD CONSTRAINT CHK\_Teacher\_Tid CHECK (Tid REGEXP '^[0-9]{9}$'),  ADD CONSTRAINT CHK\_Teacher\_Tage CHECK (Tage <= 999 AND Tage >= 0);  ALTER TABLE Course  ADD CONSTRAINT CHK\_Course\_Cid CHECK (Cid REGEXP '^[0-9]{8}$'),  ADD CONSTRAINT CHK\_Course\_Ccredit CHECK (Ccredit >= 0 AND Ccredit <= 99);  ALTER TABLE SC  ADD CONSTRAINT CHK\_SC\_Score CHECK (Score <= 100); |

#### 任务二：插入数据

请阐述云数据库、分布式数据库的特性。

1. 云数据库特性：

* 维护简便：云服务提供商负责数据库的维护和升级，减少了用户的管理工作。
* 成本效益：用户只需为使用的资源付费，无需投资昂贵的硬件和软件。
* 安全性：云数据库提供多种安全特性，如数据加密、访问控制和安全合规性。
* 多租户：云数据库通常支持多租户架构，允许多个用户或组织共享同一数据库实例。
* 备份和恢复：云数据库通常提供自动备份和快速恢复功能，减少数据丢失的风险。

1. 分布式数据库特性：

* 透明访问：用户可以像访问单一数据库一样访问分布式数据库，而不需要关心数据是如何分布的。
* 容错性：分布式数据库通过冗余存储和复制来提高系统的容错能力。
* 并行处理：分布式数据库可以并行处理查询和事务，提高性能。
* 数据一致性：分布式数据库需要处理数据一致性问题，确保所有节点上的数据都是同步的。

#### 任务三：查询数据

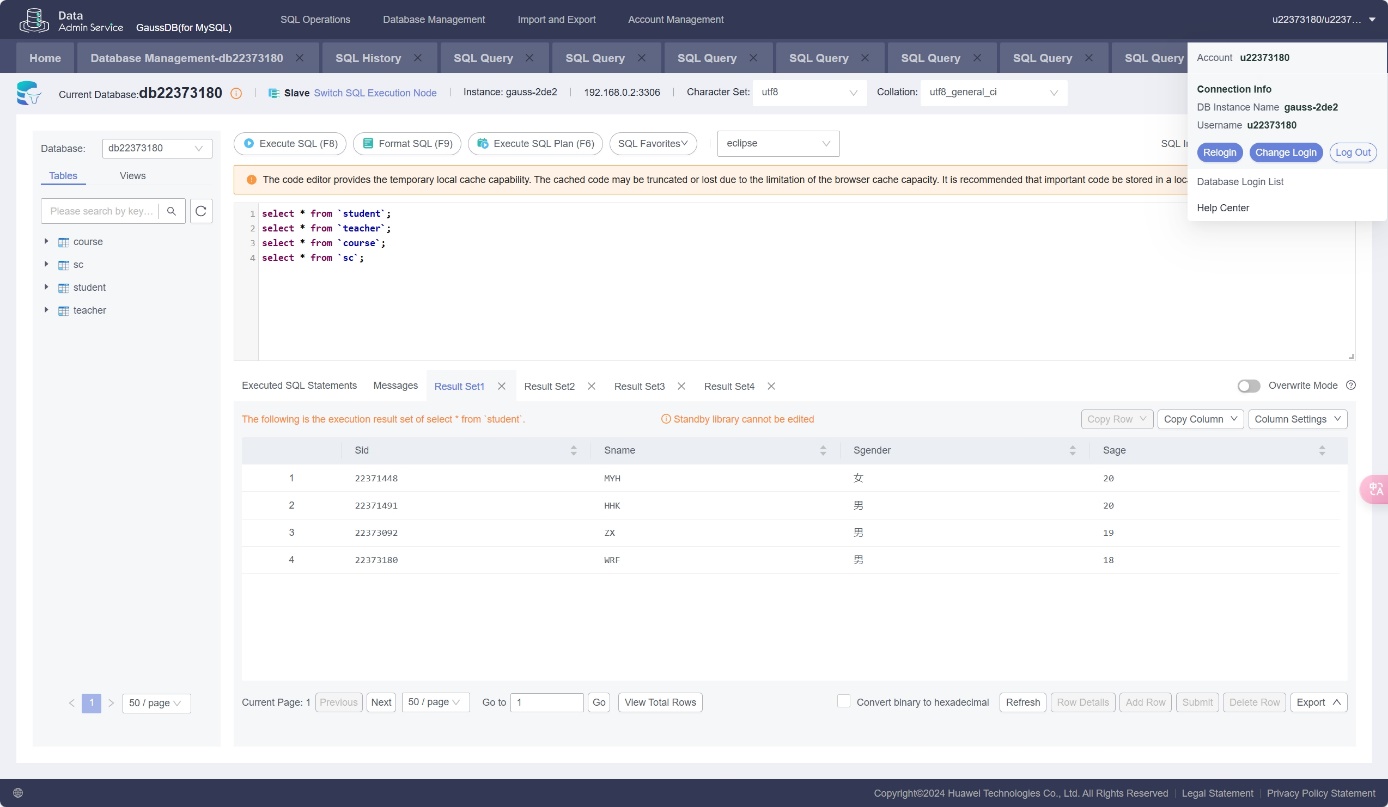
写出操作1和操作2的查询语句，以及操作2的查询条件A、B、C，并提交查询结果截图。

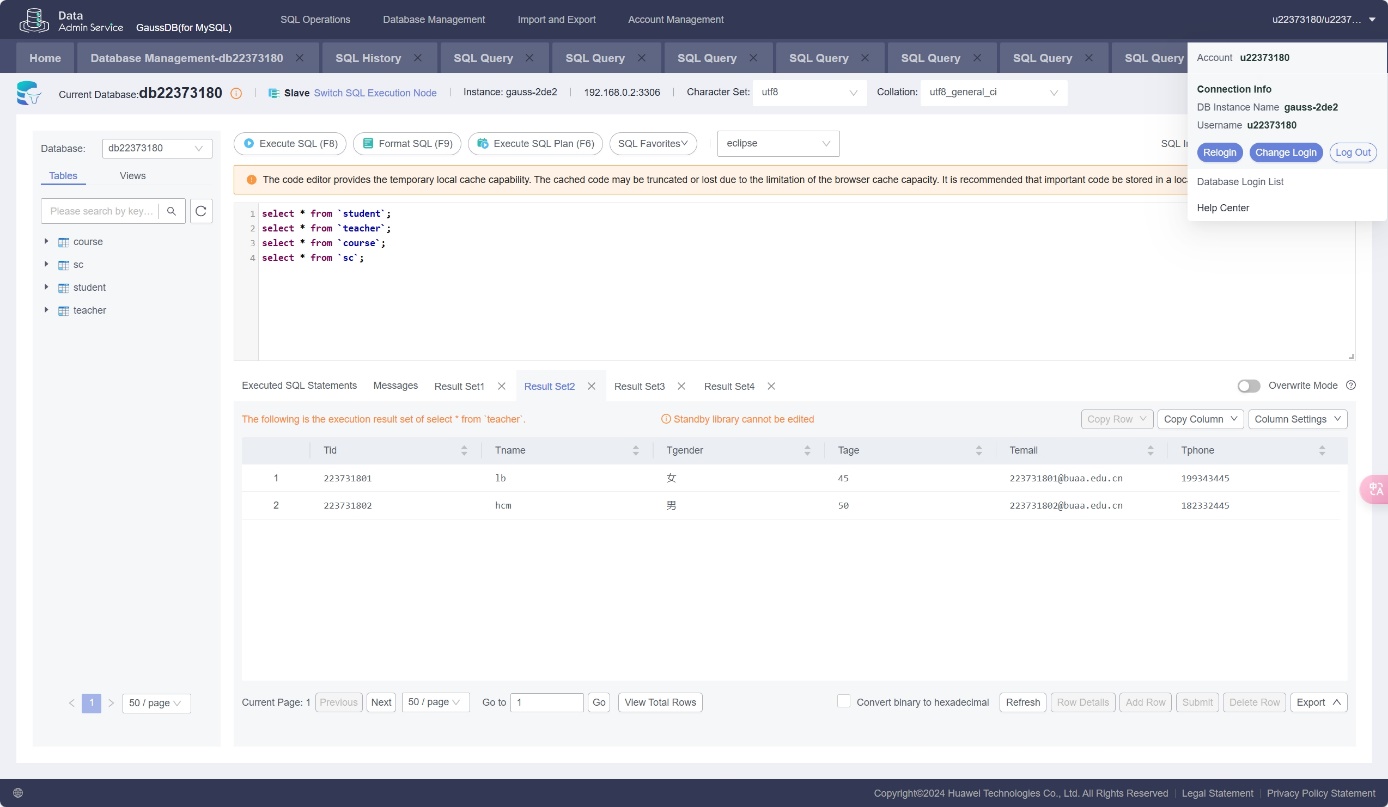
操作一：

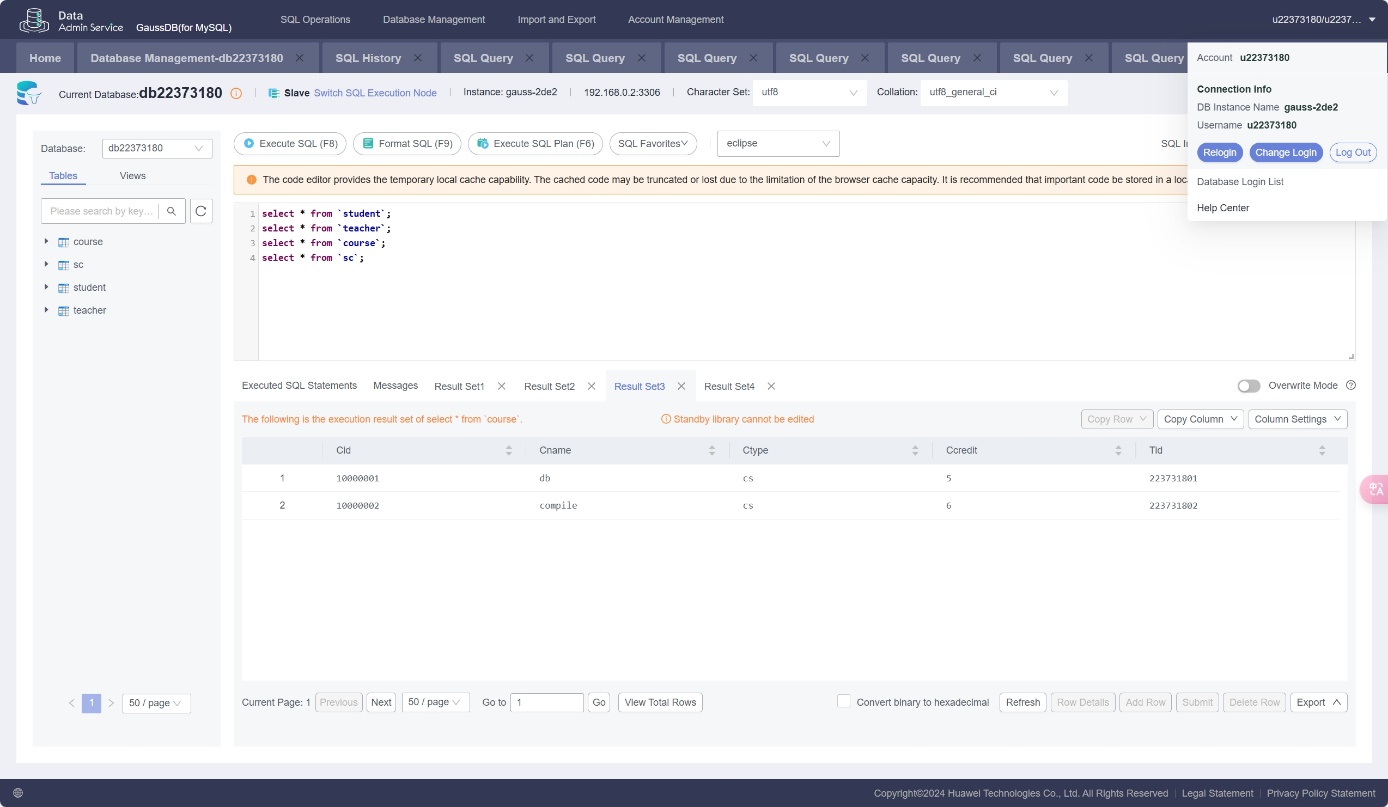
查询语句

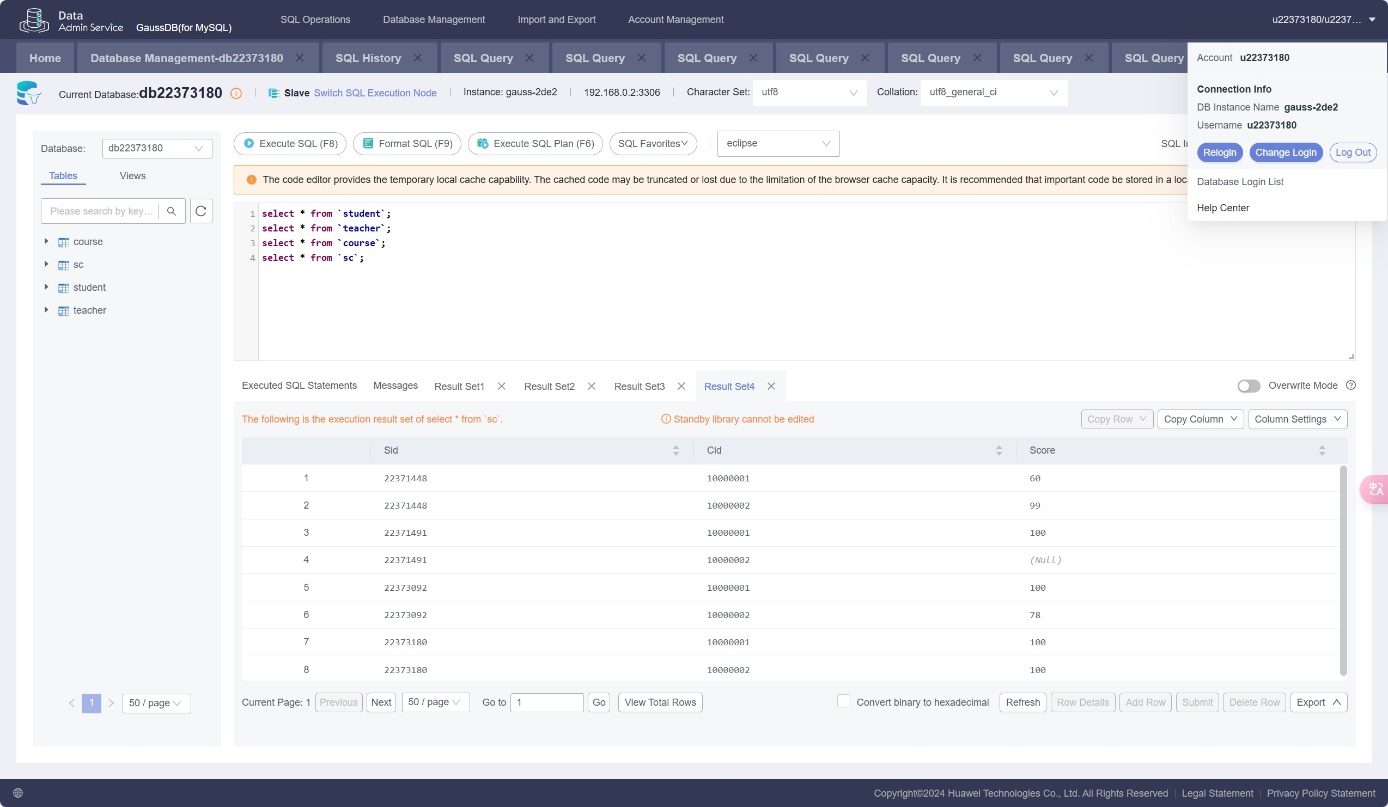
|  |
| --- |
| select \* from `student`;  select \* from `teacher`;  select \* from `course`;  select \* from `sc`; |

查询结果









操作二：

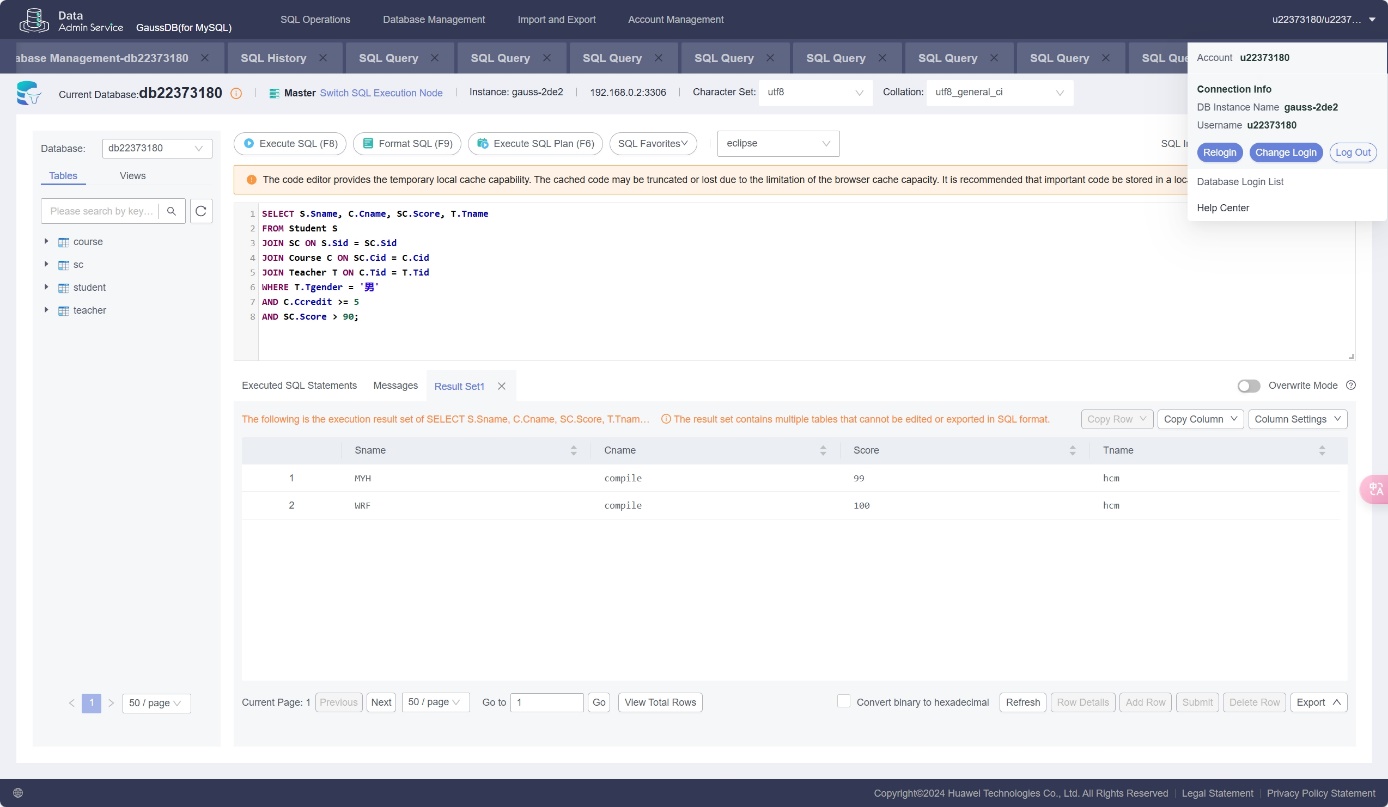
查询条件

|  |
| --- |
| A: 男  B: 5  C: 90 |

查询语句

|  |
| --- |
| SELECT S.Sname, C.Cname, SC.Score, T.Tname  FROM Student S  JOIN SC ON S.Sid = SC.Sid  JOIN Course C ON SC.Cid = C.Cid  JOIN Teacher T ON C.Tid = T.Tid  WHERE T.Tgender = '男'  AND C.Ccredit >= 5  AND SC.Score > 90; |

查询结果



#### 任务四：建立视图和索引

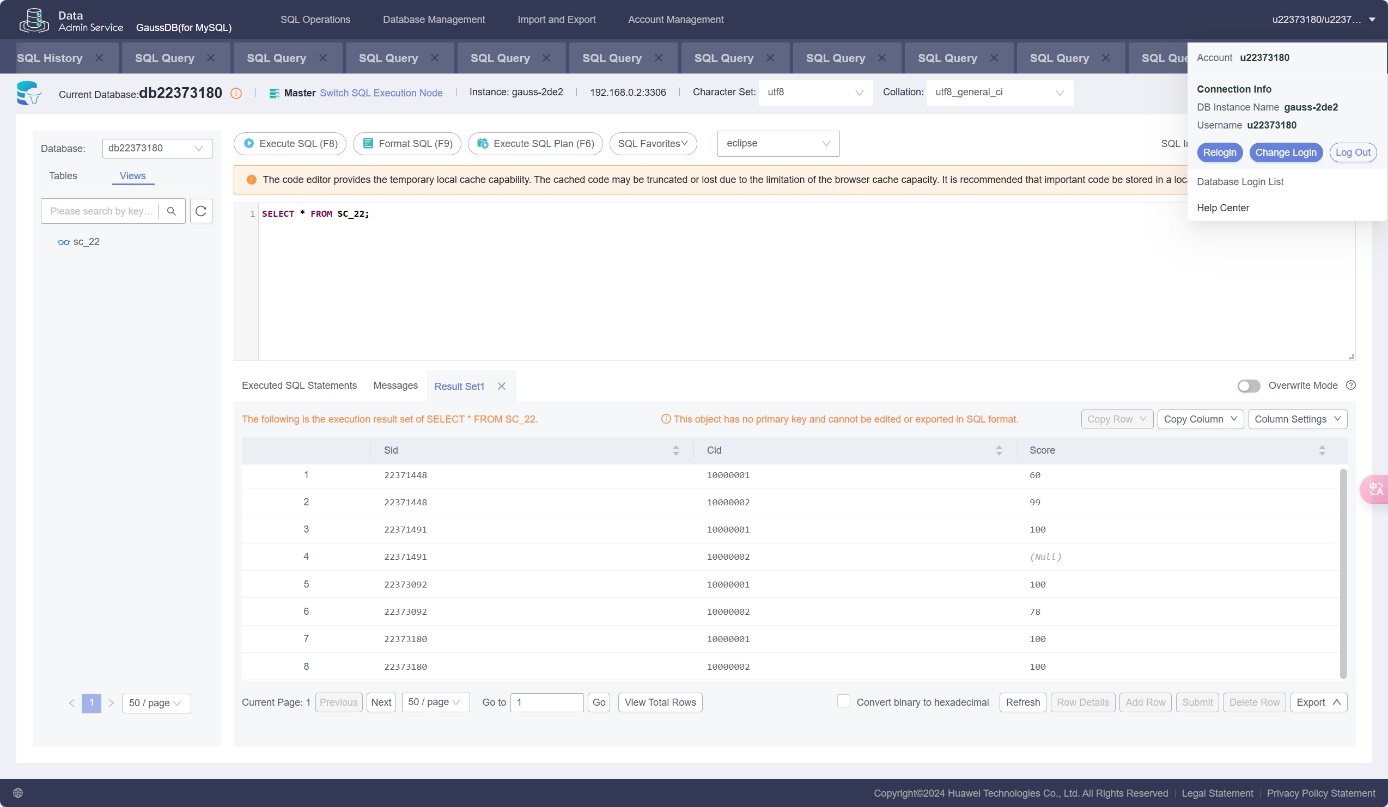
1. 写出操作1中建立视图的语句，以及条件X；在查询窗口中查询该视图并输出其所有列，提交查询结果截图。

条件X: 22

建立视图的语句：

|  |
| --- |
| CREATE VIEW SC\_22 AS  SELECT Sid, Cid, Score  FROM SC  WHERE Sid LIKE '22%'; |

查询结果



1. 写出操作2中建立普通索引的语句。

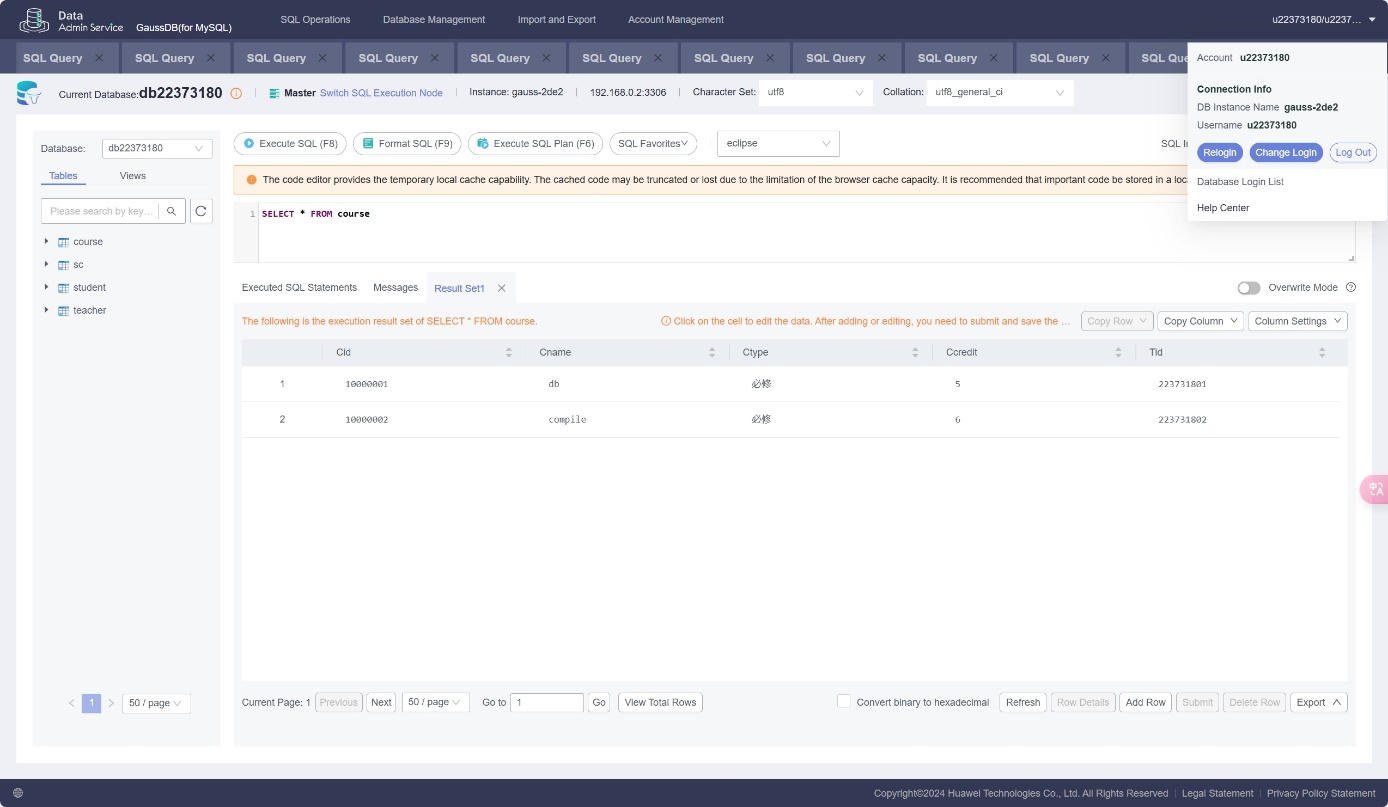
|  |
| --- |
| CREATE INDEX idx\_22373180 ON Teacher (Tid, Tname, Tage); |

#### 任务五：更新数据

1. 写出操作1和操作2中更新数据的语句，以及操作2的条件Y、Z，并提交更新后的数据截图。

操作一更新数据语句：

|  |
| --- |
| UPDATE Course  SET Ctype = '必修'  WHERE Cid IN (  SELECT Cid  FROM (  SELECT Cid  FROM Course  ORDER BY Cid DESC  LIMIT 2  ) AS subquery  ); |

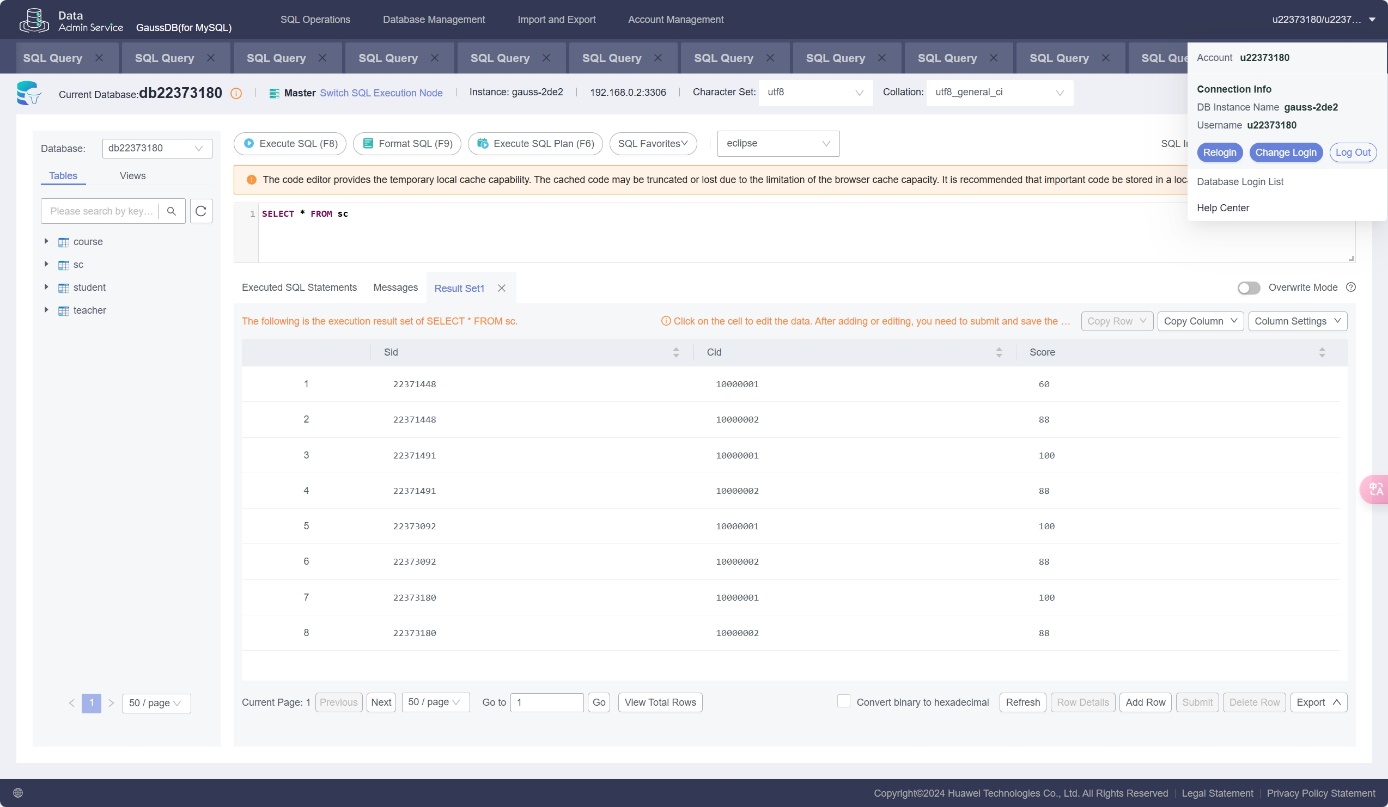


操作二更新数据语句：

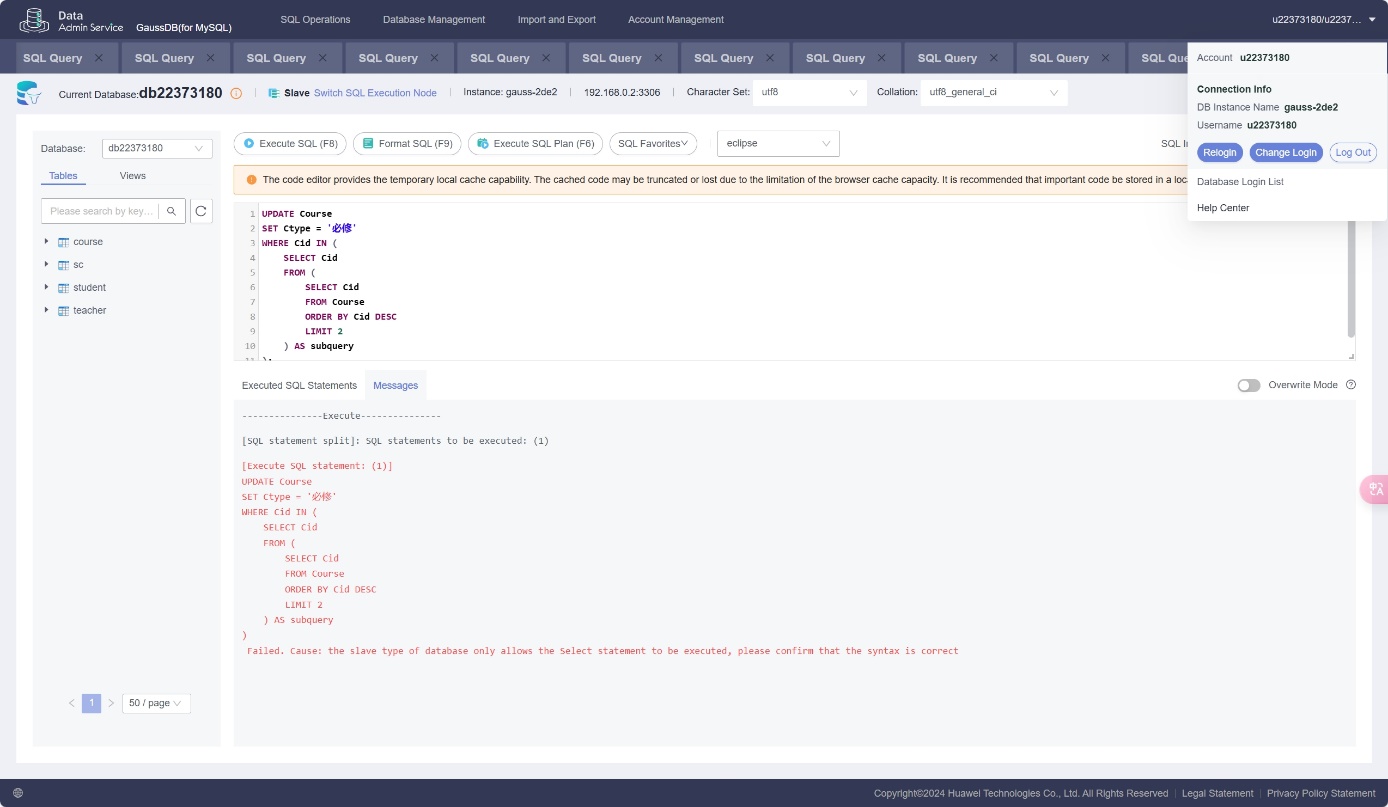
|  |
| --- |
| UPDATE sc  SET Score = 88  WHERE Cid in (  SELECT Cid  FROM course  WHERE Tid = (  SELECT Tid  FROM teacher  WHERE Tname = 'hcm'  )  ); |

操作二条件：

|  |
| --- |
| Y: hcm  Z: 88 |



1. 将第3条操作的结果截图附上，并说明分布式数据库读写分离的好处。



分布式数据库读写分离的好处：读写操作分离：使得数据库管理员可以更专注于写操作的性能优化，而不必担心读操作的影响；写操作通常需要锁定数据以保证数据一致性，而读操作则不需要。分离读写可以减少锁的争用，从而提高写操作的效率。

#### 任务六：删除数据

1. 写出操作1中删除数据的语句，以及条件A，并提交查询该同学修的课程及其成绩的语句及结果截图。

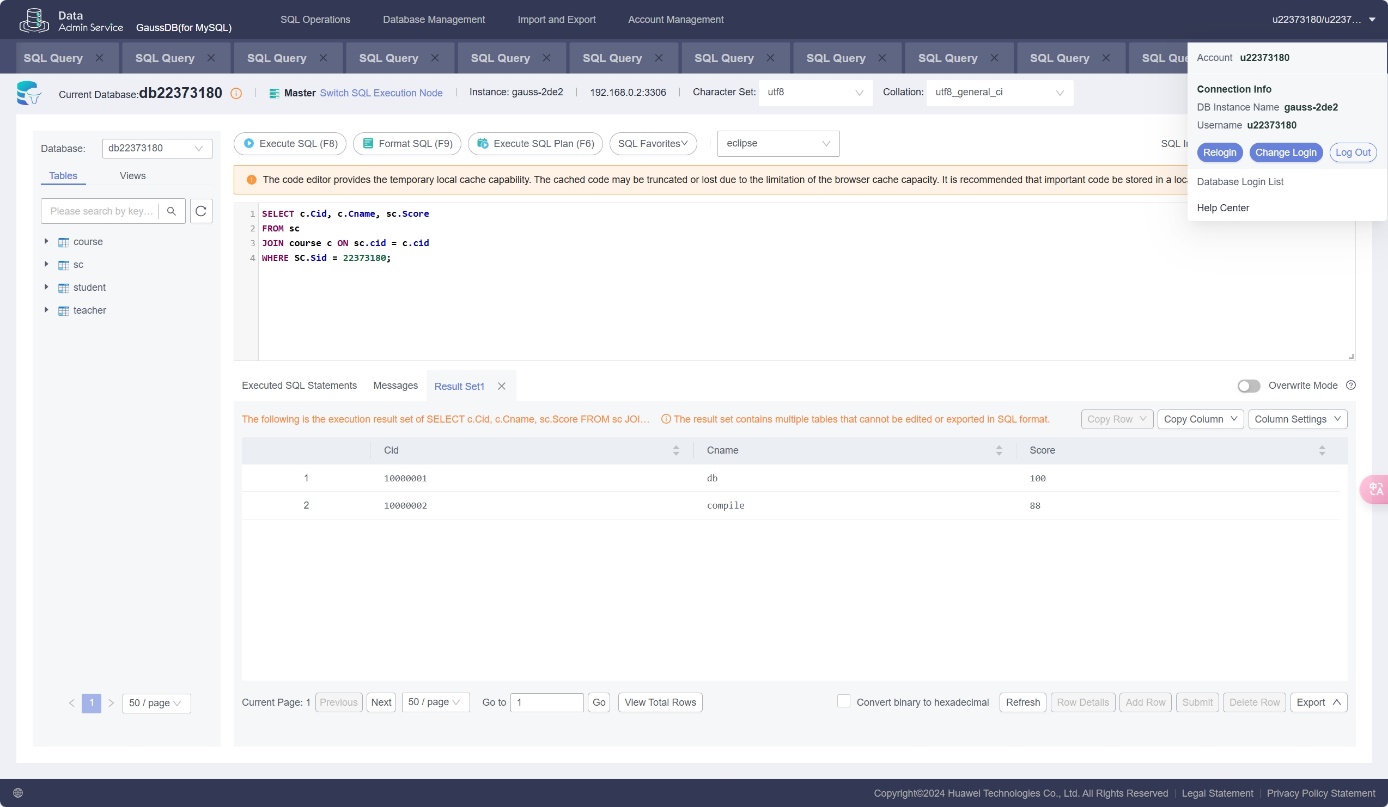
操作一删除数据的语句：

|  |
| --- |
| DELETE FROM student WHERE Sid = 22373180;  DELETE FROM sc WHERE Sid = 22373180; |

条件A: 22373180

询该同学修的课程及其成绩的语句：

|  |
| --- |
| SELECT c.Cid, c.Cname, sc.Score  FROM sc  JOIN course c ON sc.cid = c.cid  WHERE SC.Sid = 22373180; |

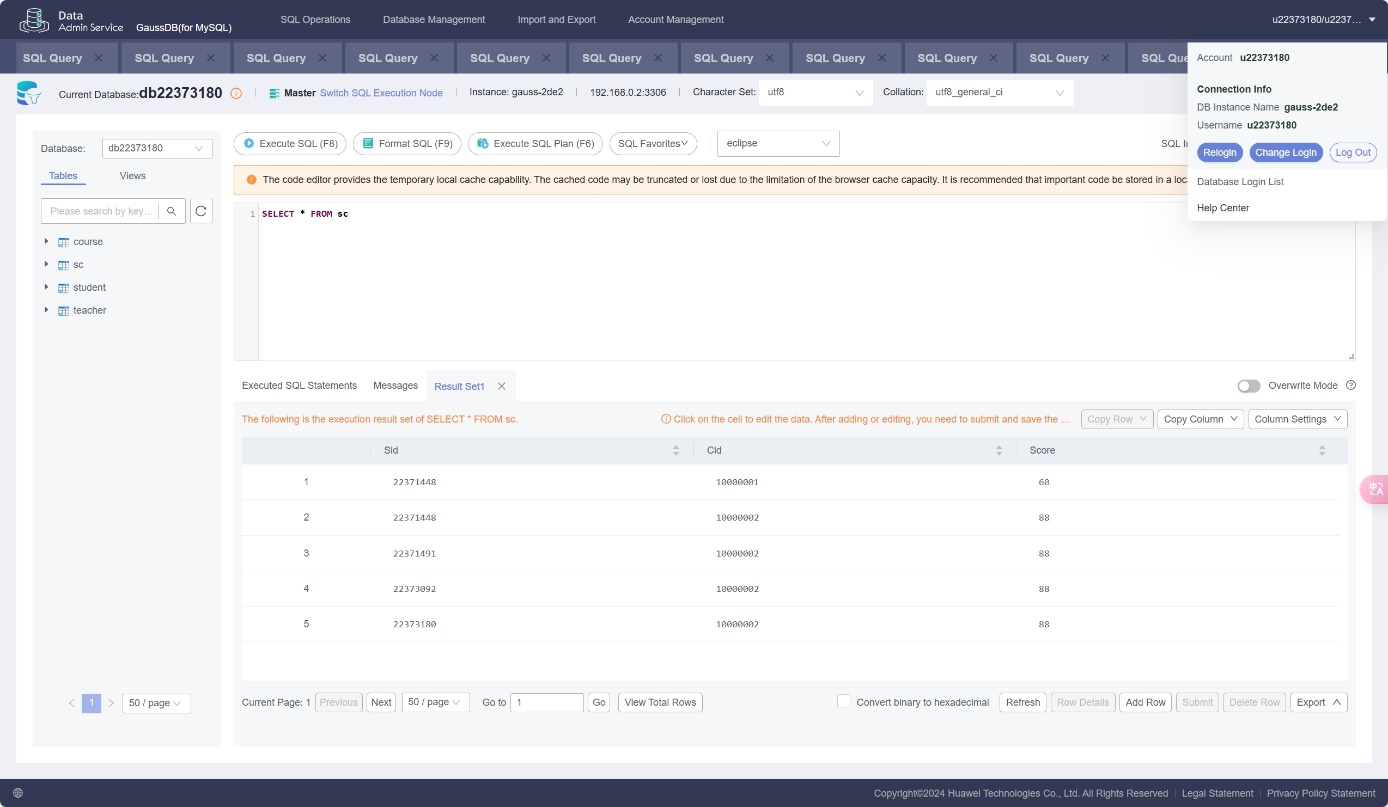


1. 写出操作2条中的删除语句，以及条件B，并提交查询删除后的结果截图。

条件B: 90

删除语句：

|  |
| --- |
| DELETE FROM sc  WHERE Score >= 90; |



#### 任务七：设置触发器

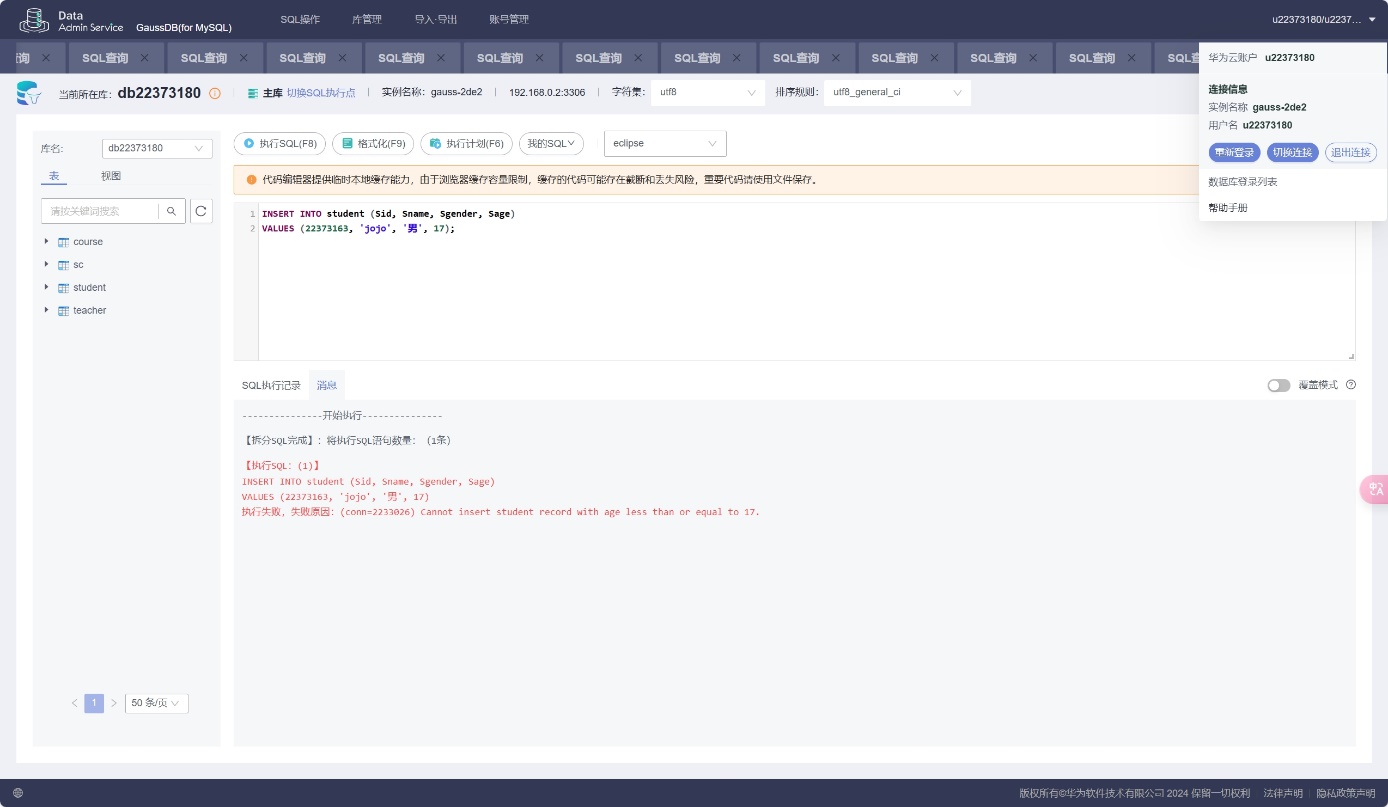
写出1, 2设置触发器的语句，以及条件C、D，并提交插入数据失败的截图。

C: 17

1设置触发器语句：

|  |
| --- |
| DELIMITER //  CREATE TRIGGER insert\_lower\_17  BEFORE INSERT ON student FOR EACH ROW  BEGIN  IF NEW.Sage <= 17 THEN  SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Cannot insert student record with age less than or equal to 17.';  END IF;  END //  DELIMITER ; |

插入数据失败的截图：



条件D: 22

|  |
| --- |
| DELIMITER //  CREATE TRIGGER insert\_lower\_22  BEFORE INSERT ON teacher FOR EACH ROW  BEGIN  IF NEW.Tage <= 22 THEN  SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Cannot insert teacher record with age less than or equal to 22.';  END IF;  END //  DELIMITER ; |

