

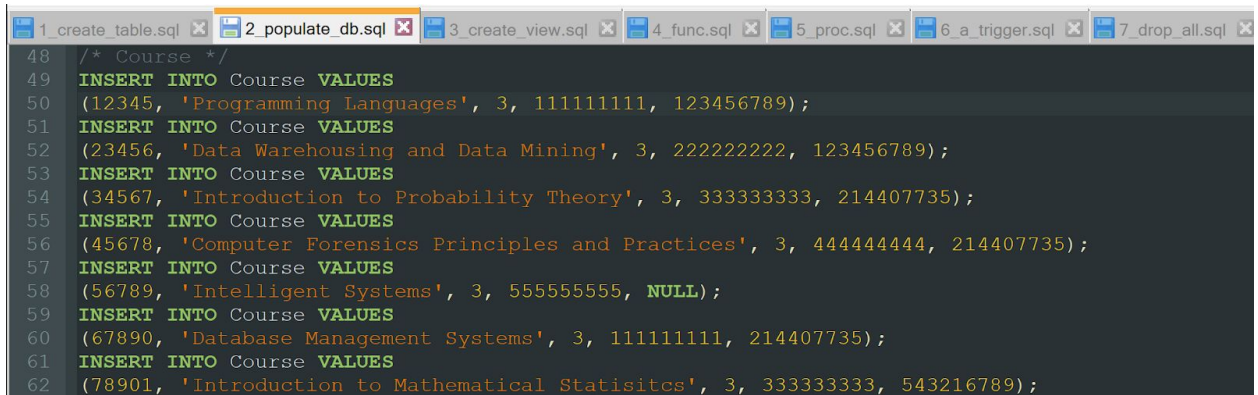
Homework Assignment 4: result

Overview:

- Design a Java Program to practice the use of JDBC to access MySQL database. In your program, you need to implement the following:
 - Display all the students (ssn, student name) who enrolled in the courses of “Programming Languages”.
 - Query “TA_course” view (created in homework2) to display all the tuples in that view.
 - Call the “Course_Instructor” function (created in homework2). Given course name “Programming Languages”, display the instructor name. You need to insert data into the database before you call this function, such that the function can return at least one result.

Before running any of the following code from Java or MYSQL we first need to include a ‘Programming Languages’ course into the database. After, we need to create the database and include the JDBC Jar file is in the classpath of our Java IDE:

- Step 1:** Include ‘Programming Languages’ into .sql file.



```
1_create_table.sql 2_populate_db.sql 3_create_view.sql 4_func.sql 5_proc.sql 6_a_trigger.sql 7_drop_all.sql
48 /* Course */
49 INSERT INTO Course VALUES
50 (12345, 'Programming Languages', 3, 111111111, 123456789);
51 INSERT INTO Course VALUES
52 (23456, 'Data Warehousing and Data Mining', 3, 222222222, 123456789);
53 INSERT INTO Course VALUES
54 (34567, 'Introduction to Probability Theory', 3, 333333333, 214407735);
55 INSERT INTO Course VALUES
56 (45678, 'Computer Forensics Principles and Practices', 3, 444444444, 214407735);
57 INSERT INTO Course VALUES
58 (56789, 'Intelligent Systems', 3, 555555555, NULL);
59 INSERT INTO Course VALUES
60 (67890, 'Database Management Systems', 3, 111111111, 214407735);
61 INSERT INTO Course VALUES
62 (78901, 'Introduction to Mathematical Statistics', 3, 333333333, 543216789);
```

- Step 2:** Create Table.

```
mysql> source CSC174/hw3/1_create_table.sql
Query OK, 0 rows affected (0.02 sec)
```

- Step 3:** Populate.

```
mysql> source CSC174/hw3/2_populate_db.sql
Query OK, 1 row affected (0.00 sec)
```

- Step 4:** Create views.

```
mysql> source CSC174/hw3/4_func.sql
Query OK, 0 rows affected (0.00 sec)
```

- Step 5:** Create functions.

```
mysql> source CSC174/hw3/3_create_view.sql
Query OK, 0 rows affected (0.00 sec)
```

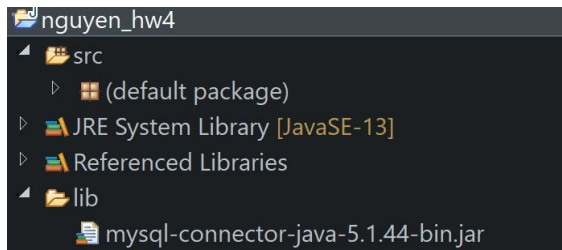
- Step 6:** Create procedures.

```
mysql> source CSC174/hw3/5_proc.sql
Query OK, 0 rows affected (0.00 sec)
```

- **Step 7:** Create trigger **Note this isn't needed, but I will use it for validation**.

```
mysql> source CSC174/hw3/6_a_trigger.sql
Query OK, 0 rows affected (0.01 sec)
```

- **Step 8:** Add the .jar file to the java classpath.



Now that the prerequisites are complete we can test the results of our JDBC:

Task 1: Display all the students (ssn, student name) who enrolled in the courses of “Programming Languages”.

- Running the following lines of code from `nguyen_hw4.java` results in the following:

```
nguyen_hw4.java
8 import java.sql.*;
9
10 public class nguyen_hw4 {
11
12     public static void main (String args[]) {
13         String url= "jdbc:mysql://athena.ecs.csus.edu/cs174119";
14         String username= "cs174119";
15         String password = "hdhmqwdk";
16
17         nguyen_hw4 example = new nguyen_hw4();
18         example.displayData(url,username,password);
19     }
20
21     public void displayData(String urlStr,String username,String password) {
22
23         try {
24             Class.forName ("com.mysql.jdbc.Driver").getDeclaredConstructor().newInstance();
25
26             Connection conn = DriverManager.getConnection(urlStr,username,password);
27             System.out.println ("Connected to the MySQL database");
28
29
30             Statement stmt1 = conn.createStatement();
31             ResultSet rs1=null;
32             rs1=stmt1.executeQuery("SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages'");
33             while (rs1.next()) {
34                 System.out.println("SSN: " + rs1.getString(1) + ", Name: " + rs1.getString(2));
35             }
36         }
37     }
38 }
```

```
<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 7:22:39 PM)
Connected to the MySQL database
SSN: 123456789, Name: Moe Ron
SSN: 214407735, Name: Seymour Butz
SSN: 678954321, Name: Hugh Jass
Disconnected
```

Figure 1: running java for student info enrolled in “Programming Languages”.

- To compare to MySQL run this: Because its **SELECT *** it includes all the info but string 1 and 2 are the **SSN and StudentName**.

```
mysql> SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages' AND C.CourseNo=E.CourseNo AND E.SSN = S.SSN;
```

| InstructorID | SSN | StudentName | Address | Email | SSN | CourseNo | Grade | CourseNo | CourseName | NoOfStudents |
|--------------|-----------|--------------|---|----------------------|-----------|----------|-------|----------|-----------------------|--------------|
| 123456789 | 111111111 | Moe Ron | 16514 Golden Valley Pkwy, Lathrop, CA 95330 | moeron@fake.com | 123456789 | 12345 | C | 12345 | Programming Languages | 3 |
| 214407735 | 111111111 | Seymour Butz | 2727 W March Ln, Stockton, CA 95219 | seymourbutz@fake.com | 214407735 | 12345 | A | 12345 | Programming Languages | 3 |
| 678954321 | 111111111 | Hugh Jass | 4600 Madison Ave, Sacramento, CA 95841 | hughjass@fake.com | 678954321 | 12345 | F | 12345 | Programming Languages | 3 |

Figure 2: running the same sql query in java on mysql.

- To validate let's enroll the student Ivana Tinkle to the "Programming Languages" course. To do so we run 'INSERT INTO Enrolled VALUES (789012345, 12345, 'B');'. Then run the above commands to verify.
 - Note:** you can insert through MySQL or through JDBC. In my case, I chose MySQL to show the change in NoOfStudents.

```
mysql> INSERT INTO Enrolled VALUES (789012345, 12345, 'B');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages' AND C.CourseNo=E.CourseNo AND E.SSN = S.SSN;
```

| InstructorID | SSN | StudentName | Address | Email | SSN | CourseNo | Grade | CourseNo | CourseName | NoOfStudents |
|--------------|-----------|--------------|---|----------------------|-----------|----------|-------|----------|-----------------------|--------------|
| 123456789 | 111111111 | Moe Ron | 16514 Golden Valley Pkwy, Lathrop, CA 95330 | moeron@fake.com | 123456789 | 12345 | C | 12345 | Programming Languages | 4 |
| 214407735 | 111111111 | Seymour Butz | 2727 W March Ln, Stockton, CA 95219 | seymourbutz@fake.com | 214407735 | 12345 | A | 12345 | Programming Languages | 4 |
| 678954321 | 111111111 | Hugh Jass | 4600 Madison Ave, Sacramento, CA 95841 | hughjass@fake.com | 678954321 | 12345 | F | 12345 | Programming Languages | 4 |
| 789012345 | 111111111 | Ivana Tinkle | 2475 Sunrise Blvd, Rancho Cordova, CA 95670 | ivanatinkle@fake.com | 789012345 | 12345 | B | 12345 | Programming Languages | 4 |

4 rows in set (0.00 sec)

Figure 3: Ivana Tinkle added "Programming Languages", NoOfStudents updated to 4.

```
nguyen_hw4.java
8 import java.sql.*;
9
10 public class nguyen_hw4 {
11
12     public static void main (String args[]) {
13         String url= "jdbc:mysql://athena.ecs.csus.edu/cs174119";
14         String username= "cs174119";
15         String password = "hdmqwdk";
16
17         nguyen_hw4 example = new nguyen_hw4();
18         example.displayData(url,username,password);
19     }
20
21     public void displayData(String urlStr,String username,String password) {
22
23         try {
24             Class.forName ("com.mysql.jdbc.Driver").getDeclaredConstructor().newInstance();
25
26             Connection conn = DriverManager.getConnection(urlStr,username,password);
27             System.out.println ("Connected to the MySQL database");
28
29
30             Statement stmt1 = conn.createStatement();
31             ResultSet rs1=null;
32             rs1=stmt1.executeQuery("SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages'");
33             while (rs1.next()) {
34                 System.out.println("SSN: " + rs1.getString(1) + ", Name: " + rs1.getString(2));
35             }
36         }
37     }
38 }
```

```
<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 7:34:28 PM)
Connected to the MySQL database
SSN: 123456789, Name: Moe Ron
SSN: 214407735, Name: Seymour Butz
SSN: 678954321, Name: Hugh Jass
SSN: 789012345, Name: Ivana Tinkle
Disconnected
```

Figure 4: the insertion of Ivana is present in JDBC.

- For the sake of showing the functionality of the trigger, we will remove Hugh Jass due to his letter grade being an 'F'. To do so run the code 'DELETE FROM Enrolled WHERE SSN=678954321 AND CourseNo=12345;'. Then run code to view the changes.


```
mysql> SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages' AND C.CourseNo=E.CourseNo AND E.SSN = S.SSN;
```

| SSN | StudentName | Address | Email | SSN | CourseNo | Grade | CourseNo | CourseName | NoOfStudents |
|-----------|--------------|---|----------------------|-----------|----------|-------|----------|-----------------------|--------------|
| 123456789 | Moe Ron | 16514 Golden Valley Pkwy, Lathrop, CA 95330 | moeron@fake.com | 123456789 | 12345 | C | 12345 | Programming Languages | 4 |
| 214407735 | Seymour Butz | 2727 W March Ln, Stockton, CA 95219 | seymourbutz@fake.com | 214407735 | 12345 | A | 12345 | Programming Languages | 4 |
| 678954321 | Hugh Jass | 4600 Madison Ave, Sacramento, CA 95841 | hughjass@fake.com | 678954321 | 12345 | F | 12345 | Programming Languages | 4 |
| 789012345 | Ivana Tinkle | 2475 Sunrise Blvd, Rancho Cordova, CA 95670 | ivanatinkle@fake.com | 789012345 | 12345 | B | 12345 | Programming Languages | 4 |

```
rows in set (0.00 sec)
```

```
mysql> DELETE FROM Enrolled WHERE SSN=678954321 AND CourseNo=12345;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages' AND C.CourseNo=E.CourseNo AND E.SSN = S.SSN;
```

| SSN | StudentName | Address | Email | SSN | CourseNo | Grade | CourseNo | CourseName | NoOfStudents |
|-----------|--------------|---|----------------------|-----------|----------|-------|----------|-----------------------|--------------|
| 123456789 | Moe Ron | 16514 Golden Valley Pkwy, Lathrop, CA 95330 | moeron@fake.com | 123456789 | 12345 | C | 12345 | Programming Languages | 3 |
| 214407735 | Seymour Butz | 2727 W March Ln, Stockton, CA 95219 | seymourbutz@fake.com | 214407735 | 12345 | A | 12345 | Programming Languages | 3 |
| 789012345 | Ivana Tinkle | 2475 Sunrise Blvd, Rancho Cordova, CA 95670 | ivanatinkle@fake.com | 789012345 | 12345 | B | 12345 | Programming Languages | 3 |

```
rows in set (0.00 sec)
```

Figure 5: Deleted Hugh from Programming Languages.

```
nguyen_hw4.java
9
10 public class nguyen_hw4 {
11
12     public static void main (String args[]) {
13         String url= "jdbc:mysql://athena.ecs.csus.edu/cs174119";
14         String username= "cs174119";
15         String password = "hdhmqwdk";
16
17         nguyen_hw4 example = new nguyen_hw4();
18         example.displayData(url,username,password);
19     }
20
21     public void displayData(String urlStr,String username,String password) {
22
23         try {
24             Class.forName ("com.mysql.jdbc.Driver").getDeclaredConstructor().newInstance();
25
26             Connection conn = DriverManager.getConnection(urlStr,username,password);
27             System.out.println ("Connected to the MySQL database");
28
29             System.out.println("\nThe SSN and Name of students enrolled in Programming Languages.");
30             Statement stmt1 = conn.createStatement();
31             ResultSet rs1=null;
32             rs1=stmt1.executeQuery("SELECT * FROM Student AS S, Enrolled AS E, Course AS C WHERE C.CourseName='Programming Languages' AND C.CourseNo=E.CourseNo AND E.SSN = S.SSN");
33             while (rs1.next()) {
34                 System.out.println("SSN: " + rs1.getString(1) + ", Name: " + rs1.getString(2));
35             }
36             stmt1.close();
37         }
38     }
39 }
```

```
<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 9:16:29 PM)
Connected to the MySQL database

The SSN and Name of students enrolled in Programming Languages.
SSN: 123456789, Name: Moe Ron
SSN: 214407735, Name: Seymour Butz
SSN: 789012345, Name: Ivana Tinkle

Disconnected
```

Figure 6: Hugh is no longer present in jdbc.

Task 2: Query “TA_course” view (created in homework2) to display all the tuples in that view.

- First things first is to query the **TA_Course** view on mysql. To do so run the code ‘**SELECT * FROM TA_Course;**’

```
mysql> SELECT * FROM TA_Course;
```

| TA_Name | TA_Email | Course_Name |
|-------------------|--------------------------|---|
| Moe Ron | moeron@fake.com | Programming Languages |
| Moe Ron | moeron@fake.com | Data Warehousing and Data Mining |
| Seymour Butz | seymourbutz@fake.com | Introduction to Probability Theory |
| Seymour Butz | seymourbutz@fake.com | Computer Forensics Principles and Practices |
| Seymour Butz | seymourbutz@fake.com | Database Management Systems |
| Amanda Hugginkiss | amandahugginkissfake.com | Introduction to Mathematical Statistics |

```
6 rows in set (0.00 sec)
```

```
mysql>
```

Figure 7: Results of TA_Course.

- Then running the following code created the same results. I am uncertain on how to get them in table form. I will look into that.

```
nguyen_hw4.java
37 stmt1.close();
38 /*
39 System.out.println("\nQuerying the tuples of TA_Course.");
40 Statement stmt2 = conn.createStatement();
41 ResultSet rs2=null;
42 rs2=stmt2.executeQuery("SELECT * FROM TA_Course");
43 while (rs2.next()) {
44     String TA_Name = rs2.getString("TA_Name");
45     String TA_Email = rs2.getString("TA_Email");
46     String Course_Name =rs2.getString("Course_Name");
47     System.out.println("TA_Name: " + TA_Name + ", TA_Email: " + TA_Email + ", Course_Name: "
48 }
49 stmt2.close();
50 /*
```

Console

<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 8:59:02 PM)

Connected to the MySQL database

Querying the tuples of TA_Course.

TA_Name: Moe Ron, TA_Email: moeron@fake.com, Course_Name: Programming Languages

TA_Name: Moe Ron, TA_Email: moeron@fake.com, Course_Name: Data Warehousing and Data Mining

TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Introduction to Probability Theory

TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Computer Forensics Principles and Practices

TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Database Management Systems

TA_Name: Amanda Hugginkiss, TA_Email: amandahugginkissfake.com, Course_Name: Introduction to Mathematical Statistics

Disconnected

Figure 8: Results from the JDBC after running the following code.

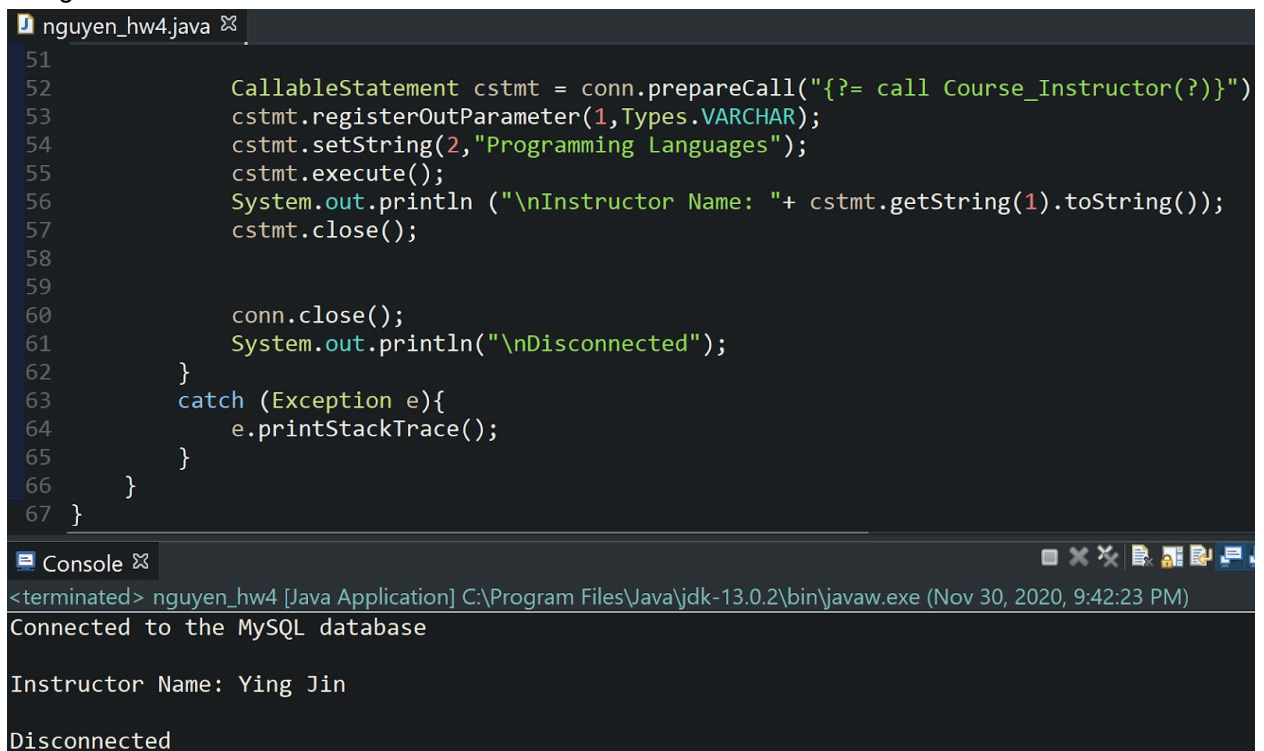
Task 3: Call the “Course_Instructor” function (created in homework2). Given course name “Programming Languages”, display the instructor name. You need to insert data into the database before you call this function, such that the function can return at least one result.

- To begin we can run this on MYSQL to get an example output. To do so run the following code:
`'SELECT Course_Instructor('Programming Languages') AS Instructor_Name; '`

```
mysql> SELECT Course_Instructor('Programming Languages') AS Instructor_Name;
+-----+
| Instructor_Name |
+-----+
| Ying Jin        |
+-----+
1 row in set (0.00 sec)
```

Figure 9: The results of calling the function Course_Instructor.

- Now we run JDBC and compare the results. Note: that I have specifically split my code to output in segments.



```
nguyen_hw4.java
51
52     CallableStatement cstmt = conn.prepareCall("{?= call Course_Instructor(?)}");
53     cstmt.registerOutParameter(1,Types.VARCHAR);
54     cstmt.setString(2,"Programming Languages");
55     cstmt.execute();
56     System.out.println ("\nInstructor Name: "+ cstmt.getString(1).toString());
57     cstmt.close();
58
59
60     conn.close();
61     System.out.println("\nDisconnected");
62 }
63 catch (Exception e){
64     e.printStackTrace();
65 }
66 }
67 }
```

```
<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 9:42:23 PM)
Connected to the MySQL database

Instructor Name: Ying Jin

Disconnected
```

Figure 10: The results of called the function Course_Instructor on JDBC

Now an overall run of my JDBC and its output:

```
nguyen_hw4.java
20 }
21
22 public void displayData(String urlStr,String username,String password) {
23     try {
24         Class.forName ("com.mysql.jdbc.Driver").getDeclaredConstructor().newInstance();
25
26         // Gaining Connection to the MySQL Database //
27         Connection conn = DriverManager.getConnection(urlStr,username,password);
28         System.out.println ("Connected to the MySQL database");
29
30         // SSN and Name of Students in Programming Languages Course //
31         System.out.println("\nThe SSN and Name of students enrolled in Programming Languages Course");
32         Statement stmt1 = conn.createStatement();
33         ResultSet rs1=null;
34         rs1=stmt1.executeQuery("SELECT * FROM Student AS S, Enrolled AS E,
35         while (rs1.next()) {
36             System.out.println("SSN: " + rs1.getString(1) + ", Name: " + rs1.getString(2));
37         }
38         stmt1.close();
39
40         // Querying the Tuples for TA_Course //
41         System.out.println("\nQuerying the tuples of TA_Course.");
42         Statement stmt2 = conn.createStatement();
43         ResultSet rs2=null;
44         rs2=stmt2.executeQuery("SELECT * FROM TA_Course");
45         while (rs2.next()) {
46             String TA_Name = rs2.getString("TA_Name");
47             String TA_Email = rs2.getString("TA_Email");
48             String Course_Name = rs2.getString("Course_Name");
49             System.out.println("TA_Name: " + TA_Name + ", TA_Email: " + TA_Email + ", Course_Name: " + Course_Name);
50         }
51         stmt2.close();
52
53         // Calling the Course_Instructor //
54         System.out.println("\nCalling the function Course Instructor.");
55         CallableStatement cstmt = conn.prepareCall("{?= call Course_Instructor(?,?)");
56         cstmt.registerOutParameter(1,Types.VARCHAR);
57         cstmt.setString(2,"Programming Languages");
58         cstmt.execute();
59         System.out.println ("Instructor Name: "+ cstmt.getString(1).toString());
60         cstmt.close();
61
62         // Closing the MySQL Connection //
63         conn.close();
64         System.out.println("\nDisconnected");
65     }
66     catch (Exception e){
67         e.printStackTrace();
68     }
69 }
```

```
Console
<terminated> nguyen_hw4 [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Nov 30, 2020, 9:57:05 PM)
Connected to the MySQL database

The SSN and Name of students enrolled in Programming Languages.
SSN: 123456789, Name: Moe Ron
SSN: 214407735, Name: Seymour Butz
SSN: 789012345, Name: Ivana Tinkle

Querying the tuples of TA_Course.
TA_Name: Moe Ron, TA_Email: moeron@fake.com, Course_Name: Programming Languages
TA_Name: Moe Ron, TA_Email: moeron@fake.com, Course_Name: Data Warehousing and Data Mining
TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Introduction to Probability Theory
TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Computer Forensics Principles and Practices
TA_Name: Seymour Butz, TA_Email: seymourbutz@fake.com, Course_Name: Database Management Systems
TA_Name: Amanda Hugginkiss, TA_Email: amandahugginkissfake.com, Course_Name: Introduction to Mathematics

Calling the function Course Instructor.
Instructor Name: Ying Jin

Disconnected
```