2018380039\_Dikshya Kafle Lab report DBMS

Name: Dikshya Kafle

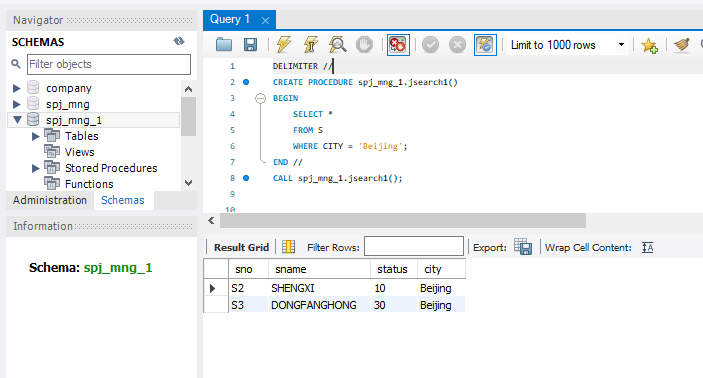
Student Number: 2018380039

**Experiment 5**

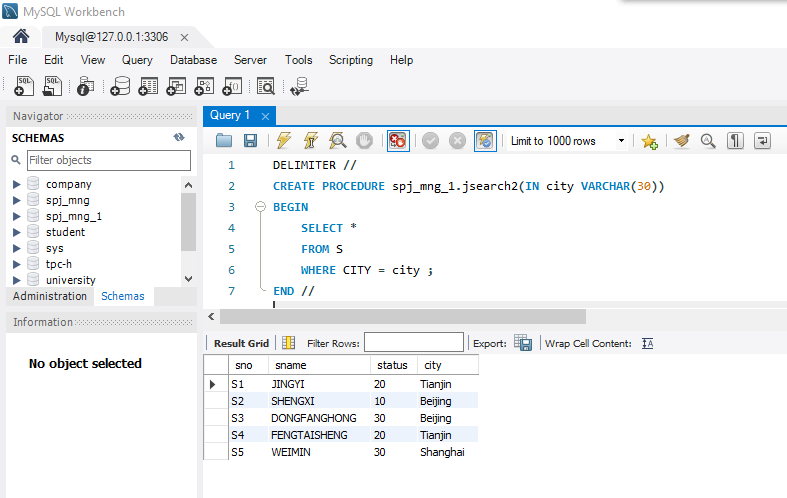
To practice how to create the trigger/procedure/function.

**Goal**

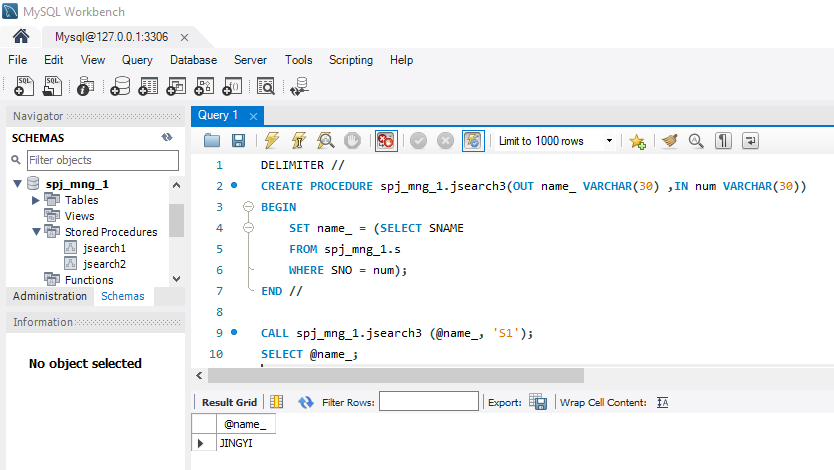
1. **For SPJ\_ MNG database, create and execute the following procedures.**
2. **Create a stored procedure with no parameters named jsearch1. The function of the procedure is: when the procedure is executed, it returns all the information of Beijing suppliers in table S. Call the procedure and verifies the results.**



1. Create a procedure with input parameters named jsearch2. The function of the procedure is: when you enter the city name of a supplier (such as Beijing), it will return all the information of the specific supplier. Call the procedure and verify the results.



1. Create a procedure (or a function) jsearch3 with input and output parameters. The function of this procedure/funciton is: when a supplier number (input parameter SNO) is entered, the name of the supplier (output parameter SNAME) will be returned. Call the procedure/function and verify the results.



1. Create a procedure jsearch4 using cursors, call the procedure and verify the result after. The function of the procedure is: when a project number JNO is input, the names (SNAMEs) of all suppliers supplying the engineering parts will be returned. These supplier names are spliced into a string and separated by comma ','.

For example: input: J2, output: ‘JINGYI, SHENGXIN, WEIMIN'. (10 points)

1. View the text information of the procedures jsearch1 and jsearch2. (5 points)

Hint:use the command‘show create procedure jsearch1’. For convenience of viewing, you can add \G at the end of the above command and display it in the form of two columns of text.

1. View the basic status information of the procedures jsearch1 and jsearch2. (5 points)

Hint: Show procedure status like 'jsearch%';

1. Delete the procedure jsearch1. (5 points)
2. Create and execute the following triggers for student database: (40 points in total)
3. Delete the foreign key constraint on the SC table and create INSERT trigger named insert\_s for the table SC. The function of the trigger is: when the user inserts a record into the SC table, if the inserted CNO value is not the existing value of CNO in table C, the user will be prompted a message "the data not in table C cannot be inserted" and the insertion of the data will be prevented; if the inserted SNO value is not an existing value of SNO in table S, the user will be prompted that "the data not in table S cannot be inserted" and the insertion of the data will be prevented. After the trigger created successfully, insert some records into the SC table to verify whether the trigger works normally. (5 points)
4. Create a DELETE trigger named dele \_ s1 for table S. The function of the trigger is: to prompt the user a message of "the data in this table cannot be deleted" and prevent the user from deleting the data in table S. After the trigger is created successfully, delete some record in table S to verify whether the trigger works normally. (5 points)
5. Create a DELETE trigger named dele \_ s2 for the table S, the function of the trigger is to delete the his course selection record in SC table when deleting a student’s record in S table. After the trigger is created successfully, delete some records in the table S and verify whether the trigger works normally (confirm whether the relevant data of S table and SC table are both deleted). (5 points)
6. Create an UPDATE trigger for table S named update\_s, the function of this trigger is to prohibit updating the contents of the column "sdept" in table S (make the updating be failed and prompt a message of "the column of ‘sdept’ cannot be updated). After the trigger is created successfully , update the content of the "sdept" column in the table S to verify whether the trigger works normally. (5 points)
7. Delete update\_s trigger. (5 points)
8. Design a before update trigger and after update trigger, and compare the difference between the two triggers. (5 points)
9. Create a new curriculum score statisticstable CAVGGRADE (CNO, snum, examsnum, avggrade), for the columns are indicating the course number, the number of students who choose the course, the number of students taking the exam, and the average score of the course respectively. Use a trigger to achieve the following functions: when insert, delete or update a person's score of SC table, automatically update the records in table CAVGGRADE correspondingly. Note that if the grade of a student in SC table is NULL, it indicates that the student has not taken the exam, and it is not necessary to calculate the average score. However, when grade is 0, that is, the score of the student is 0, theaverage grade needs to be calculated. (10 points)

Hint: in mysql, you need to create three triggers for insert, update and delete actions. You can first design and implement a common procedure, and then call the stored procedure in the 3 triggers.

1. Create an table employee (EID, ename, salary) for the employees, assuming there are 1000 employee data in the table, complete the following requirements. (total 20 points, 10 points for each question)
2. In order to automatically generate 1000 employee data, a user-defined function GENERATEEID is created to automatically generate employee ID. The employee ID is required to be an 8-digit number. The higher four digits indicate the current year in which the employee data is inserted, and the last four digits increase in the order from 0001 to 9999. For example, the first record inserted in the year of 2015 is 20150001, and all 1000 employee IDs are 20150001-20151000. Call this function to insert 1000 pieces of data automatically. (Note:when inserting data that the employee name can be any value , and the value of salary is between 2000 and 5000)
3. The company plans to increase the salary of each employee according to certain rules. Please use cursor to create a procedure and execute the procedure to complete the salary adjustment. The salary increase rules are as follows:

* The salary below 3000 yuan, with a monthly increase of 300 yuan;
* The salary between 3000-4000 yuan, with a monthly increase of 200 yuan;
* The salary more than or equal to 4000 yuan, with a monthly increase of 50 yuan;

**Content**