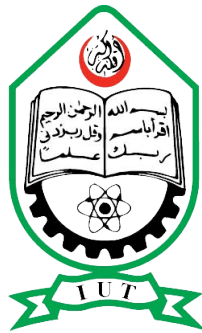

DBMS LAB 09 TASKS

Prepared by:
Mohammad Anas Jawad
Lecturer, IUT CSE



Department of Computer Science and Engineering
Islamic University of Technology
July 22, 2020

Contents

1	Scenario	3
2	Tasks and Instructions	4
3	SampleSolution	6

1 SCENARIO

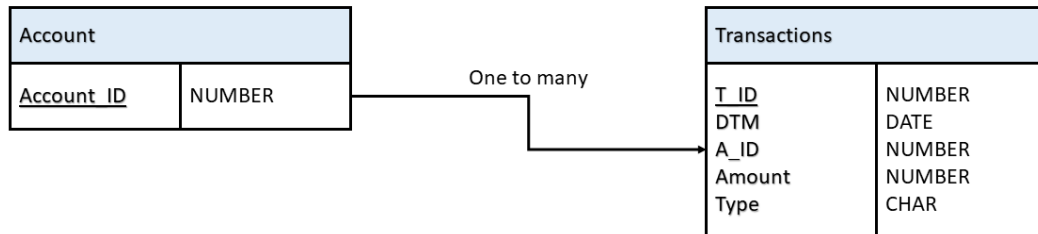


Figure 1: Given Schema

For simplicity, ACCOUNT table contains only the account ID. TRANSACTIONS table contain the information related to each transaction that occurs in the bank. The table stores transaction ID, date of the transaction, account involved in the transaction, the amount transacted. The TYPE column stores 0 if the money is transferred to the account, that is, the money is added to the account. It stores 1 if the money is transferred from the account, that is, money is subtracted from the account.

There are 3 types of accounts in the bank.

- If the user has a total final balance of more than 1000000 and the total amount of transactions (both debit and credit) the person has made totals more than 5000000, that person is regarded as Commercially Important Person (CIP).
- If the user has a total final balance that is more than 500000 but less than 900000 and the total amount of transactions the person has made totals more than 3500000 but less than 4500000, that person is regarded as Very Important Person (VIP).
- If the user has a total final balance that is less than 40000 and the total amount of transactions the person has made totals less than 3000000, that person is regarded as Ordinary.

2 TASKS AND INSTRUCTIONS

Carry out the following instructions:

1. Download and execute `TableGenerator.cpp`. It will create a file named `table.sql` which contains the required SQL queries to create and populate the database. The values will be random except for the account numbers. (These have been simplified for later purposes).
2. Make sure Oracle Database is installed in your PC. Login to your database (either using Command Prompt or SQL Command Line). Enter the command: `@directory\table.sql`. Here, (directory) should be replaced by the file path of `table.sql` file. If the file is stored in D: drive, the command will look like:

`@D:\table.sql`
3. Create a new Java project and add `ojdbc6.jar` file as an external JAR file. (The JAR file has been provided in Google Classroom).
4. Go through section 3 to take a look at a sample solution. You can use it as a reference for your solution. [Additional hint: loops and if-else blocks in java have the same syntax as in C++.]
5. Now, write a JAVA code to:
 - Establish connection to the database.
 - Fetch required data from the `TRANSACTIONS` table.
 - Count the total number of accounts, number of CIP, VIP and Ordinary persons.
 - Also count the number of persons that were uncategorized.

The output of your program should look like this:

```
1
2      Total Number of Accounts: 147
3      CIP count: 25
4      VIP count: 2
5      Ordinary count: 1
6      Uncategorized: 119
7
```

Note that the numbers will not be the same for you. As your table entries will be generated randomly, everyone should have different answers. **If I find any output with similar values, you can already guess what will happen.**

6. Add a comment above each variable declaration, explaining what it stores.
7. Add comments above each function call (if any), explaining what it does and what parameters are passed to the function.
8. Create a zip file containing your JAVA code and `table.sql` file. Rename the zip file using the format `Lab09_STUDENT_ID.zip` and submit it via Google Classroom. Example: For student with ID 154443, the file name will be `Lab09_154443.zip`

3 SAMPLESOLUTION

SampleSolution.java

```
1 import java.sql.*;
2 public class SampleSolution {
3     public static void main(String[] args) {
4         String username = "senpai";
5         String password = "senpai";
6         String url = "jdbc:oracle:thin:@localhost:1521/XE";
7         String sqlQuery = "SELECT A_ID, AMOUNT, TYPE FROM TRANSACTIONS"
8         ;
9
10        int[] balance = new int[1000];
11        int[] total = new int[1000];
12        int account;
13        int amount;
14        String type; /* Hint: These variables will help you with your
15        code. You might also need some other variables though! */
16        try {
17            // 1) Register the driver class
18            Class.forName("oracle.jdbc.driver.OracleDriver");
19            // 2) Create the connection object
20            Connection con = DriverManager.getConnection(url, username,
21            password);
22            // 3) Create the Statement object
23            Statement statement = con.createStatement();
24            System.out.println("Connection to database successful");
25            // 4) Execute the query
26            ResultSet result = statement.executeQuery(sqlQuery);
27            while (result.next()) {
28                account = result.getInt("a_id"); //Could also be
29                written as result.getInt(1);
30                amount = result.getInt("amount"); //Could also be
31                written as result.getInt(2);
32                type = result.getString("type"); //Could also be
```

```
written as result.getInt(3);
28         System.out.println(account + " " + amount + " " + type)
;
29         /* Additional Code */
30     }
31     /* Additional Code */
32     // 5) Close the connection object
33     con.close();
34     statement.close();
35     result.close();
36 } catch (SQLException e) {
37     System.out.println("Error while connecting to database.
Exception code: " + e);
38 } catch (ClassNotFoundException e) {
39     System.out.println("Failed to register driver. Exception
code: " + e);
40 }
41 System.out.println("Thank You!");
42 }
43 }
```