**南昌大学实验报告**



学生姓名： 杨孟衡 学 号： 8002118240 专业班级： 软工1809

实验类型：☑ 验证 □ 综合 □ 设计 □ 创新 实验日期： 3.26 实验成绩：

1. 实验项目名称

实验三 索引和视图

1. 实验内容

1. 使用SQL语句ALTER TABLE分别删除studentsdb数据库的student\_info表、grade表、curriculum表的主键索引。

2. 使用SQL语句为curriculum表的课程编号创建唯一性索引，命名为cno\_idx。

3. 使用SQL语句为grade表的“分数”字段创建一个普通索引，命名为grade\_idx。

4. 使用SQL语句为grade表的“学号”和“课程编号”字段创建一个复合唯一索引，命名为grade\_sid\_cid\_idx。

5. 查看grade表上的索引信息。

6. 使用SQL语句删除索引grade\_idx。再次查看grade表上的索引信息。

7. 使用SQL语句CREATE VIEW建立一个名为v\_stu\_c的视图，显示学生的学号、姓名、所学课程的课程编号，并利用视图查询学号为0003的学生情况。

8. 基于student\_info表、curriculum表和grade表，建立一个名为v\_stu\_g的视图，视图包括所有学生的学号、姓名、课程名称、分数。使用视图v\_stu\_g查询学号为0001的学生的课程平均分。

9. 使用SQL语句修改视图v\_stu\_g，显示学生的学号、姓名、性别。

10.利用视图v\_stu\_g为student\_info表添加一行数据：学号为0010、姓名为陈婷婷、性别为女。

11.利用视图v\_stu\_g删除学号为0010的学生记录。

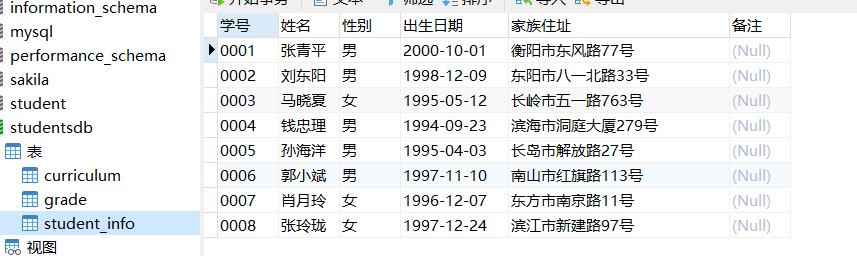
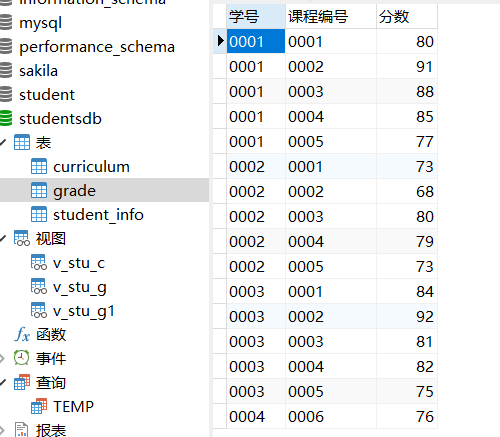
12.利用视图v\_stu\_g修改姓名为张青平的学生的高等数学的分数为87。

13.使用SQL语句删除视图v\_stu\_c和v\_stu\_g。

实验环境：windows10操作系统,，mysql8.0.16，Navicat12.0

1. 实验要求
2. 截图SQL代码及运行结果图，截图清晰。
3. 学习通平台提交实验报告。
4. 文件名：姓名+学号+实验几+班级
5. 实验步骤及处理结果

三个实验数据表数据总览：



1.SELECT语句-索引：

* 打开Navicat，连接mysql数据库，登录用户，选择数据库studentsdb，新建一个查询temp；

（1）利用ALTER TABLE语句删除主键索引：

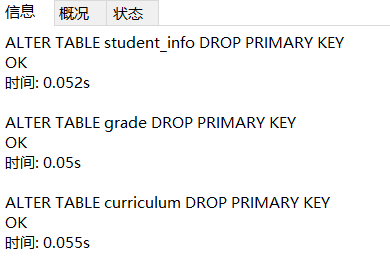
* 在打开的查询页面中输入以下sql语句：

ALTER TABLE student\_info DROP PRIMARY KEY;

ALTER TABLE grade DROP PRIMARY KEY;

ALTER TABLE curriculum DROP PRIMARY KEY;

* 运行查询，查询结果如图：

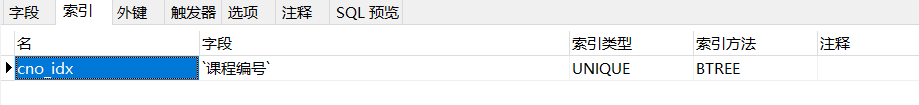
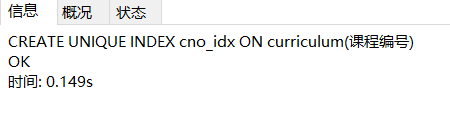


（2）使用sql语句为curriculum表的课程编号创建唯一性索引：

* 在打开的查询页面中输入以下sql语句：

CREATE UNIQUE INDEX cno\_idx ON curriculum(课程编号);

* 运行查询，查询结果如图：

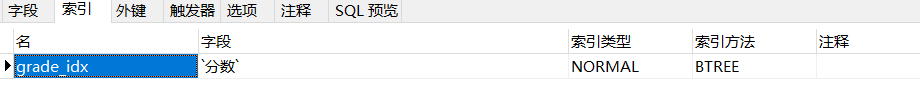
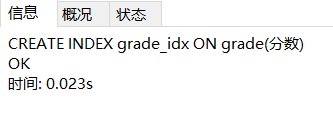


（3）使用sql语句为grade表的分数字段创建一个普通索引：

* 在打开的查询页面中输入以下sql语句：

CREATE INDEX grade\_idx ON grade(分数);

* 运行查询，查询结果如图：

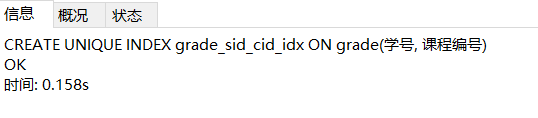


（4）使用sql语句为grade表的学号和课程编号字段创建一个复合唯一索引：

* 在打开的查询页面中输入以下sql语句：

CREATE UNIQUE INDEX grade\_sid\_cid\_idx ON grade(学号, 课程编号);

* 运行查询，查询结果如图：

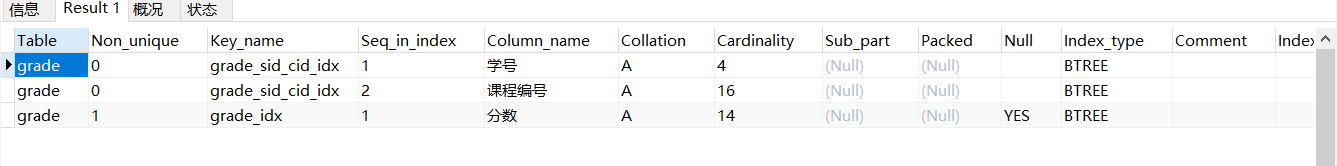


（5）查看grade表上的索引信息：

* 在打开的查询页面中输入以下sql语句：

SHOW INDEX FROM grade;

* 运行查询，查询结果如图：



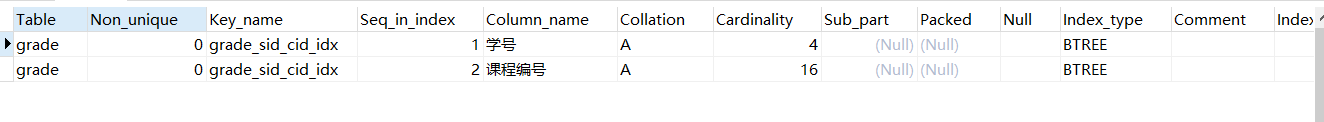
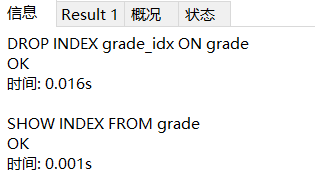
（6）使用sql语句删除索引grade\_idx，再次查看索引信息：

* 在打开的查询页面中输入以下sql语句：

DROP INDEX grade\_idx ON grade;

SHOW INDEX FROM grade;

* 运行查询，查询结果如图：



2.SELECT语句-视图：

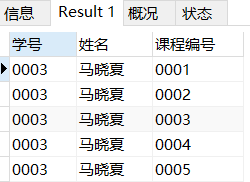
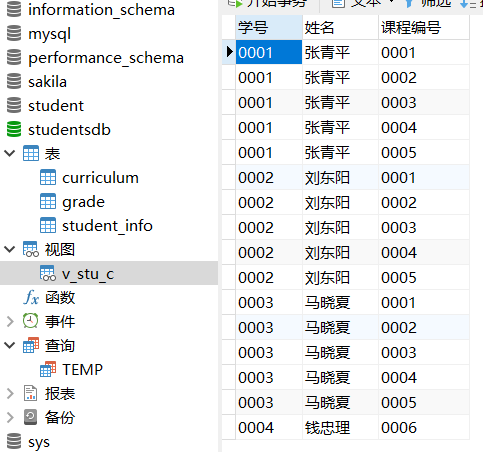
（1）使用sql语句CREATE VIEW建立视图，显示学号，姓名，所学课程的课程编号，并利用视图查询学号为0003的学生情况：

* 在打开的查询页面中输入以下sql语句：

CREATE VIEW v\_stu\_c AS SELECT S.学号, S.姓名, G.课程编号 FROM student\_info S, grade G WHERE S.学号=G.学号;

SELECT \* FROM v\_stu\_c WHERE 学号='0003';

* 运行查询，查询结果如图：



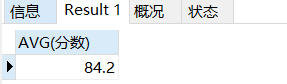
（2）基于student\_info，grade, curriculum表建立一个视图，包括学号，姓名，课程名称，分数，使用视图查询学号为0001的学生的课程平均分：

* 在打开的查询页面中输入以下sql语句：

CREATE VIEW v\_stu\_g AS SELECT S.学号, S.姓名, C.课程名称, G.分数 FROM student\_info S LEFT JOIN grade G ON S.学号=G.学号 LEFT JOIN curriculum C ON G.课程编号=C.课程编号;

SELECT \* FROM v\_stu\_g WHERE 学号='0001';

* 运行查询，查询结果如图：

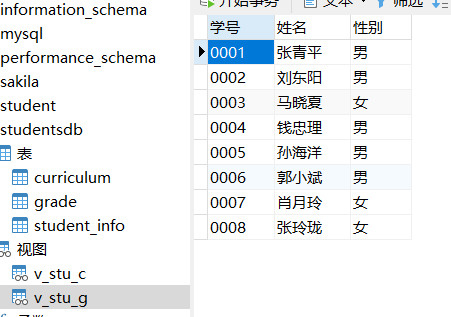
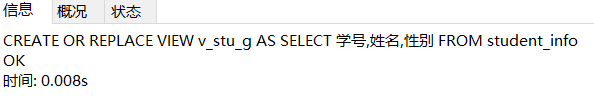


（3）使用sql语句修改视图v\_stu\_g，显示学生的学号，姓名，性别：

* 在打开的查询页面中输入以下sql语句：

CREATE OR REPLACE VIEW v\_stu\_g AS SELECT 学号,姓名,性别 FROM student\_info;

* 运行查询，查询结果如图：

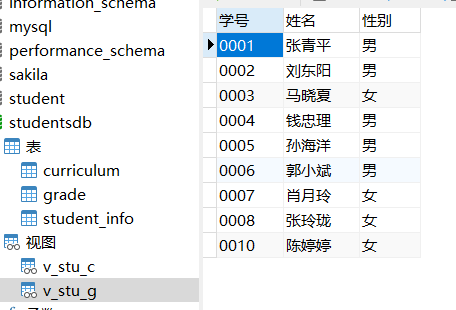


（4）利用视图为student\_info表添加一行数据：

* 在打开的查询页面中输入以下sql语句：

INSERT INTO v\_stu\_g(学号,姓名,性别) VALUES('0010', '陈婷婷', '女');

* 运行查询，查询结果如图：

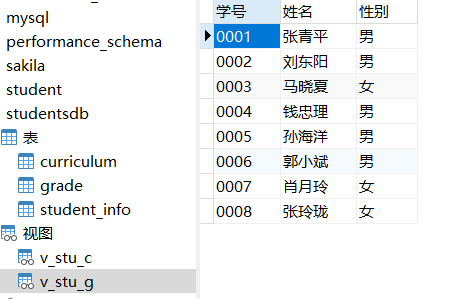


（5）利用视图删除学号为0010的学生记录：

* 在打开的查询页面中输入以下sql语句：

DELETE FROM v\_stu\_g WHERE 学号='0010';

* 运行查询，查询结果如图：



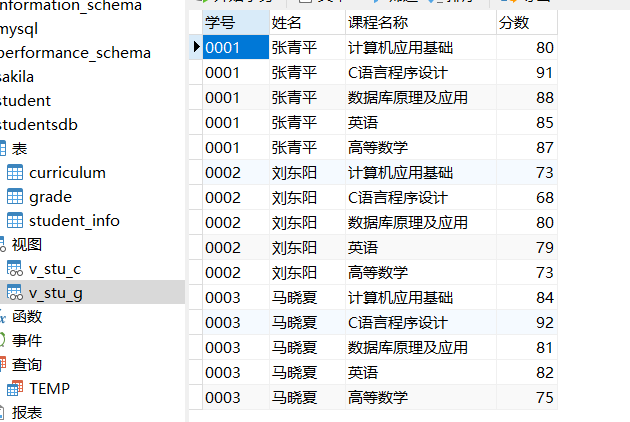
（6）利用视图修改姓名为张青平的学生的高等数学的分数为87：

* 在打开的查询页面中输入以下sql语句：

CREATE OR REPLACE VIEW v\_stu\_g AS SELECT S.学号, S.姓名, C.课程名称, G.分数 FROM student\_info S, grade G, curriculum C WHERE S.学号=G.学号 AND G.课程编号=C.课程编号;

UPDATE v\_stu\_g SET 分数=87 WHERE 姓名='张青平' AND 课程名称='高等数学';

* 运行查询，查询结果如图：



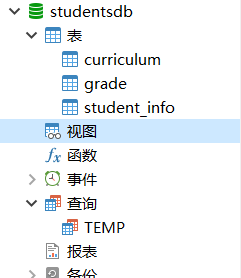
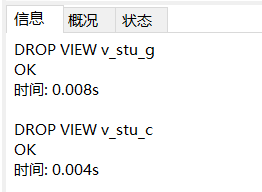
（7）删除视图v\_stu\_c和v\_stu\_g：

* 在打开的查询页面中输入以下sql语句：

DROP VIEW v\_stu\_g;

DROP VIEW v\_stu\_c;

* 运行查询，查询结果如图：



1. 思考体会或对改进实验的建议

本次实验是本学期数据库学习第三次试验，难度适中，个人感受一般，从这次实验里复习了索引和视图相关的sql语句，加深了对索引和视图的理解和使用，进一步增强了对于视图和数据表并非是同类概念的认识，以及视图和数据表之间的数据更新操作准允条件，视图可以方便的选择数据表中的数据进行显示而没有必要将数据表显示给用户，增强了安全性的同时，灵活性也大大增强。本次实验也没有花太多的时间，主要还是一个复习巩固以及实操mysql语句的作用，让我又好好的复习了mysql语句与sql语句之间的差别，大同小异，但小细节错一点点都不行，这教会了我作为程序猿的基本素质，抓细节！

花时间花的比较多的就是视图那里，虽然不是时间也不是很长，但是加深了对视图的应用理解，特别是利用视图更新数据表中的数据，需要程序猿仔细去斟酌，仔细检查语句正确性，否则在这里容易发生BUG。不过多来一点这类题目也可以检验我们自己编写代码，修改代码的能力，培养逻辑思维能力，总之这是一次非常不错的实验。

1. 参考资料

[1]何玉洁.数据库系统教程（第二版）[M].北京：人民邮电出版社, 978-7-115-40245-5