**武汉大学国家网络安全学院**

**实验报告**

**课程名称 数据库**

**专业年级 信安21级**

**姓名学号 赵伯俣2021302181156**

**赵晨阳2021302181066**

**实验学期 2022-2023　 学年 第二 学期**

**课堂时数 32 课外时数**

**填写时间 2023 年 5 月 13 日**

## 一、实验目的

1. 掌握数据库环境（LAMP，Linux+Apache+MySQL+PHP）的搭建过程。

2. 熟悉MySQL的基本操作（创建用户、登录、执行SQL、创建数据库、导入数据库、数据库完整性检查、数据库备份）。

3. 学会创建DDL脚本，包括创建、删除表，指定主键、候选键和外键；学会修改并展示表结构，学会创建、删除索引。

4. 掌握数据的增、删、查、改等操作，学会分析在操作过程中出现失败的原因；掌握视图的创建、删除和更新；理解视图消解机制，学会更新视图的操作并分析更新过程中出现失败的原因。

5. 掌握order by、group by、having等子句的使用；掌握各种谓词的使用；掌握集合函数的使用；掌握在时间字段上的查询。

6. 掌握嵌套子查询；掌握多表的连接查询，包括左连接、右连接、外连接、自连接。

## 二、实验内容

1. 在Windows系统中安装VMWare虚拟机，在VMWare中安装Ubuntu系统，在Ubuntu中搭建LAMP实验环境。

2. 使用MySQL进行一些基本操作：

（1）登录MySQL，在MySQL中创建用户，并对新建的用户赋予权限。

（2）创建并跳转到新的数据库，显示所有数据库和当前数据库。

（3）显示所有的表和查看表的属性。

（4）导入employees和sakila两个样例数据库，对这两个数据库进行完整性检查，对这两个数据库进行备份、导入与导出。

3. SQL DDL使用

（1） 创建DDL脚本，包括创建、删除表，指定主键、候选键和外键；

（2）修改并展示表结构

（3）创建、删除索引。

4. SQL DDL & DML使用

（1）数据的增、删、查、改等操作

（2）掌握视图的创建、删除和更新

5. 简单查询

（1）使用order by、group by、having等子句；

（2）使用各种谓词；

（3）使用集合函数；

（4）在时间字段上查询。

6. 复杂查询

（1）嵌套子查询；

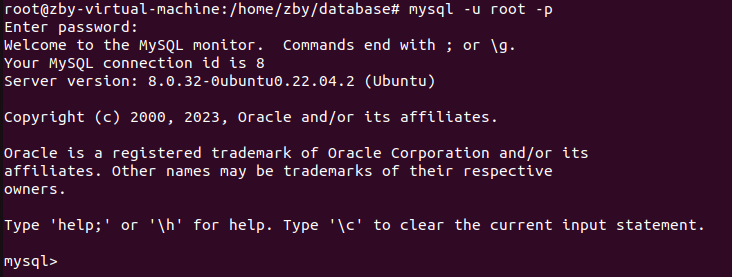
（2）多表连接查询，包括左连接、右连接、外连接、自连接。

## 三、实验步骤

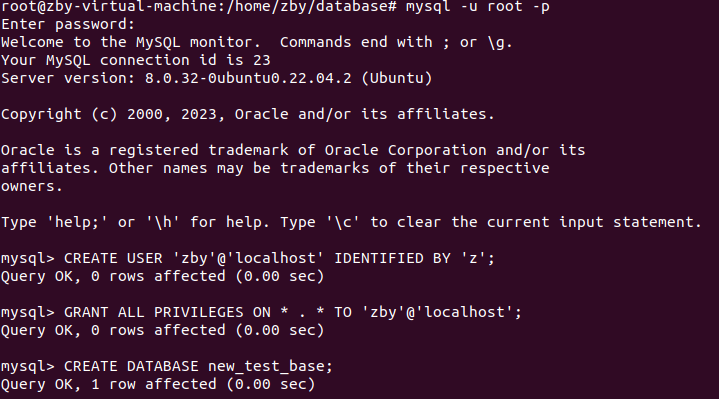
1. 首先在装有ubuntu22.04.2系统的Vmware虚拟机上使用命令sudo apt-get install apache2 ，sudo apt-get install mysql-server， sudo apt-get install php安装LAMP。

2. 熟悉MySQL的基本操作步骤如下：

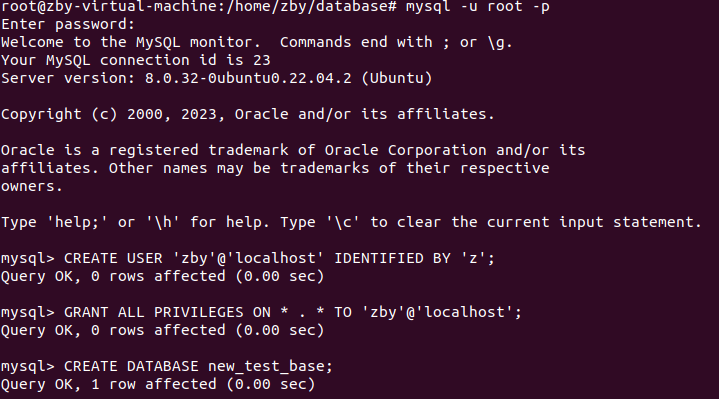
（1）进入MySQL：mysql -u root -p，命令操作结果如下



（2）创建新用户：CREATE USER 'zby'@'localhost' IDENTIFIED BY 'z';创建自己用户名zby和登陆密码z。命令执行情况如下所示



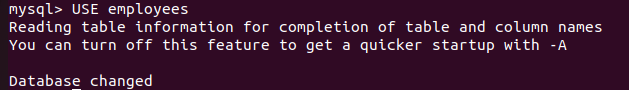
（3）赋予权限：GRANT ALL PRIVILEGES ON \* . \* TO 'zby'@'localhost';命令执行情况如下所示



（4）创建新的数据库：CREATE DATABASE new\_test\_database;创建名为new\_test\_database的新测试数据库。

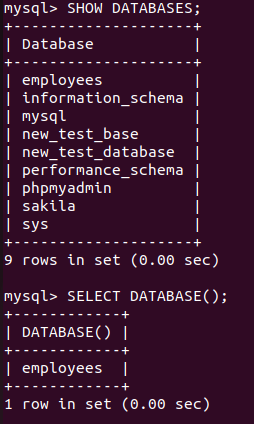
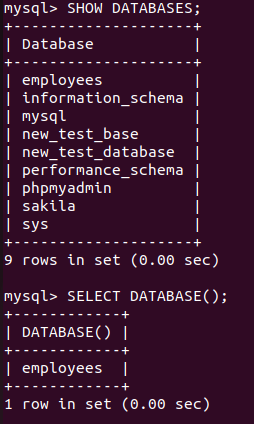
20230513152141

（5）跳转到employees数据库：USE employees;



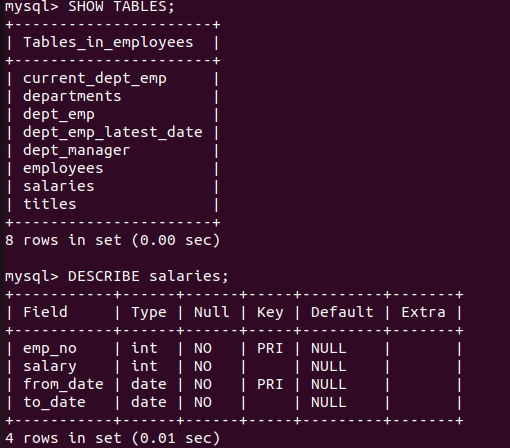
（6）显示所有数据库和当前数据库：

SHOW DATABASES; SELECT DATABASE();

（7）显示所有的表并查看salaries表的属性：

SHOW TABLES;DESCRIBE salaries;



（8）导入两个样例数据库

① 解压数据包：unzip test\_db\_master.zip

② 进入数据包目录：cd test\_db\_master

③ 导入employee数据库：sudo sudo mysql -t < employees.sql

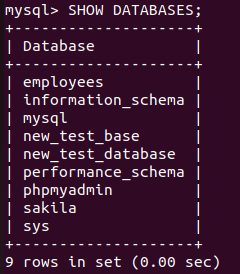
④ 解压数据包：unzip sakila-db.zip

⑤ 进入数据包目录：cd sakila-db

⑥ 进入MySQL，导入sakila schema数据：SOURCE sakila-schema.sql

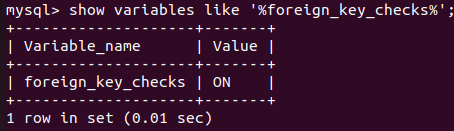
⑦ 导入sakila数据：SOURCE sakila-data.sql

导入结果如下图所示：employees库和sakila库均已经导入到mysql中。

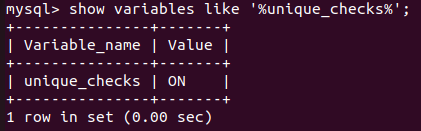


（9）对导入的两个数据库进行完整性检查

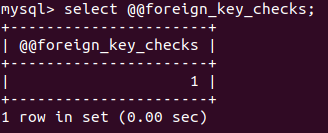
① 查看外键检查状态：show variables like ‘%foreign\_key\_checks%’;



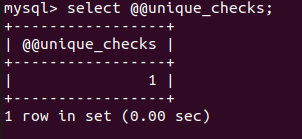
② 查看主键检查状态：show variables like ‘%unique\_checks%’;



③ 查看外键检查结果：select @@foreign\_key\_checks;



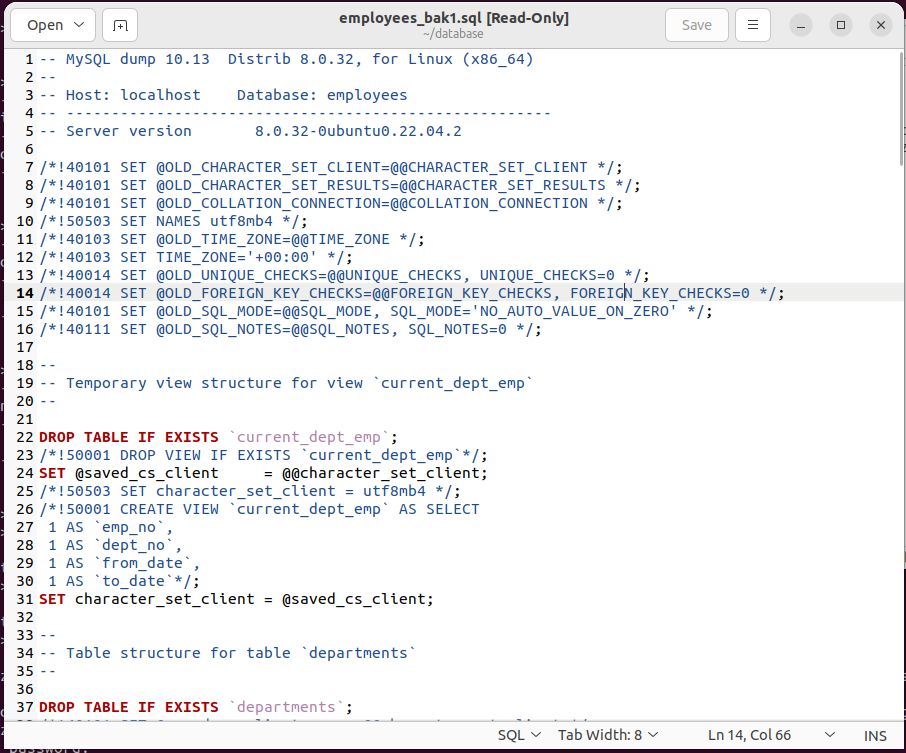
④ 查看主键检查结果：select @@unique\_checks;



（10）数据库备份（以employee为例）

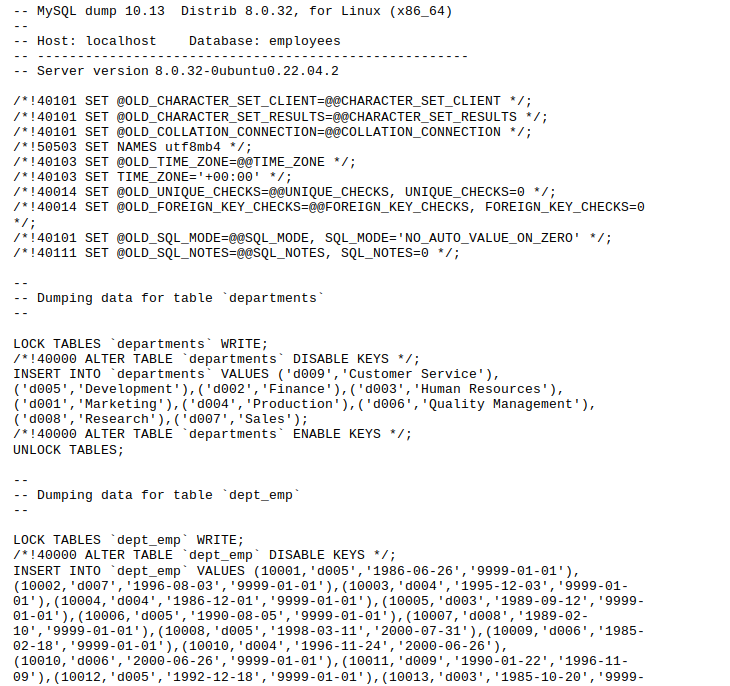
① 不导出任何数据，只导出数据库表结构：

mysqldump -u zby -p --no-data employees> employees\_bak1.sql导出后结果如下图所示



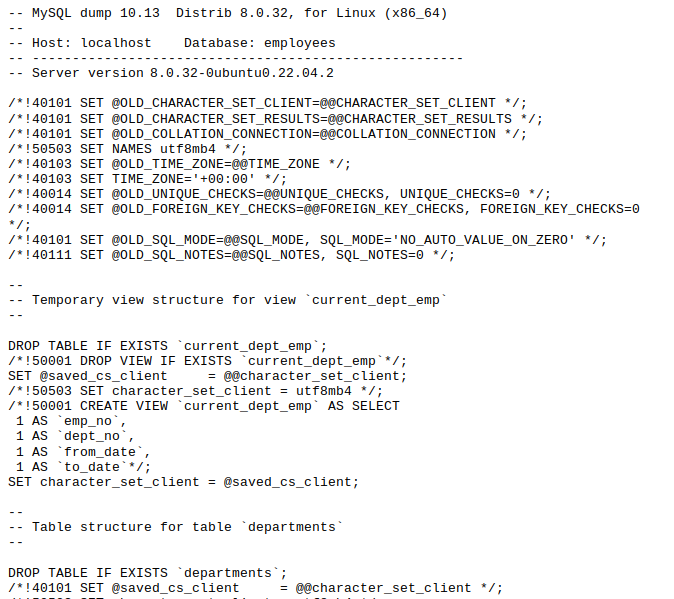
② 只导出数据，而不添加CREATE TABLE 语句：

mysqldump -u zby -p --no-create-info employees> employees\_bak2.sql，导出后结果如下图所示



③ 导出全部数据库：

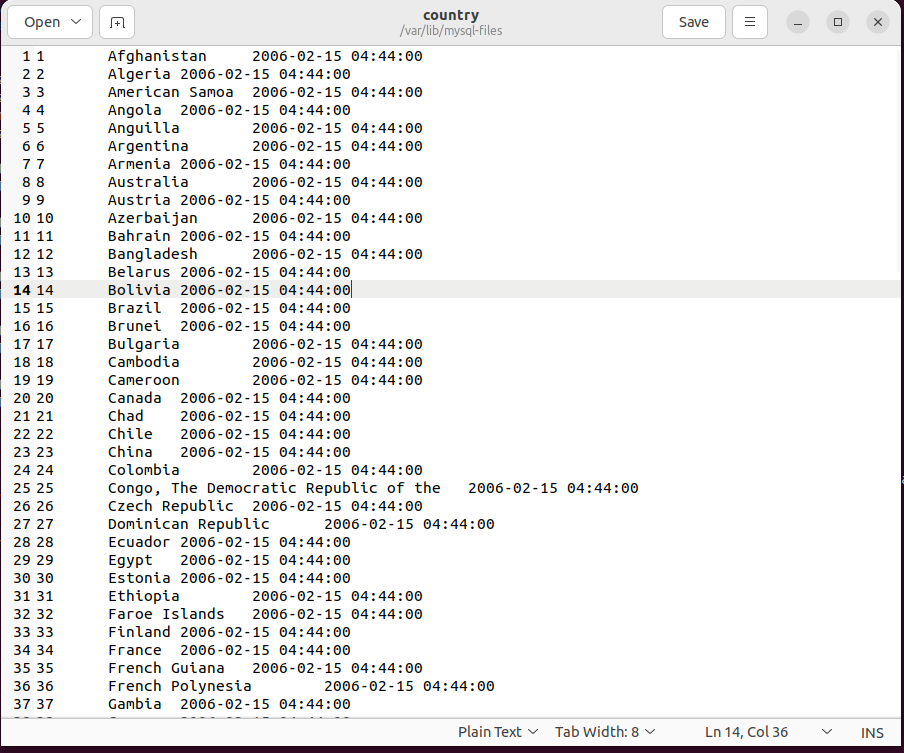
mysqldump -u test\_02 -p employees> employees\_bak3.sql



（11）导出和导入（以sakila为例）

① 导出为TXT文件：

select \* from country into outfile '/var/lib/mysql-files/country';导出结果如下图所示：



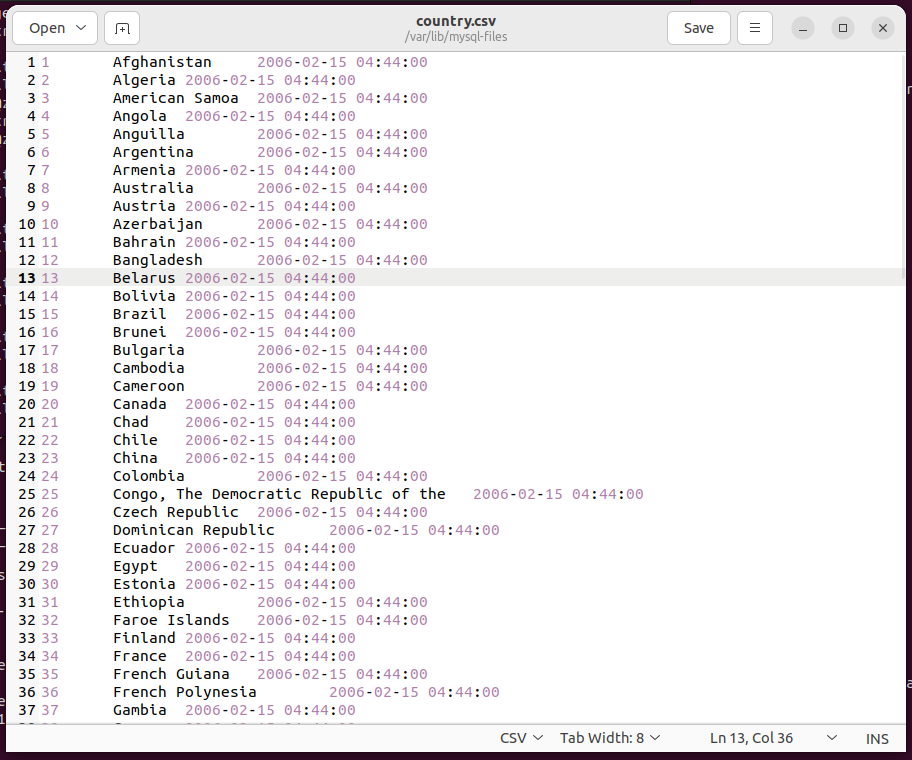
② 导入TXT文件：

load data infile '/var/lib/mysql-files/country' into table country;因为在导入时该数据库中已经存在导入数据，所以会存在以下形式的报错。

20230513162420

③ 导出为CSV文件：

select \* from country into outfile '/var/lib/mysql-files/country.csv';



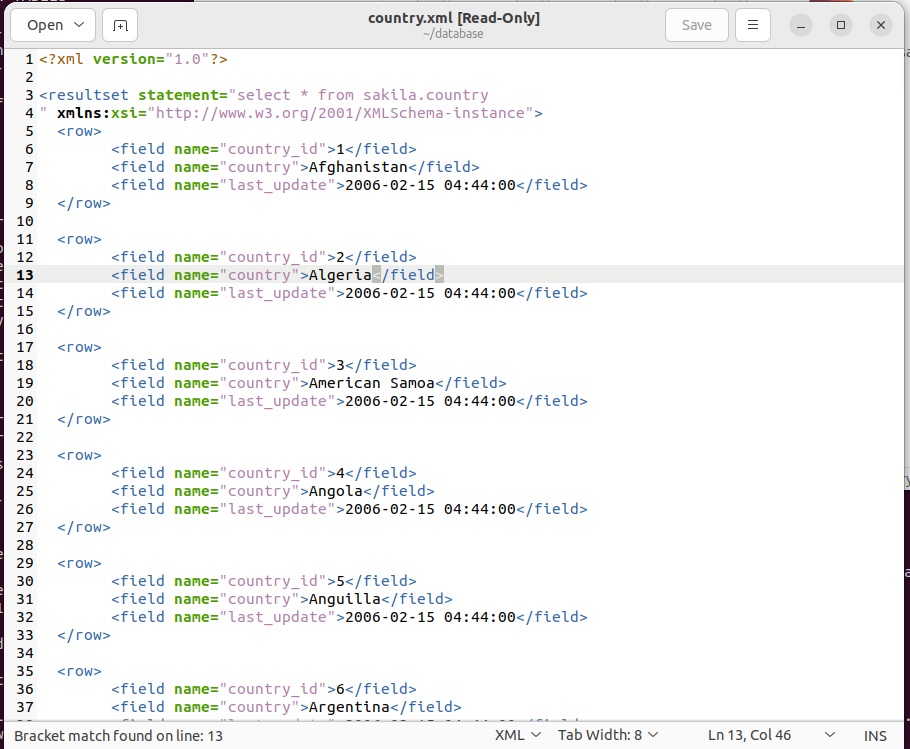
④ 导入CSV文件：

load data infile '/var/lib/mysql-files/country.csv' into table country;因为在导入时该数据库中已经存在导入数据，所以会存在以下形式的报错。

20230513162713

⑤ 导出为xml文件：

mysql -u zby -p --xml -e 'select \* from sakila.country' > country.xml,结果如下图所示。



3. SQL DDL使用

（1）在本次实验中所采用的数据库如下图1所示，使用create语句创建项目所有的表；执行情况如下图2所示

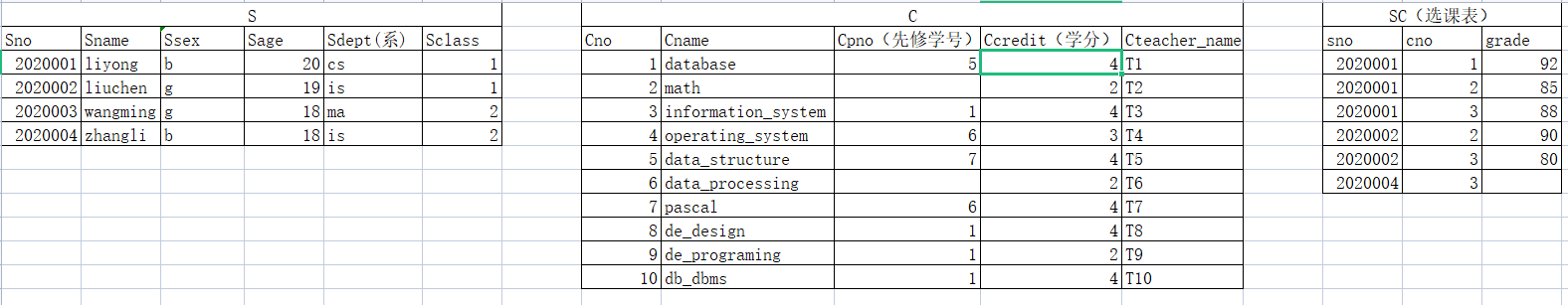


图1

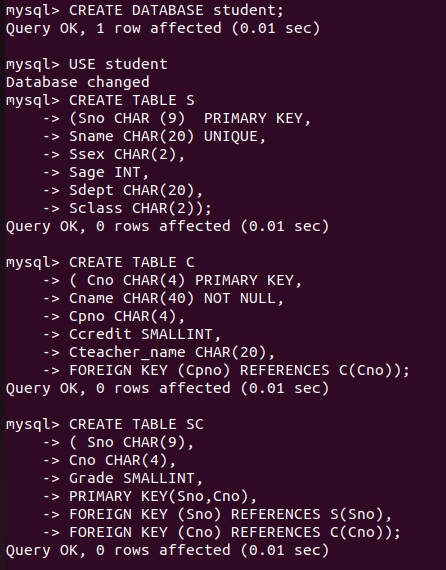
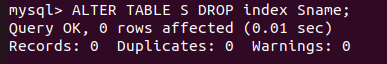
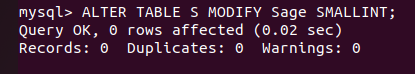


图2

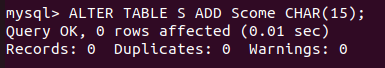
（2）使用alter命令修改表结构，删除学生表中学生姓名必须取唯一值的约束。操作结果如下图所示。



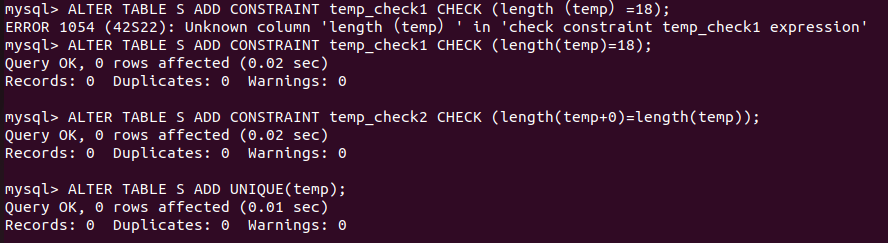
（3）将年龄的数据结构修改为SMALLINT，该字段不能为空，默认值为“m”；



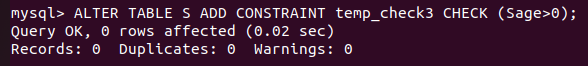
（4）向学生表添加“入学时间”列数据类型为CHAR（15）



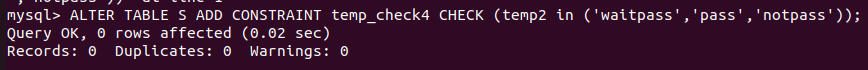
（5）修改某个表，添加类型为varchar(18)的字段temp，并添加check约束，要求该字段的长度为18，并且只能由数字组成，并且指定该字段为候选键；



（6）设置某个表中年龄的check约束为大于0；

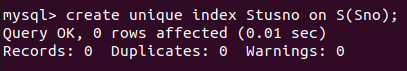


(7)设置temp2字段默认值为“waitpass”，并设置该字段的check约束为“notpass”、“pass”、“waitpass”；

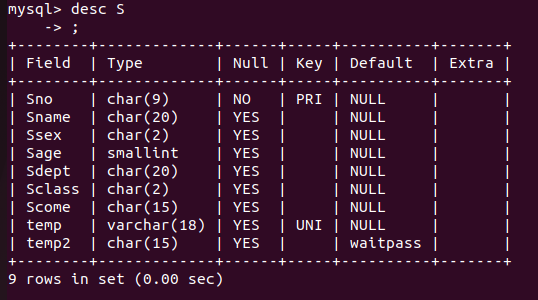


20230515110410

（8）在表S中创建索引；



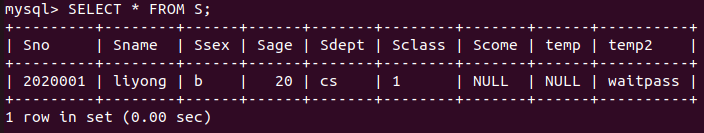
（9）使用describe命令展示表结构。



4. SQL DDL & DML使用

1. 使用insert语句将数据插入到相应的表中；

20230515111154



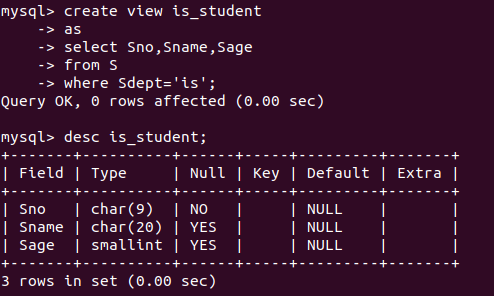
1. 使用delete语句删除表中有关联表外建对应的记录。不能正常删除，会发生如下图所示报错。因为删除内容被SC表所引用，所以不能够完成删除操作。

20230515112150

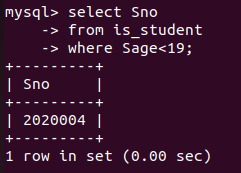
1. 使用update语句更新外键Sno数据，不能发生更改，会产生以下形式的报错，原因是S表中的Sno被SC表所引用，被引用的列不能够发生修改。

20230515112503

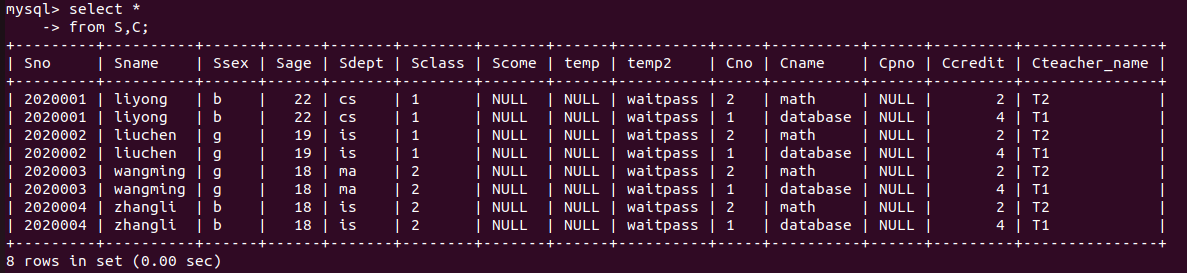
1. 创建视图；



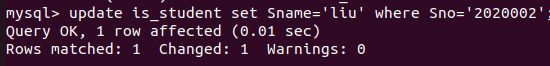
1. 对视图进行查询操作；

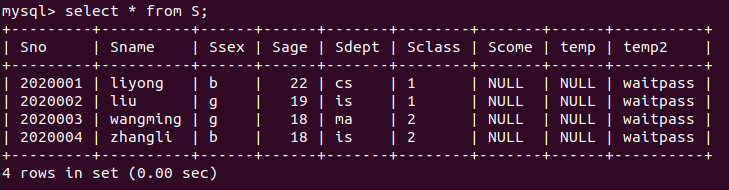


1. 对表进行联合查询操作；

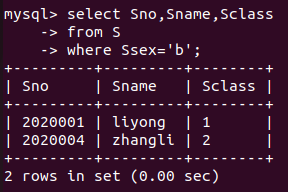


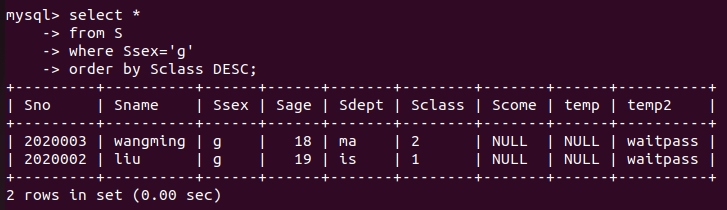
1. 更新视图，将学号为2020002的学生姓名更改为liu，可以操作成功，且S表中的数据也会发生更改。

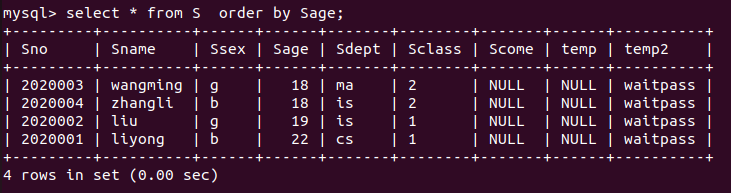


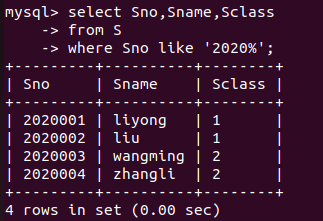


5. 简单查询

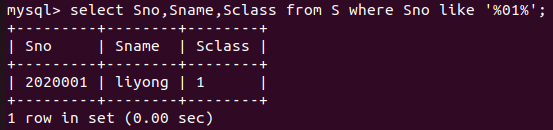
1. 查询性别为“男”的所有学生的学号、姓名和班级号；
2. 查询S表，获得性别为“女”的记录，结果按照班级ID降序排列；



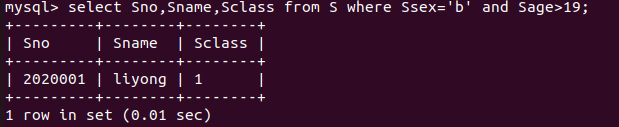
1. 查询S表，按照年龄从小到大排序；
2. 查询学号以“2020”开头的学生信息，字段包括学号、姓名、班级号；



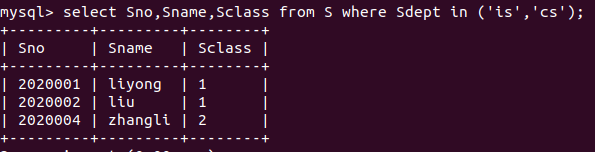
1. 查询学号中包含“01”的学生信息，字段包括学号、姓名、班级号；



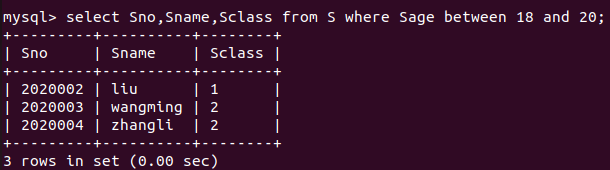
1. 查询性别为男性，且年龄大于19的学生的学号姓名和班级。



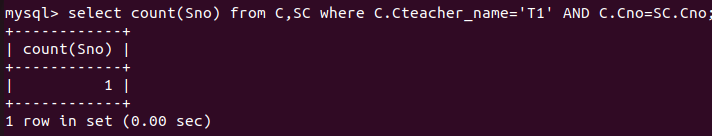
1. 查询年龄为cs和is系的学生的姓名学号和班级



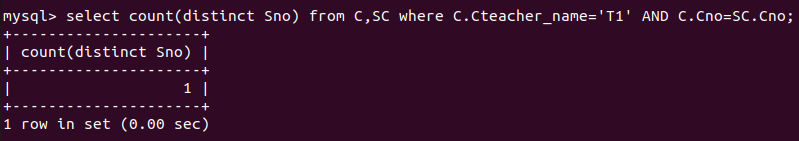
1. 查询年龄在18到20之间的学生学号。



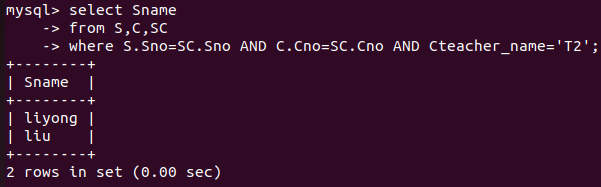
1. 查询T1老师所教课程的的选课人次（一人选两门课程，算两人次）；



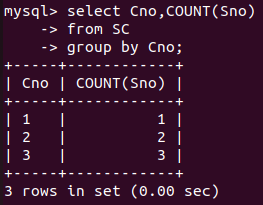
1. 查询XXX老师所教课程的的选课人数（一人选两门课程，算一人）；



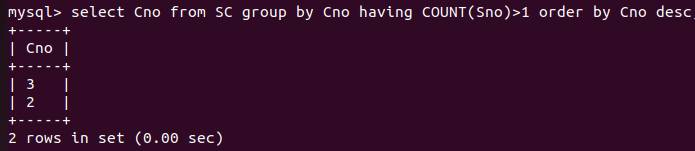
1. 查询选修T2老师课程的学生姓名；



1. 统计每门课的学生的个数（集合函数）；

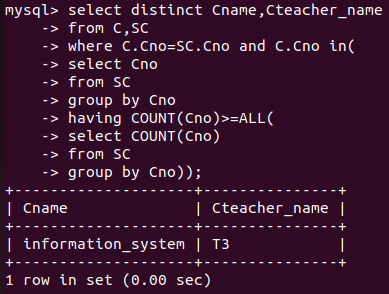


1. 查询选课人数超过1人的课程号，并按课程号降序排列

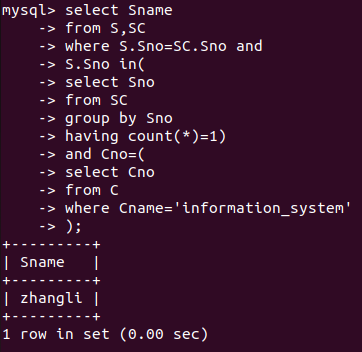


6. 复杂查询

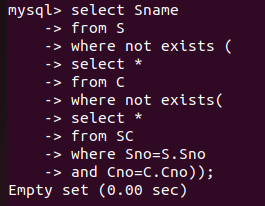
（1）查询选课人数最多的课程名称和教师姓名；



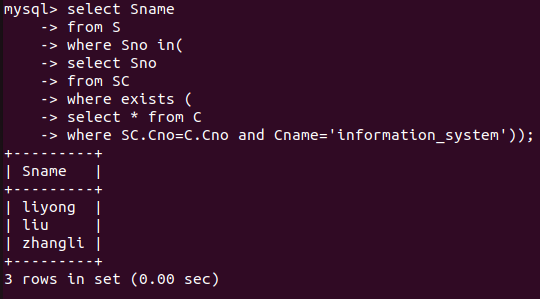
（2）查询只选了课程information\_system的学生姓名；



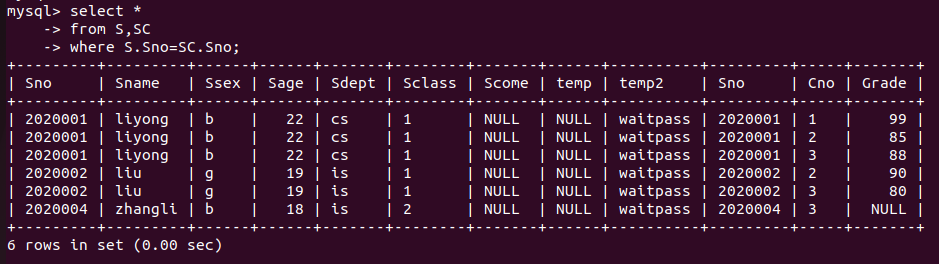
（3）查询选修了全部课程的学生姓名；



（4）查询选修了information\_system的学生姓名

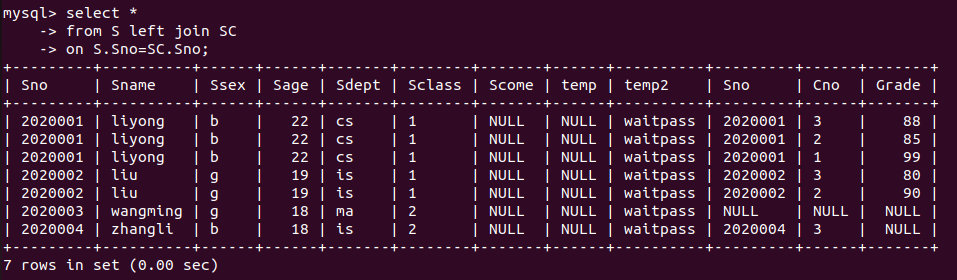


（5）据学生学号将S表与SC表做自然连接查询；



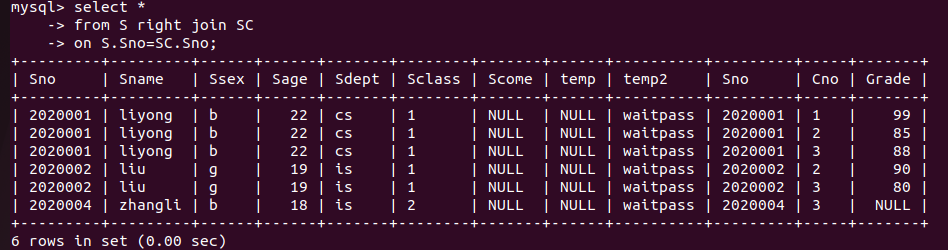
（6）据学号将S表与SC表进行左连接查询，并解释这样做所具有的业务含义；

返回包括S表中的所有记录和SC表中连接字段相等的记录。即显示所有的学生的选课信息。



（7）据学号将S表与SC表进行右连接查询，并解释这样做所具有的业务含义；

显示返回包括SC表中的所有记录和S表中连接字段相等的记录。即显示所有课程的选课学生信息。



（8）总结自然连接、左连接、右连接查询在产生的结果上面有什么区别。

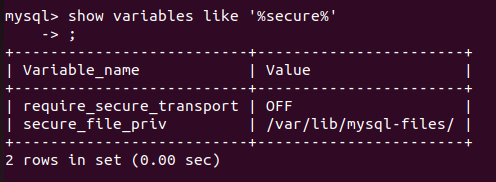
进行自然连接运算要求两个表有共同属性（列），自然连接运算的结果表是在参与操作的两个表的共同属性上进行等值连接后，再去除重复的属性后所得的新表。

右连接结果表中除了包括满足连接条件的行外，还包括右表的所有行。

左连接结果表中除了包括满足连接条件的行外，还包括右表的所有行。

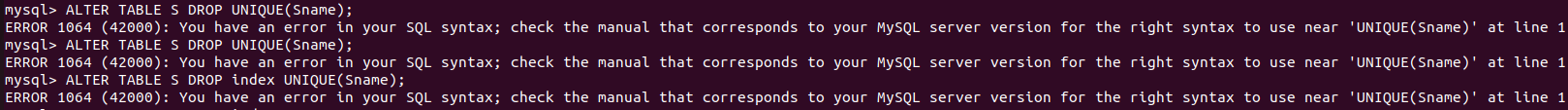
四、出现的报错及其解决方法

1.报错如下图所示20230513155415

报错原因：secure-file-priv变量的默认设置为/var/lib/mysql-files/所引起的，该值意味着只能导出到指定目录下（已验证），否则会报错，使用命令show variables like '%secure%'查看mysql数据库该变量值：

secure\_file\_priv=/var/lib/mysql-files/，限制mysqld的导入导出只能发生在/var/lib/mysql-files/目录下，所以修改导出文件的存储位置为/var/lib/mysql-files/country即可解决。

2.drop语句使用不当，使用课本中给出的语句并不能进行删除操作，将语句修改为ALTER TABLE S DROP index Sname;即可解决报错。



3.报错如下图所示20230515160607

报错原因为在进行多表连接时对于不同表中的相同表项无法区分，将产生错误的表象前加上“所在表.”即可解决报错。