2018380039\_Dikshya Kafle Lab report DBMS

**Name: Dikshya Kafle**

**Student Number: 2018380039**

**Experiment 3**

Data Integrity and Security

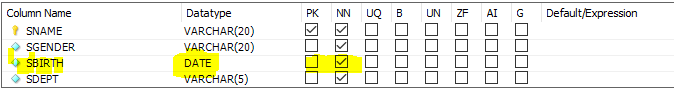
**Goal**

1. To practice how to define the data integrity.
2. To practice how to create users
3. To practice how to grant/revoke privileges of databases and tables.

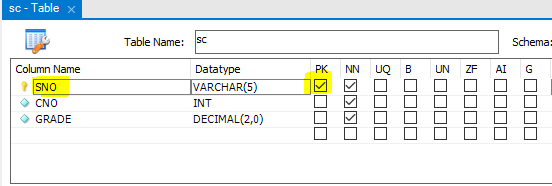
**Content**

1. Add the following constraint and index with GUI (for student database).
   * 1. **Not null：add the not null constraint to S(SBIRTH).**

To add the not null constraint to Classroom (capacity), we go to university->Tables->Right click on Classroom table -> Design. Then we just tick mark on the “Not null” corresponding to the **“SBIRTH”** column.

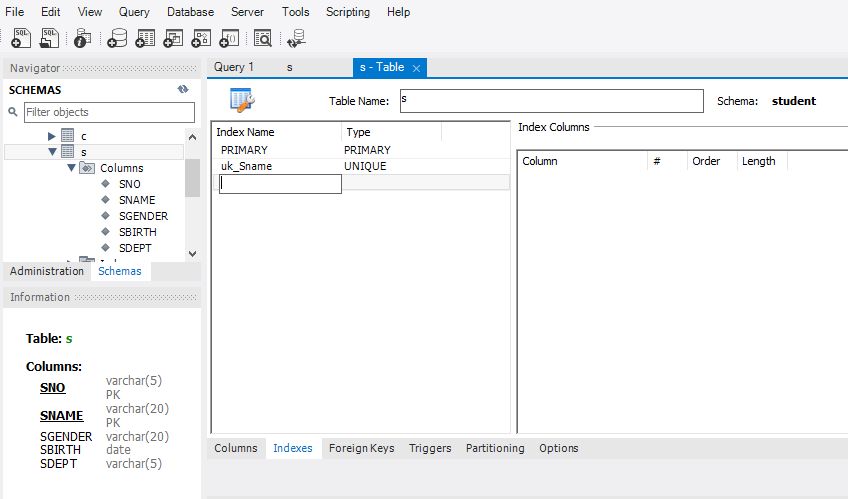


* + 1. **Primary key：set the SNO as the primary key.**



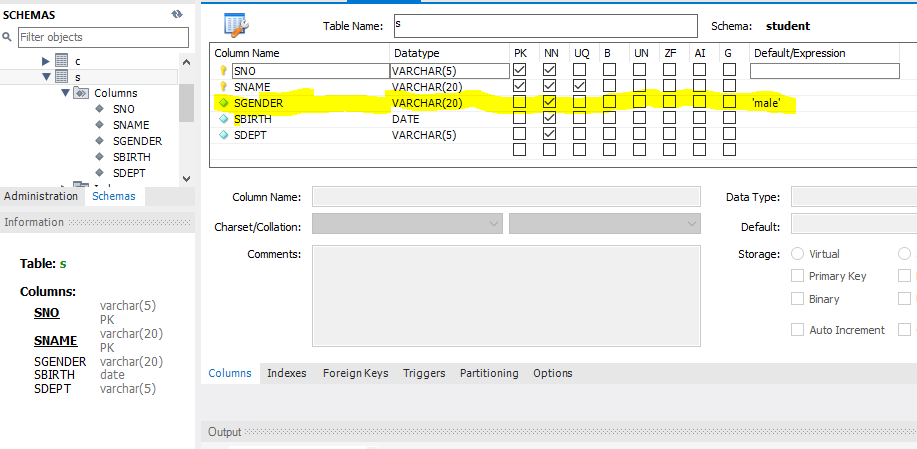
* + 1. **Unique constraint：add unique constraint for the primary key of SNAME, the constraint name is uk\_Sname.**

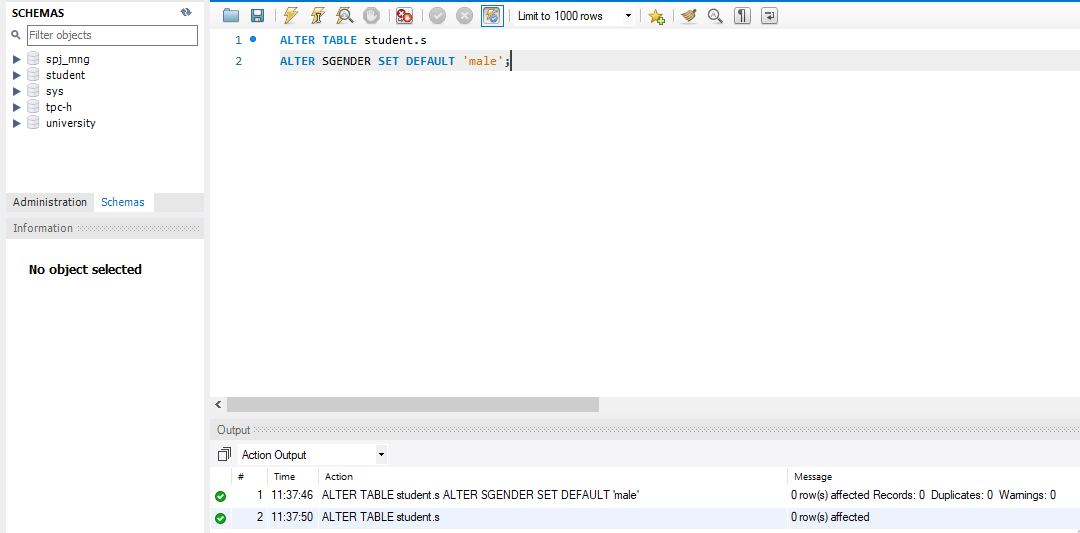
Go to “Indexe Name” in the Design window of the table Instructor. Click “Add”. In the grid under General, click Type and choose Unique Key from the drop-down list box to the right of the property. Choose the “sname” column, then set the name of the constraint as” uk\_sname” as shown below:



* + 1. **Default constraint：add the default value to S(SGENDER), the default value is “male”.**

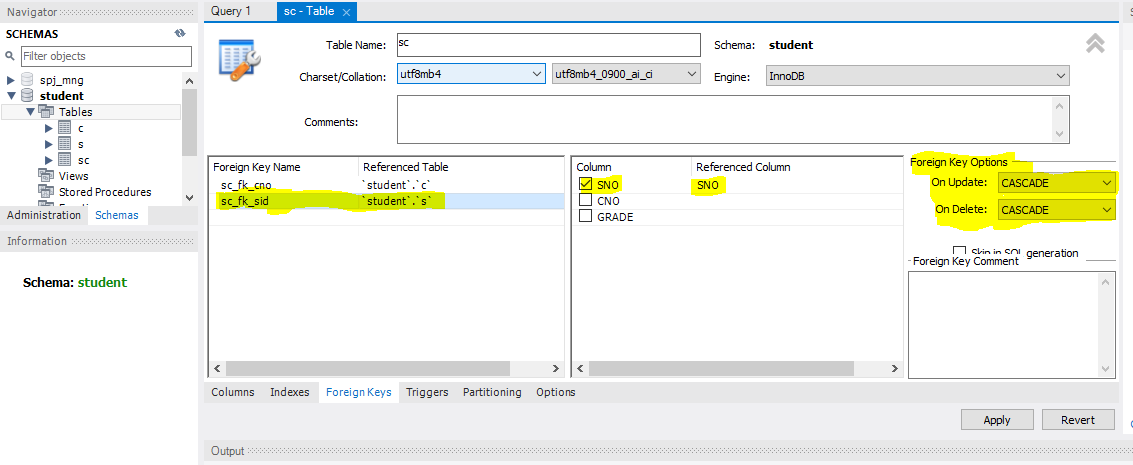
On the “Design” window of the s table, we click on the SGENDER attribute. A column properties window shows up at the bottom. We set the “Default /Expression ” property to male as shown below:

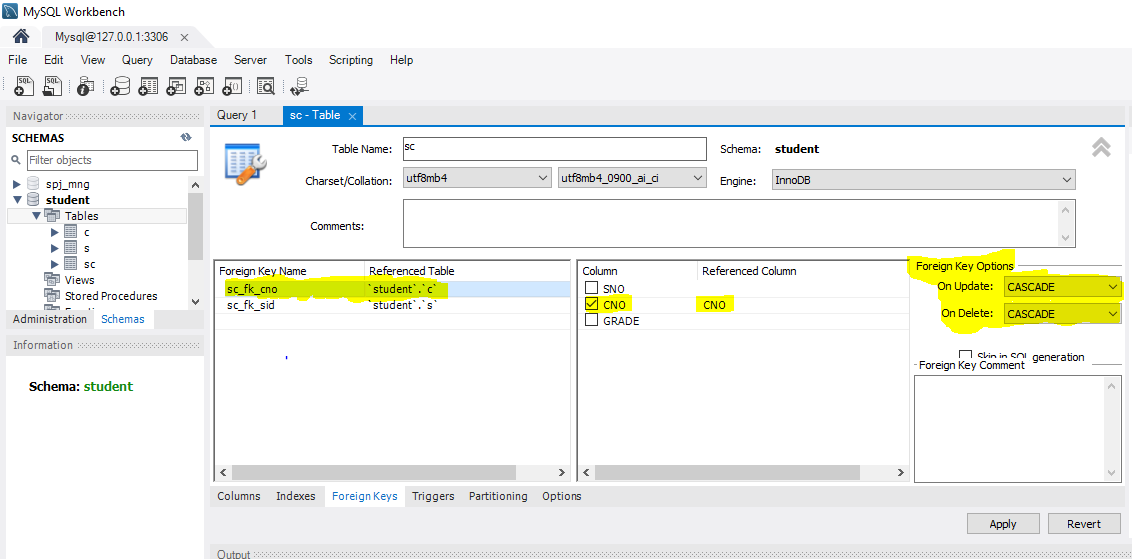




1. Add forgein key constraints to table **SC** in the database **Student** with GUI, set SNO(foreign key name is: sc\_fk\_sid) as a foreign key referencing table S, and set CNO as another foreign key referencing table ***,*** name it with sc\_fk\_ cno. Try and validate different strategies in violation of the foreign key constraints :

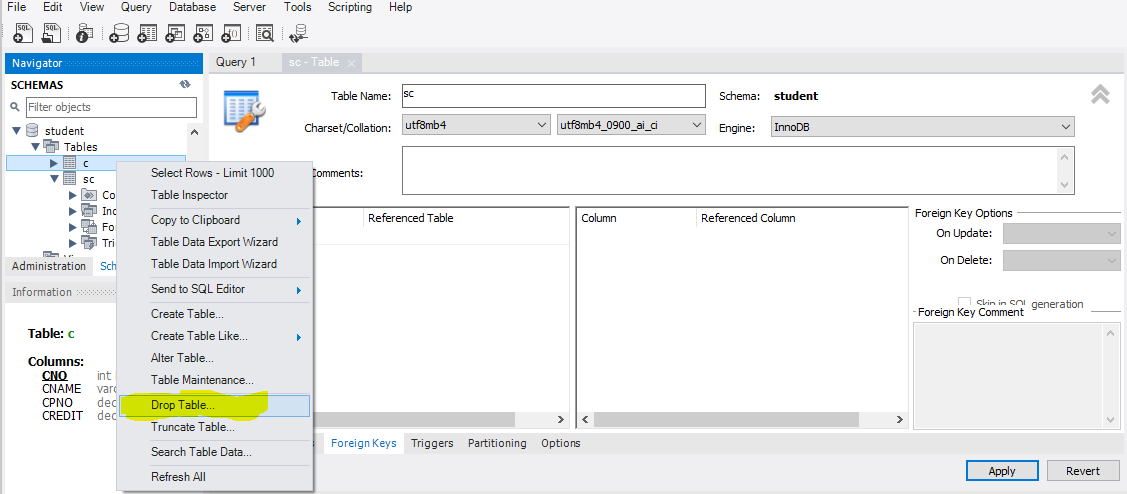
NO Action/restrict/cascade/set to null



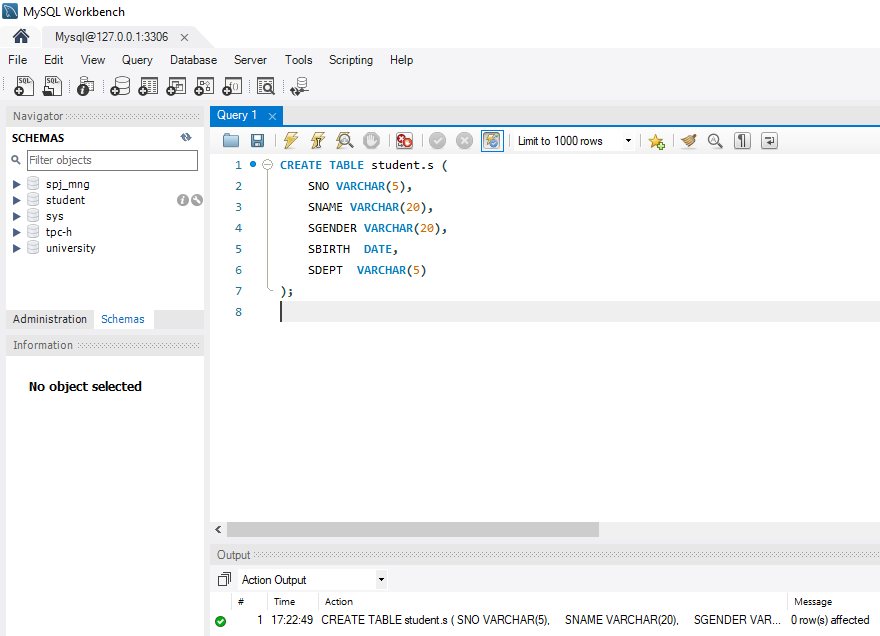


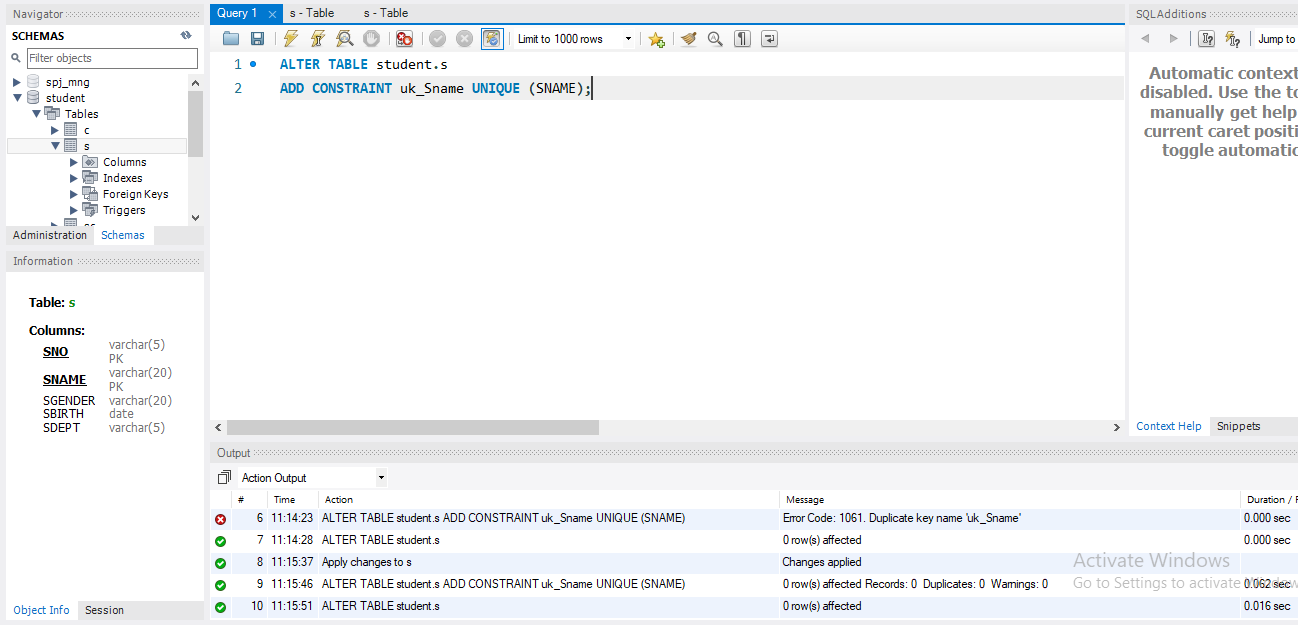
1. **Drop the three tables in database Student, and create some tables through SQL statements with the following constraints.**

By clicking the schema we get to tables we are able to choose each table at once as shown below in the picture and by choosing drop table we drop each table.

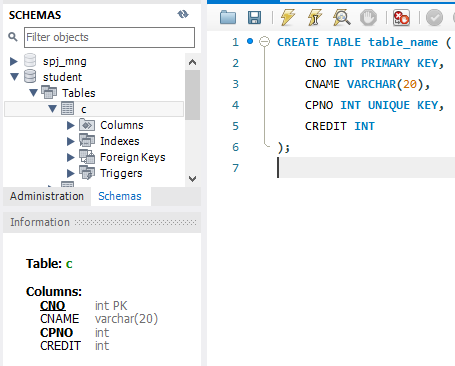


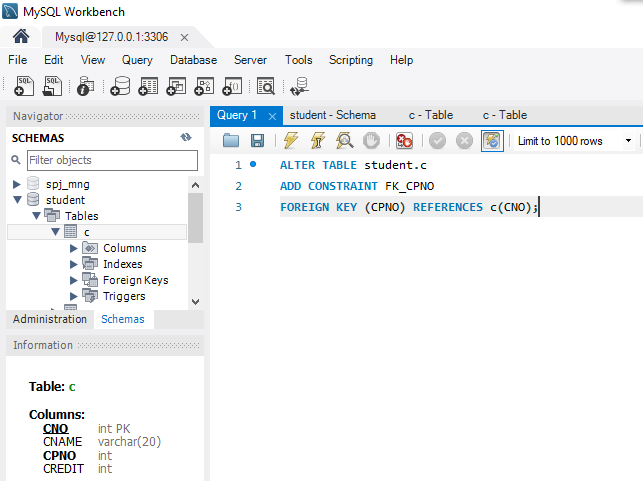
* Table ***S***：same to the constraints set in question 1(1).

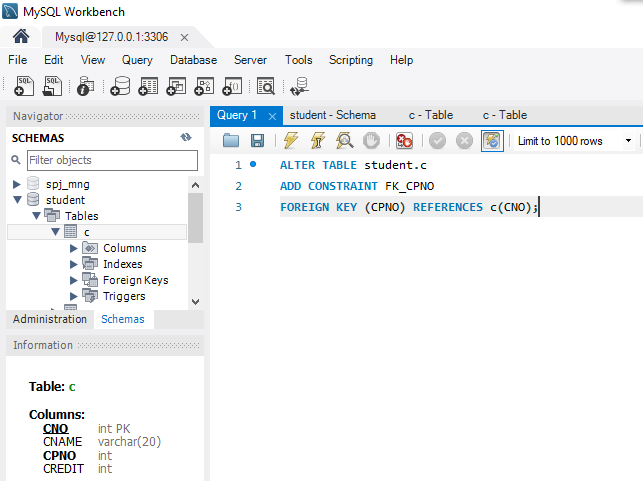




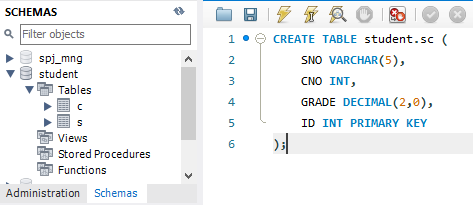
* Table ***C***: set CPNO as a foreign key，referencing table C with the attribute CNO

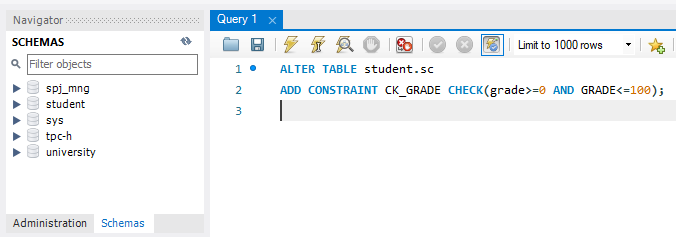


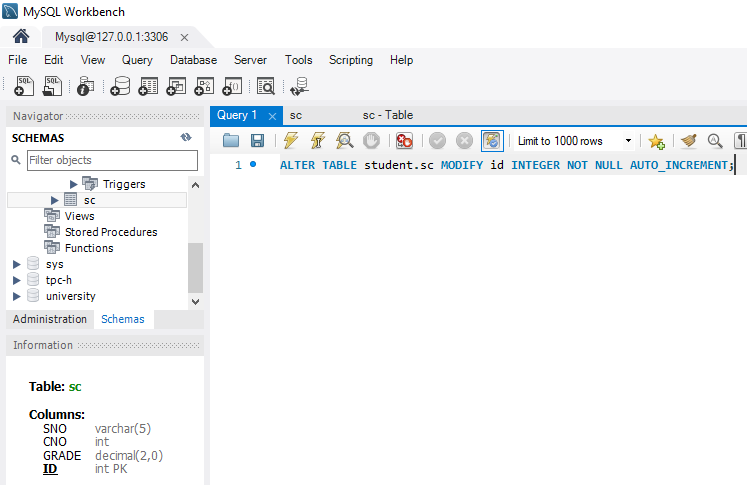




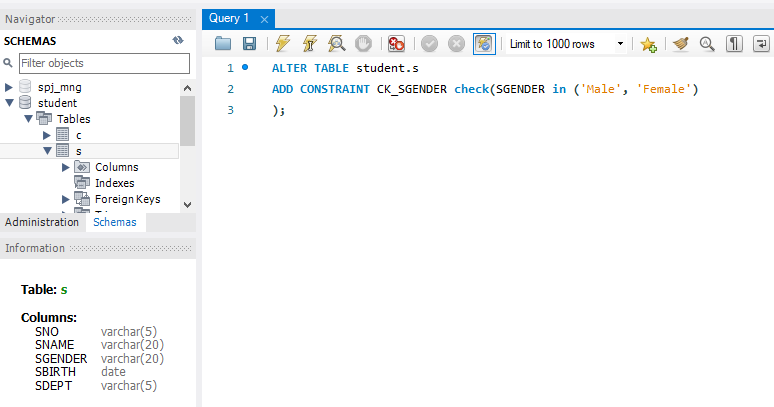
* Table **SC**：set the foreign key constraints same to question2, and set the valid range of attribute GRADE with [0,100]. In addition, add one attribute ID to table SC, and set it as a primary key, and it can increase automatically. When a new tuple is inserted to the table, its(ID) value will increase by +1.



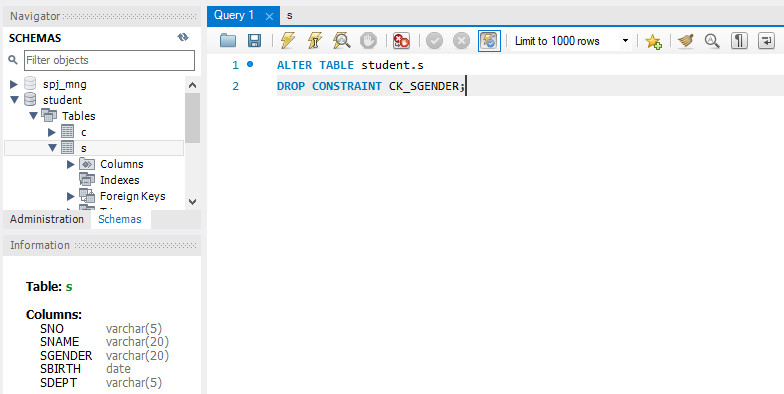




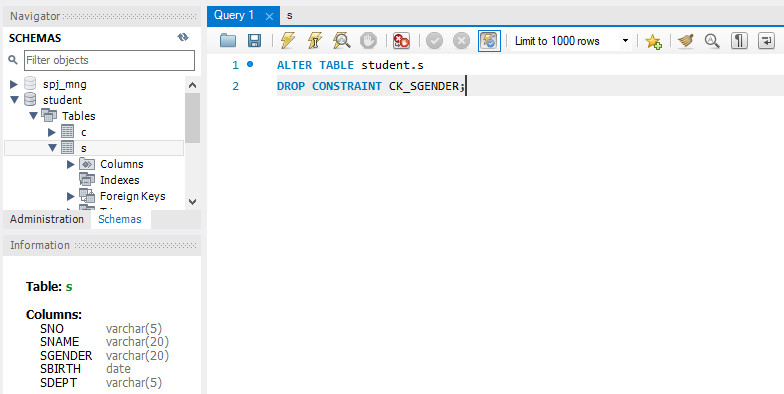
1. **Add or remove the following integrity constraints with SQL language.**
2. Add constraint: SGENDER’s value in table S can only be "male" or "female".



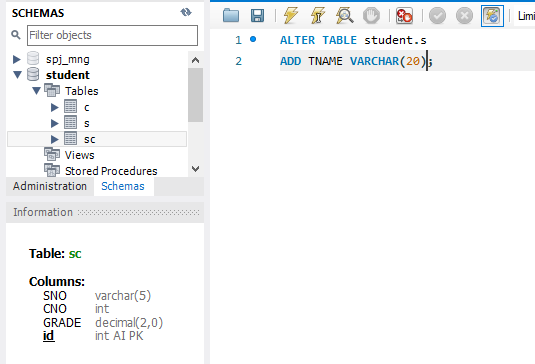
1. Delete the SGENDER value constraint created in table S.

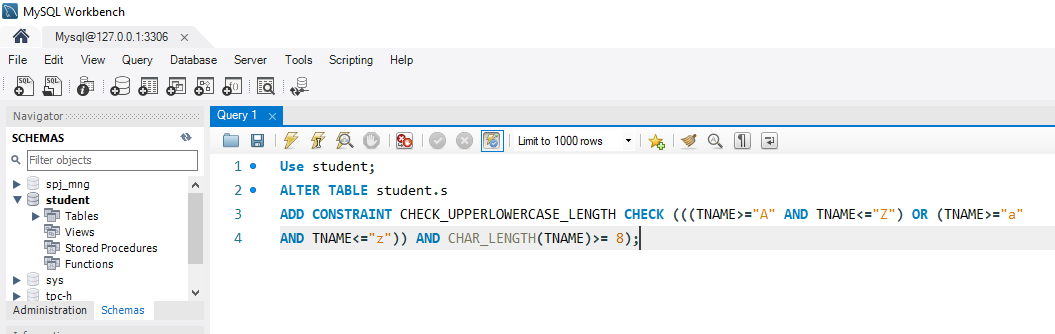


1. Remove the foreign key constraints from the SC table.



1. Add a new column TNAME (indicating the name of the student's tutor) in the student table S, and require that the tutor name must be all uppercase or lowercase letters, and the length cannot be less than 8 characters (hint: char can be used\_ Length() function and regular expression).

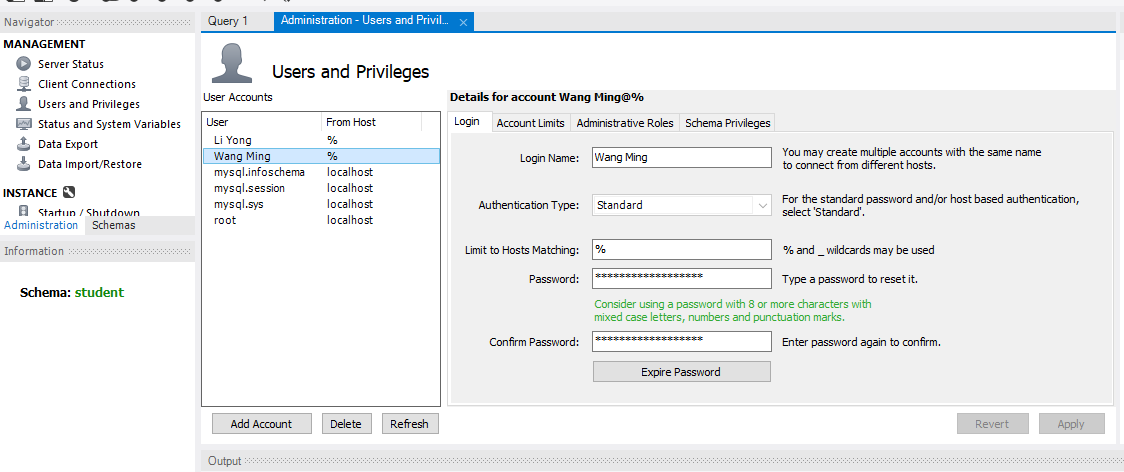


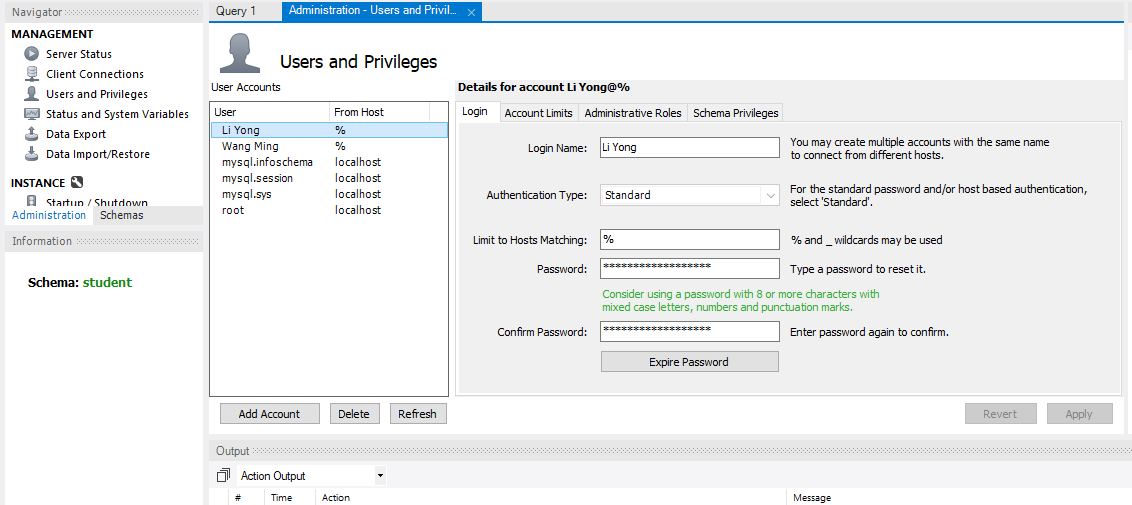


Try to finish the following requirements by connecting to the local database server

1. Create and authorize new users in GUI.
2. Create two users who can access the current student database: Wang Ming and Li Yong.

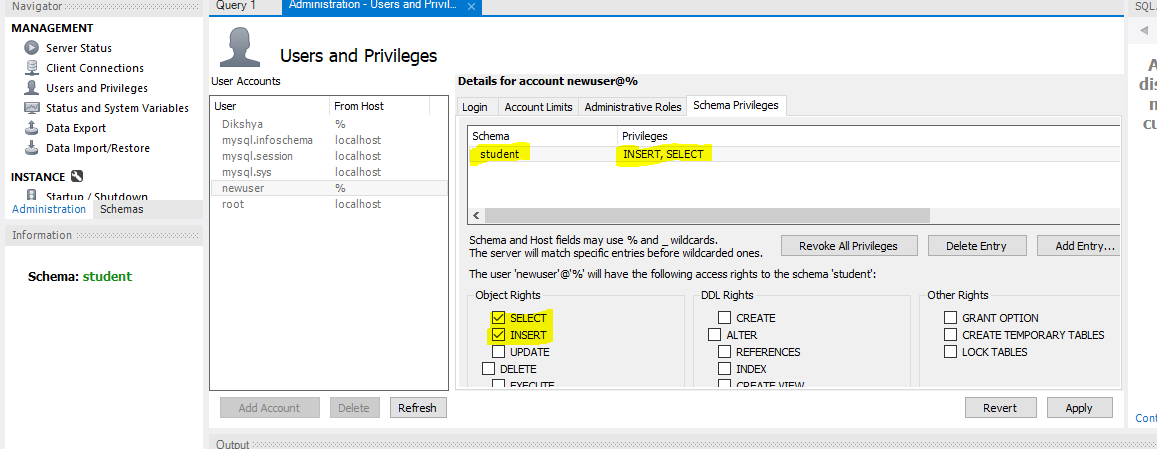
To create the new user we go to administration-user and privileges.

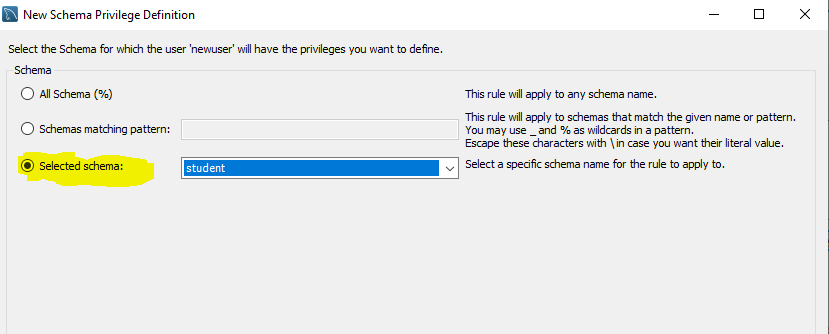


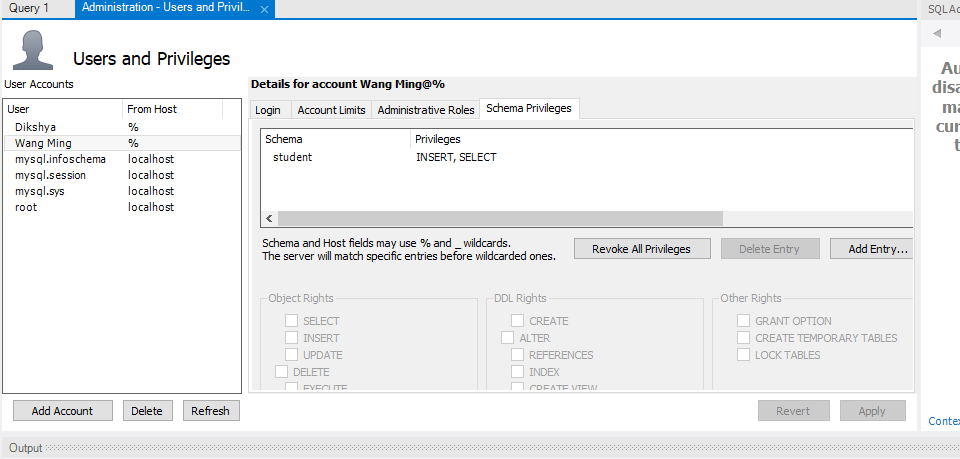


(2) Complete the following authorizing:

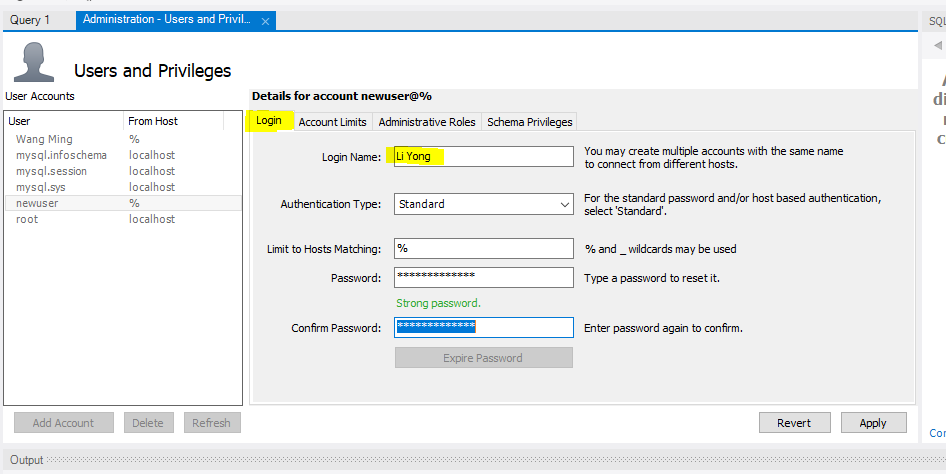
① User Wang Ming has the priviliges to select and insert all tables.

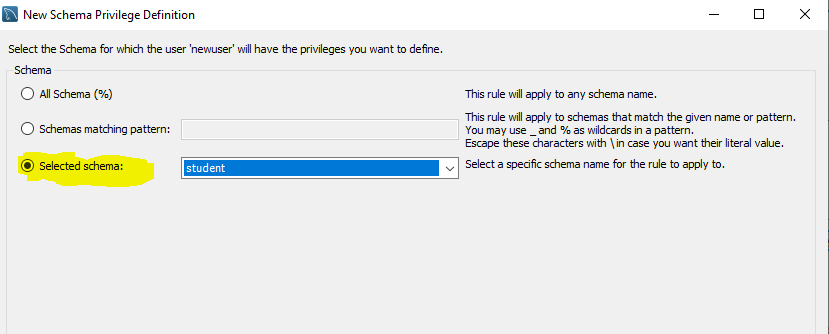


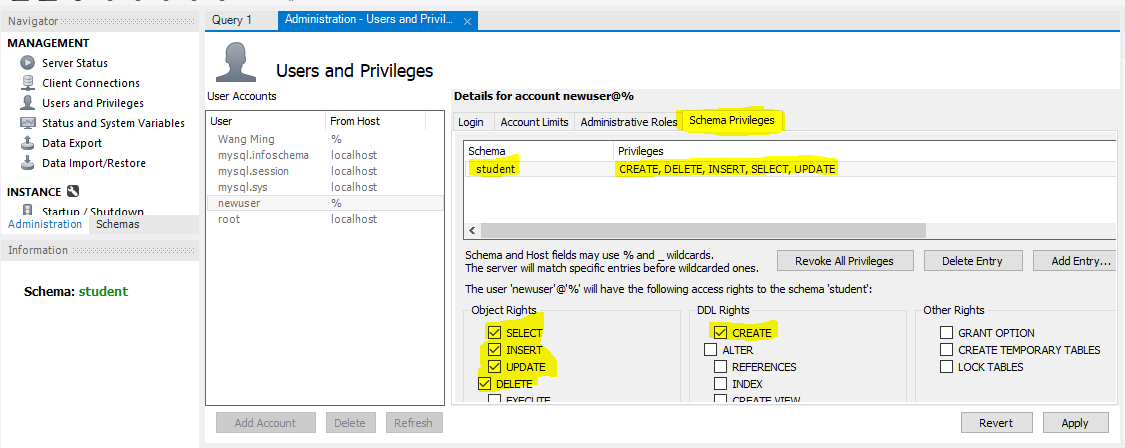


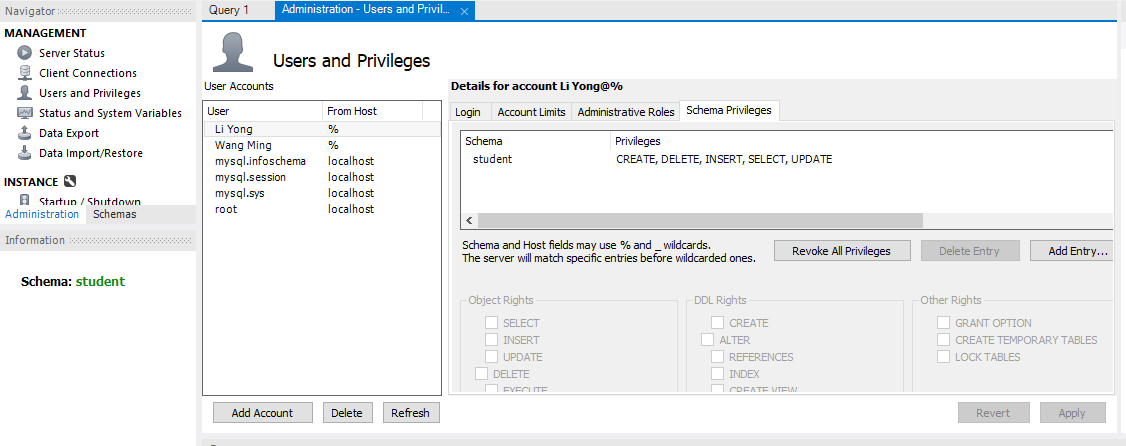


② User Li Yong has the priviliges of select, insert, delete, update and create on the database.





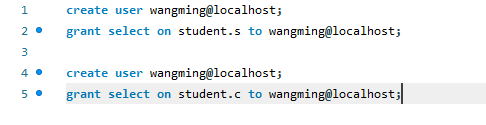




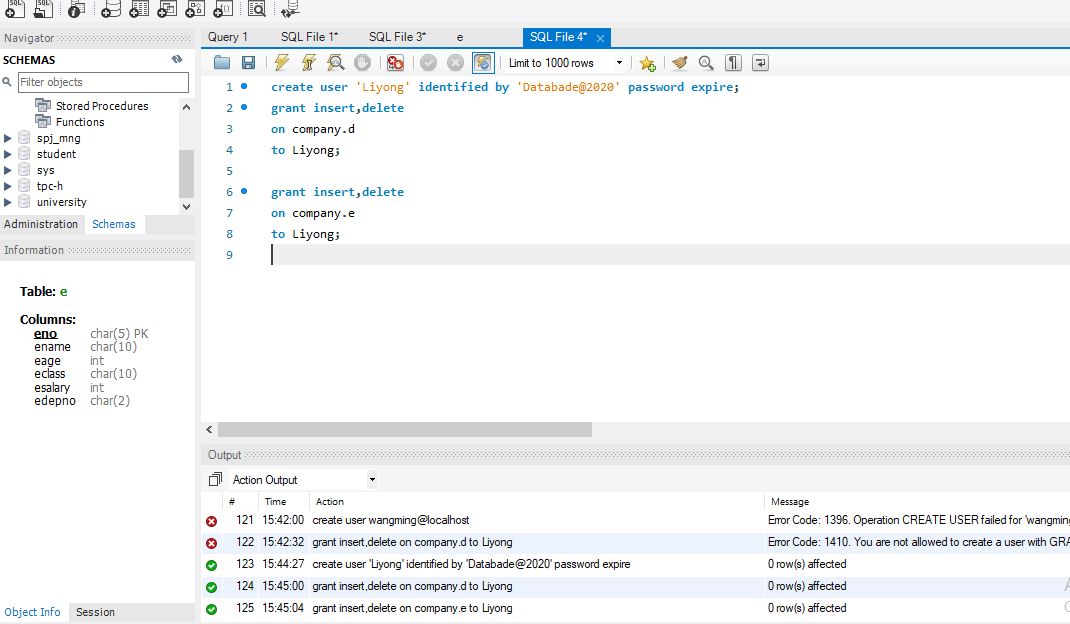
2. Use SQL statement to authorize and withdraw permissions and verify permissions.

Requirements: create relevant users and specified database tables, complete authorization and authority verification for each question, and then withdraw the authority and verify the authority.

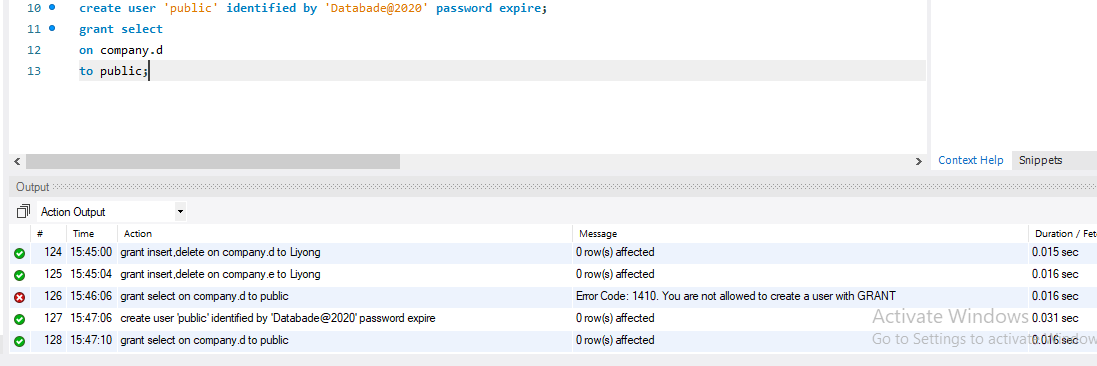
1. User Wang Ming has query privilege on two tables.



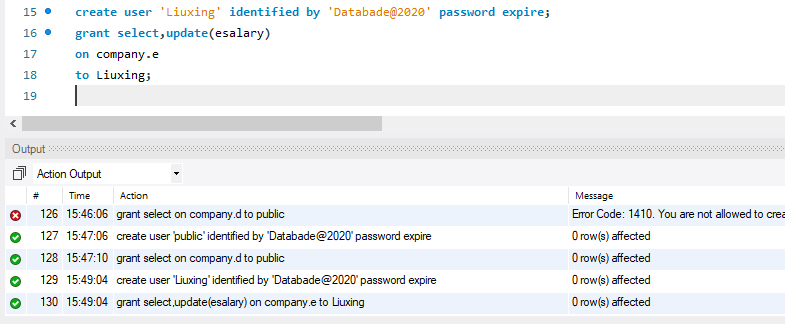
1. User Li Yong has insert and delete privileges on the two tables.



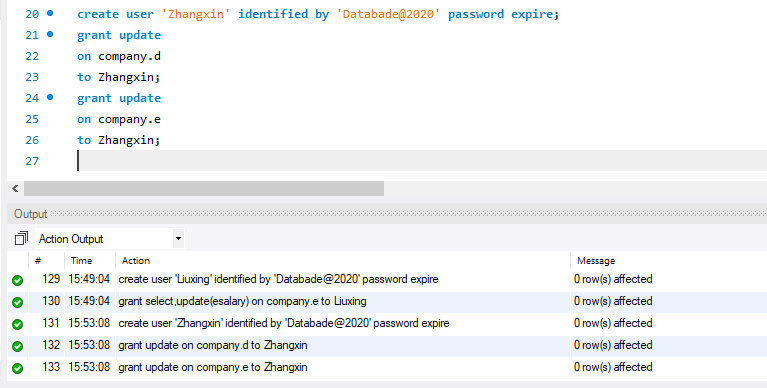
1. Each instructor only has the right to query his own record.



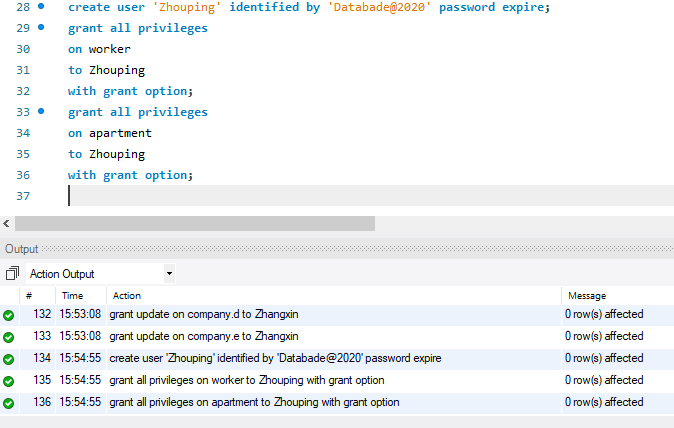
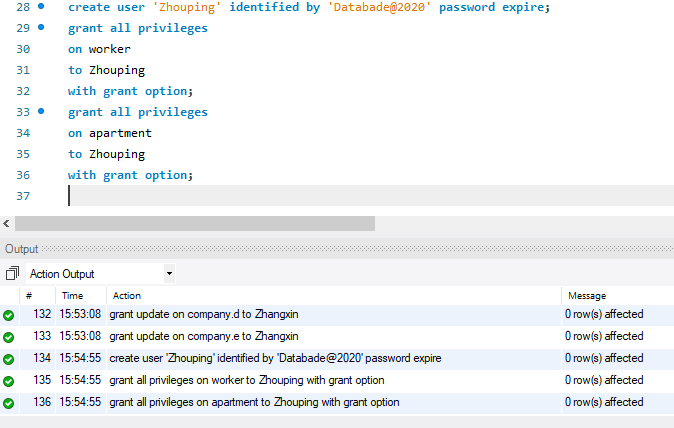
1. User Liu Xing has query privilege on employee table and update privilige on salary field.



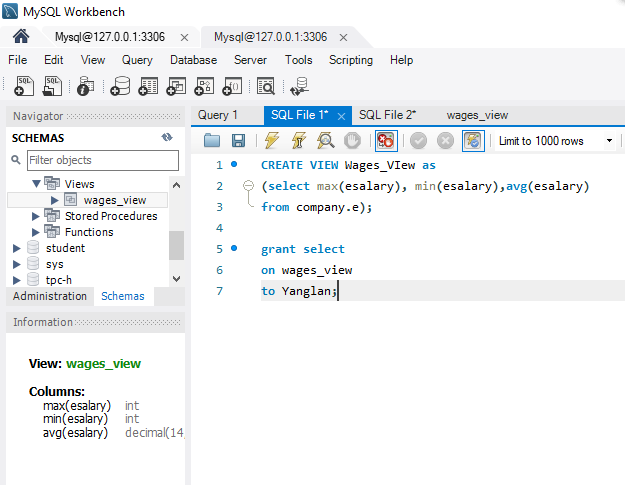
1. User Zhang Xin has the right to modify the structure of the two tables.

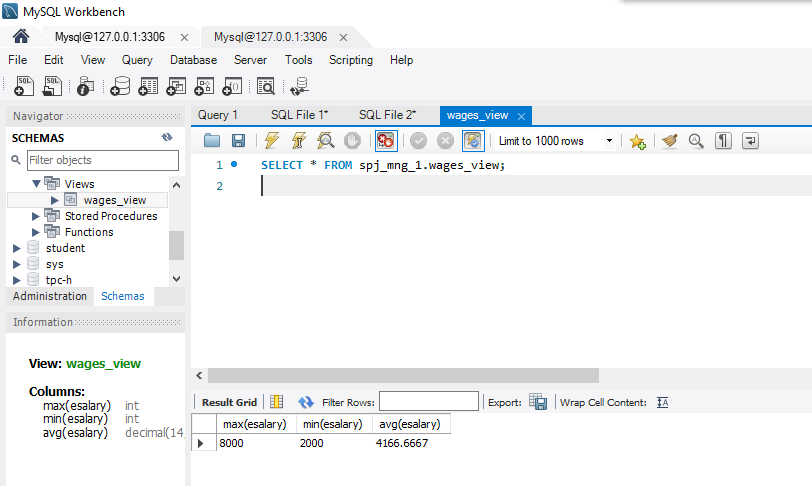


1. User Zhou Ping has all permissions on the two tables, and has the authority to authorize other users.



1. User Yang Lan has the privilige to query the maximum wage, minimum wage and average wage from each department employee, but he cannot view the salary of each person.





**Problems:**

For the view and index experiment, in particular, we faced a lot of technical failures. A basic code typing mistake, for example, prompted me to redo the whole database again. Even some of the questions were really difficult.

**Solutions:**

We were able to overcome most of our mistakes with the very helpful guidance and assistance of Professor. In order to identify our remaining errors that led to fatal errors, we also used the internet and our textbook, which also provided us with some SQL codes that we have not studied.

**Summary:**

In the first experiment, we worked on data integrity. We also learned how to add and drop multiple constraints using the GUI and SQL, such as primary key, foreign key, not null, unique and default values constraints. We knew how we could explain the view and the index. Then we moved to create logins and users and try granting VIEW permission to the newly created user.