Indian Institute of Information Technology Vadodara

CS262: Database Management System

Lab 7

Roll No. 201951105 **Name: Nishant Andoriya**

For this assignment, you can create a interface using console only.

If the **com.mysql.jdbc.Driver** showing **class not found error** then you can follow the following link **https://www.javatpoint.com/example-to-connect-to-the-mysql-database**

You can start the assignment with the following code

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Date;

public class MySQLAccess {
 private Connection connect = null;
 private Statement statement = null;
 private PreparedStatement preparedStatement = null;
 private ResultSet resultSet = null;

 public void readDataBase() throws Exception {
 try {
 // This will load the MySQL driver, each DB has its own
driver  Class.forName("com.mysql.jdbc.Driver");
 // Setup the connection with the DB
 connect = DriverManager
.getConnection("jdbc:mysql://localhost:3306/dbName", "sqluser”,”sqluserpassword");
```

Question 1.) You have already created the following database,

Consider the following relational schema for the Office of the Controller of Examinations Application.

Student (Rollno, Name, Dob, Gender, Doa, Bcode);

Implement a check constraint for

- Gender

Branch (Bcode, Bname, Dno);

Department (Dno, Dname);

Course (Ccode, Cname, Credits, Dno);

Branch_Course (Bcode, Ccode, Semester);

Enrolls (Rollno, Ccode, Sess, Grade);

Implement a check constraint for grade Value Set (‘S’, ‘A’, ‘B’, ‘C’, ‘D’, ‘E’, ‘U’); Students are admitted to Branches and they are offered by Departments. A branch is offered by only one department.

Each branch has a set of Courses (Subjects). Each student must enroll during a semester. Courses are offered by Departments. A course is offered only by one department. If a student is unsuccessful in a course he/she must enroll for the course during next session. A student has successfully completed a course if the grade obtained by is from the list (A, B, C, D, and E).

A student is unsuccessful if he/she have grade ‘U’ in a course.

Primary Keys are underlined.

a.) Create a program to interact with the database.

b.) Create a program to create a new branch, department and course. At the time of creating a course also map the course with the branch.

c.) Create a program which will take the student information from the user and insert it into the database.

d.) Create a program which entrolls a student to anyone of the already created courses. If user enters the course which is not already present in the course relation then give a message to the user that the course does not exist.

Create the following listing using program

e.) List details of Departments that offer more than 3 branches.

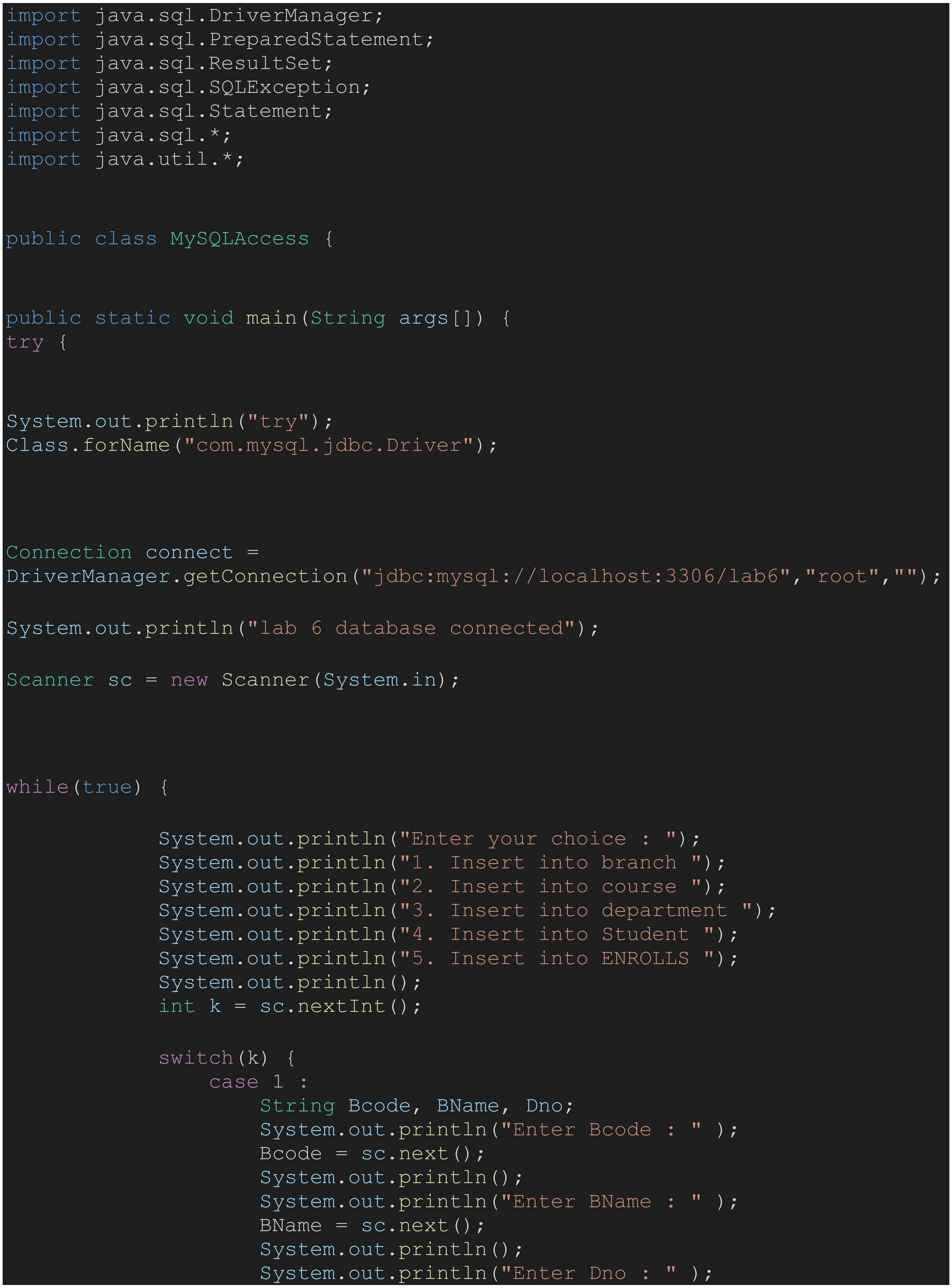
f.) List the details of Departments that offer more than 6 courses.

g.) List the details of courses that are common for more than 3 branches.

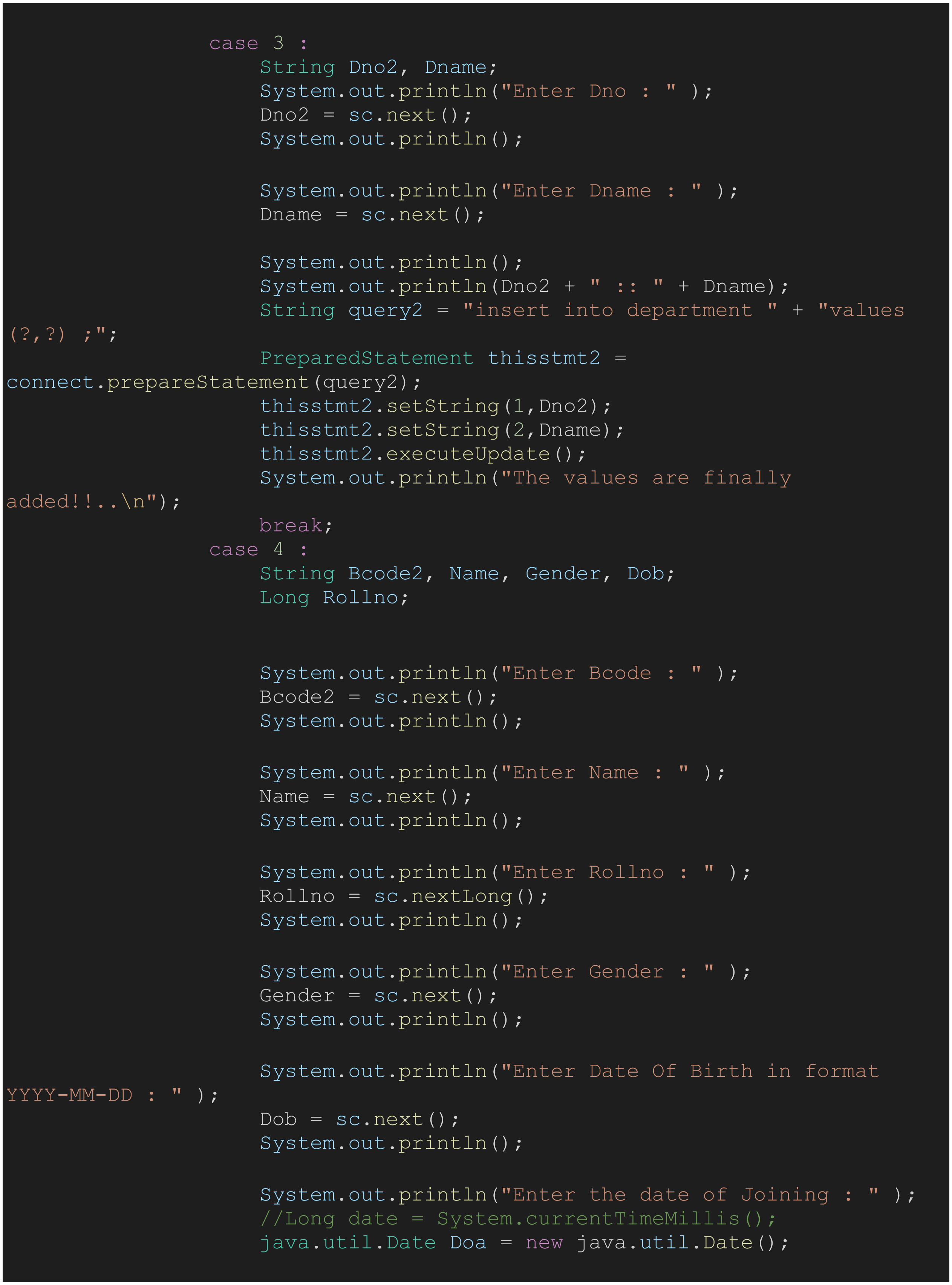
h.) List students who got ‘S’ in more than 2 courses during single enrollment.

Program for Question 1, Part - a,b,c,d :

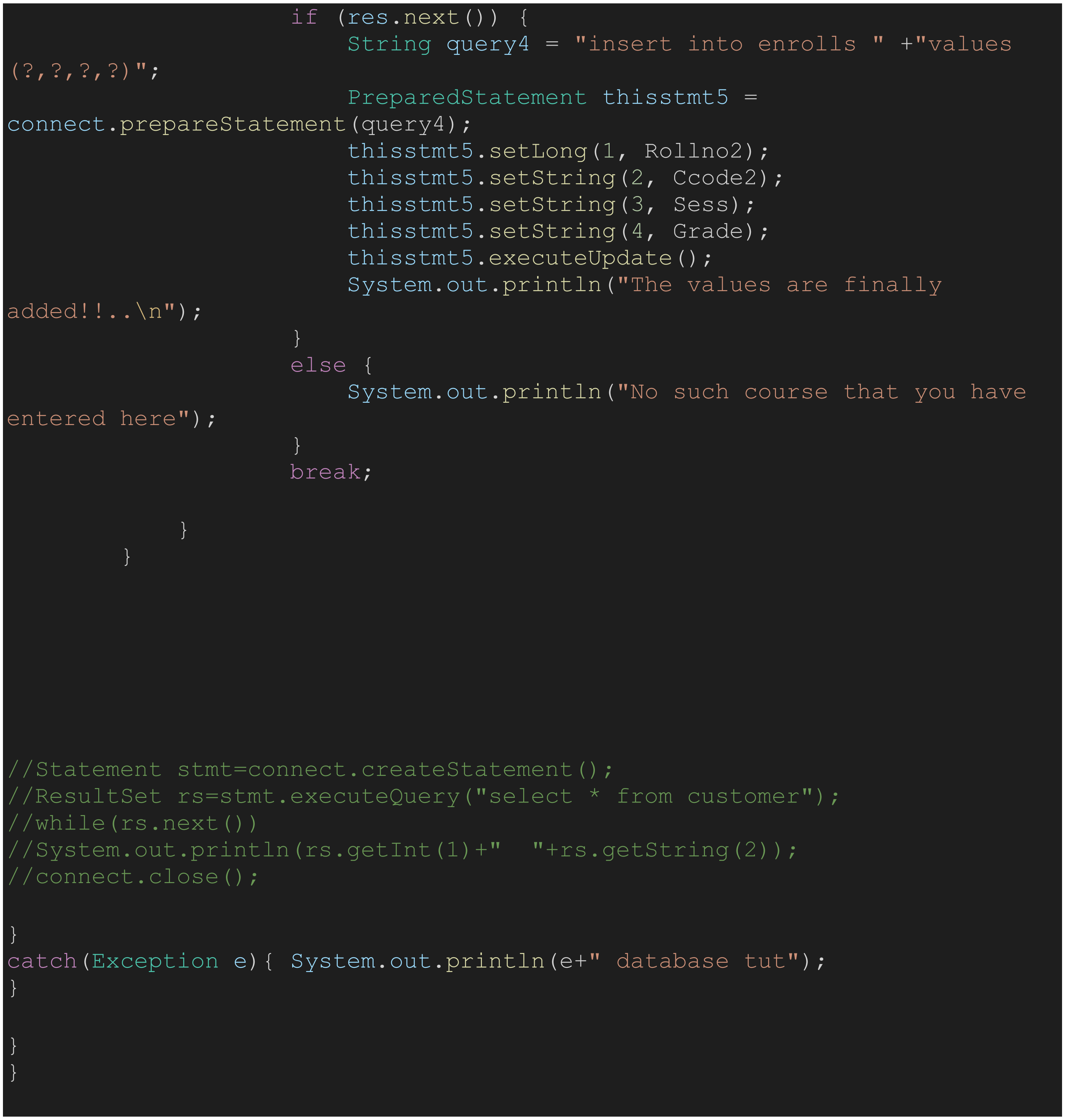
```
import java.sql.Connection;
```



```
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.sql.*;
import java.util.*;


public class MySQLAccess {


public static void main(String args[]) {
try {


System.out.println("try");
Class.forName("com.mysql.jdbc.Driver");



Connection connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/lab6","root","");

System.out.println("lab 6 database connected");

Scanner sc = new Scanner(System.in);



while(true) {

            System.out.println("Enter your choice : ");
            System.out.println("1. Insert into branch ");
            System.out.println("2. Insert into course ");
            System.out.println("3. Insert into department ");
            System.out.println("4. Insert into Student ");
            System.out.println("5. Insert into ENROLLS ");
            System.out.println();
            int k = sc.nextInt();

            switch(k) {
                case 1 :
                    String Bcode, BName, Dno;
                    System.out.println("Enter Bcode : " );
                    Bcode = sc.next();
                    System.out.println();
                    System.out.println("Enter BName : " );
                    BName = sc.next();
                    System.out.println();
                    System.out.println("Enter Dno : " );
                    Dno = sc.next();
                    System.out.println();
                    System.out.println(Bcode + " :: " + BName + " :: " +
Dno);
                    String query = "insert into branch " + "values (?,?,?)
;";
                    PreparedStatement thisstmt =
connect.prepareStatement(query);
                    thisstmt.setString(1,Bcode);
                    thisstmt.setString(2,BName);
                    thisstmt.setString(3,Dno);
                    thisstmt.executeUpdate();
                    System.out.println("The values are finally
added!!..\n");
                    Break;


                case 2 :

                    int Credits;
                    String Ccode, Cname, Dno1;
                    System.out.println("Enter Ccode : " );
                    Ccode = sc.next();
                    System.out.println();

                    System.out.println("Enter Cname : " );
                    Cname = sc.next();
                    System.out.println();

                    System.out.println("Enter Dno : " );
                    Dno1 = sc.next();
                    System.out.println();

                    System.out.println("Enter Credits : " );
                    Credits = sc.nextInt();
                    System.out.println();

                    System.out.println(Ccode + " :: " + Cname + " :: " +
Dno1 + " :: " + Credits);
                    String query1 = "insert into course " + "values
(?,?,?,?) ;";
                    PreparedStatement thisstmt1 =
connect.prepareStatement(query1);
                    thisstmt1.setString(1,Ccode);
                    thisstmt1.setString(2,Cname);
                    thisstmt1.setInt(3, Credits);
                    thisstmt1.setString(4,Dno1);
                    thisstmt1.executeUpdate();
                    System.out.println("The values are finally
added!!..\n");
                    Break;
```



```
                case 3 :
                    String Dno2, Dname;
                    System.out.println("Enter Dno : " );
                    Dno2 = sc.next();
                    System.out.println();

                    System.out.println("Enter Dname : " );
                    Dname = sc.next();

                    System.out.println();
                    System.out.println(Dno2 + " :: " + Dname);
                    String query2 = "insert into department " + "values
(?,?) ;";
                    PreparedStatement thisstmt2 =
connect.prepareStatement(query2);
                    thisstmt2.setString(1,Dno2);
                    thisstmt2.setString(2,Dname);
                    thisstmt2.executeUpdate();
                    System.out.println("The values are finally
added!!..\n");
                    break;
                case 4 :
                    String Bcode2, Name, Gender, Dob;
                    Long Rollno;


                    System.out.println("Enter Bcode : " );
                    Bcode2 = sc.next();
                    System.out.println();

                    System.out.println("Enter Name : " );
                    Name = sc.next();
                    System.out.println();

                    System.out.println("Enter Rollno : " );
                    Rollno = sc.nextLong();
                    System.out.println();

                    System.out.println("Enter Gender : " );
                    Gender = sc.next();
                    System.out.println();

                    System.out.println("Enter Date Of Birth in format
YYYY-MM-DD : " );
                    Dob = sc.next();
                    System.out.println();

                    System.out.println("Enter the date of Joining : " );
                    //Long date = System.currentTimeMillis();
                    java.util.Date Doa = new java.util.Date();
                    System.out.println(Bcode2 + " :: " + Name + " :: " +
Rollno + " :: " + Gender + " :: " + Dob + " :: " + Doa);
                    String query3 = "insert into student " + "values
(?,?,?,?,?,?) ;";

                    PreparedStatement thisstmt3 =
connect.prepareStatement(query3);

                    thisstmt3.setLong(1,Rollno);
                    thisstmt3.setString(2, Name);

////////////////////////
                    thisstmt3.setString(3, Dob);
                    thisstmt3.setString(4, Gender);
///////////////////
                      java.sql.Date DOA2 = new
java.sql.Date(Doa.getTime());
                    thisstmt3.setDate(5, DOA2);
////////////////////////
                    thisstmt3.setString(6, Bcode2);
                    thisstmt3.executeUpdate();
                    System.out.println("The values are finally
added!!..\n");
                    Break;


                case 5 :
                    Long Rollno2;
                    String Ccode2, Sess, Grade;
                    System.out.println("Enter CCode : " );
                    Ccode2 = sc.next();
                    System.out.println();

                    System.out.println("Enter Session : " );
                    Sess = sc.next();
                    System.out.println();

                    System.out.println("Enter Grade : " );
                    Grade = sc.next();
                    System.out.println();

                    System.out.println("Enter Roll Number : " );
                    Rollno2 = sc.nextLong();
                    System.out.println();

                    System.out.println(Rollno2 + " :: " + Ccode2 + " :: "
+ Sess + " :: " + Grade);
                    PreparedStatement thisstmt4 =
connect.prepareStatement("Select * from course where Ccode = ?");
                    thisstmt4.setString(1, Ccode2);
                    ResultSet res = thisstmt4.executeQuery();
```

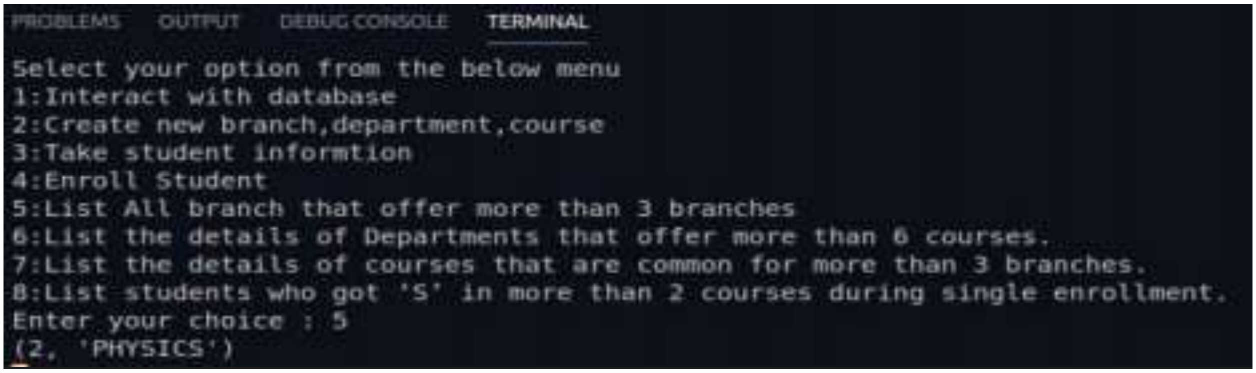


```
                    if (res.next()) {
                        String query4 = "insert into enrolls " +"values
(?,?,?,?)";
                        PreparedStatement thisstmt5 =
connect.prepareStatement(query4);
                        thisstmt5.setLong(1, Rollno2);
                        thisstmt5.setString(2, Ccode2);
                        thisstmt5.setString(3, Sess);
                        thisstmt5.setString(4, Grade);
                        thisstmt5.executeUpdate();
                        System.out.println("The values are finally
added!!..\n");
                    }
                    else {
                        System.out.println("No such course that you have
entered here");
                    }
                    break;

            }
        }







//Statement stmt=connect.createStatement();
//ResultSet rs=stmt.executeQuery("select * from customer");
//while(rs.next())
//System.out.println(rs.getInt(1)+"  "+rs.getString(2));
//connect.close();

}
catch(Exception e){ System.out.println(e+" database tut");
}

}
}
```

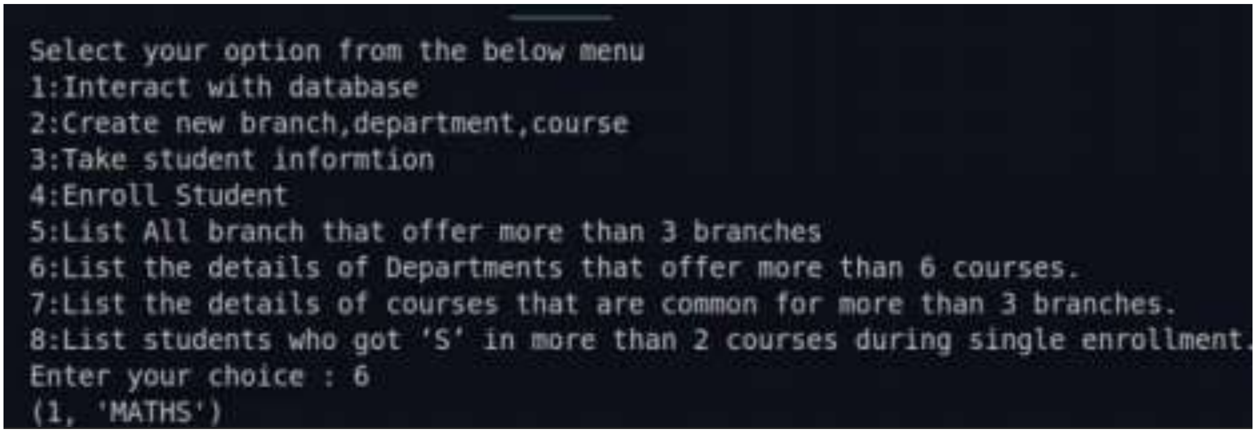
Program for Question 1, Part - e,f,g,h :

```
import java.sql.*;
import java.util.*;


public class Queries {
    public static void main(String[] args) throws Exception {


   try {

       System.out.println("try");
   Class.forName("com.mysql.jdbc.Driver");

    Connection connect =
DriverManager.getConnection("jdbc:mysql://localhost:3306/lab6","root","");

   System.out.println("lab 6 database connected");

   Scanner sc = new Scanner(System.in);


            while(true)
            {
                System.out.println("1. Query for part e");
                System.out.println("2. Query for part f");
                System.out.println("3. Query for part g");
                System.out.println("4. Query for part h");
                System.out.println();
                int k = sc.nextInt();



                switch (k) {
                    case 1 :
                        Statement stmt1 = connect.createStatement();
                        ResultSet res = stmt1.executeQuery("select * from
department where department.Dno in(Select branch.Dno from branch group by
branch.Dno having Count(branch.Dno) > 3);");
                        System.out.println("The Output of the query : ");
                        while(res.next())
                        System.out.println(res.getString(1) + " | " +
res.getString(2) + " | ");
                        System.out.println();
                        break;
                    case 2 :
                        Statement stmt2 = connect.createStatement();
                        ResultSet res1 = stmt2.executeQuery("select * from
department where department.Dno in(Select course.Dno from course group by
course.Dno having Count(course.Dno) > 6);");
                        System.out.println("The Output of the query : ");
                        while(res1.next())
                        System.out.println(res1.getString(1) + " | " +
res1.getString(2) + " | " );
                        System.out.println();
                        break;



                    case 3 :
                        Statement stmt3 = connect.createStatement();
                        ResultSet res2 = stmt3.executeQuery("select * from
course where course.Ccode in(Select branch_course.Ccode from branch_course
group by branch_course.Ccode having Count(branch_course.Ccode) > 3);");
                        System.out.println("The Output of the query  : ");
                        while(res2.next())
                        System.out.println(res2.getString(1) + " | " +
res2.getString(2) + " | " + res2.getString(3) + " | " + res2.getString(4)
+ " | ");

                        System.out.println();
                        break;



                    case 4 :
                        Statement stmt4 = connect.createStatement();
                        ResultSet res3 = stmt4.executeQuery("select * from
student where student.Rollno in(Select enrolls.Rollno from enrolls where
enrolls.Grade = 'S' group by enrolls.Rollno having Count(enrolls.Grade) >
2);");
                        System.out.println("The Output of the query : ");
                        while(res3.next())
                        System.out.println(res3.getString(1) + " | " +
res3.getString(2) + " | " + res3.getString(3) + " | " + res3.getString(4)
+ " | " + res3.getString(5) + " | " + res3.getString(6) + " | ");
                        System.out.println();
                        break;

                    default :
                        System.out.println("No such input try again");
                        break;

                }

            }

        } catch (Exception e) {
            System.out.println(e);
            }

      }
```

OUT[UT:-



```
Select your option from the below menu
1:Interact with database
2:Create new branch,department,course
3:Take student informtion
4:Enroll Student
5:List All branch that offer more than 3 branches
6:List the details of Departments that offer more than 6 courses.
7:List the details of courses that are common for more than 3 branches.
8:List students who got ‘S’ in more than 2 courses during single enrollment.
Enter your choice : 5
(2, 'PHYSICS')
```



```
Select your option from the below menu
1:Interact with database
2:Create new branch,department,course
3:Take student informtion
4:Enroll Student
5:List All branch that offer more than 3 branches
6:List the details of Departments that offer more than 6 courses.
7:List the details of courses that are common for more than 3 branches.
8:List students who got ‘S’ in more than 2 courses during single enrollment.
Enter your choice : 6
(1, 'MATHS')
```