

Islamic University of Technology

CSE-4308

Database Management Systems Lab

Lab-8 Report

Name : Mukit Mahdin

ID : 200042170

Department: CSE

Program: SWE

Batch: 2020

Group: B

Task-1:

Count the number of transactions conducted under account 49

Java Code :

```
import java.sql.*;

public class task1{

    static final String JDBC_DRIVER =
"oracle.jdbc.driver.OracleDriver";
    static final String DB_URL=
"jdbc:oracle:thin:@localhost:1521:xe";
    static final String USER="mukit";
    static final String PASS="7445";

    public static void main (String args[]){
        Connection conn=null;
        Statement stmt=null;
        try{
            Class.forName(JDBC_DRIVER);
            System.out.println("Connecting to database");
            conn=DriverManager.getConnection(DB_URL, USER, PASS);
            System.out.println("Creating statement");
            stmt=conn.createStatement();
            String sql;
            sql="select count(a_id) as AccountTransactions from
transactions where a_id='49'";
            System.out.println("Executing the query: " + sql);
            ResultSet rs=stmt.executeQuery(sql);

            while(rs.next()){
                int
AccountTransaction=rs.getInt("AccountTransactions");
                System.out.println("The number of transactions done
by A_ID 49 is : "+AccountTransaction);
            }
            rs.close();
            stmt.close();
            conn.close();
            System.out.println("Thank you for banking with us!");
        }
        catch(SQLException se){
```

```

        se.printStackTrace();
    }
    catch(Exception e) {
        e.printStackTrace();
    }
}
}

```

Task-2:

Count the number of Credit

Java Code :

```

import java.sql.*;
public class task2{
    static final String JDBC_DRIVER =
"oracle.jdbc.driver.OracleDriver";
    static final String DB_URL=
"jdbc:oracle:thin:@localhost:1521:xe";
    static final String USER="mukit";
    static final String PASS="7445";

    public static void main (String args[]){
        Connection conn=null;
        Statement stmt=null;
        try{
            Class.forName(JDBC_DRIVER);
            System.out.println("Connecting to database");
            conn=DriverManager.getConnection(DB_URL, USER, PASS);
            System.out.println("Creating statement");
            stmt=conn.createStatement();
            String sql;
            sql="select count(t_id) as NumberOfCredits from
transactions where type='0'";
            System.out.println("Executing the query: " + sql);
            ResultSet rs=stmt.executeQuery(sql);

            while(rs.next()){

```

```

        int credits=rs.getInt("NumberOfCredits");
        System.out.println("Total number of credit is :
"+credits);
    }

    rs.close();
    stmt.close();
    conn.close();
    System.out.println("Thank you for banking with us!");
}
catch(SQLException se)
{
    se.printStackTrace();
}
catch(Exception e)
{
    e.printStackTrace();
}
}
}

```

Task-3:

List the transactions that occurred in the last 6 months of 2021

Java Code :

```

import java.sql.*;
public class task3{
    static final String JDBC_DRIVER =
"oracle.jdbc.driver.OracleDriver";
    static final String DB_URL=
"jdbc:oracle:thin:@localhost:1521:xe";
    static final String USER="mukit";
    static final String PASS="7445";

    public static void main (String args[]){
        Connection conn=null;
        Statement stmt=null;
    }
}

```

```

try{
    Class.forName(JDBC_DRIVER);
    System.out.println("Connecting to database");
    conn=DriverManager.getConnection(DB_URL, USER, PASS);
    System.out.println("Creating statement");
    stmt=conn.createStatement();
    String sql;
    sql="select t_id,extract(year from dtm) as
year,extract(month from dtm) as month from transactions " +
        "group by t_id,extract(year from
dtm),extract(month from dtm)" +
        "having extract(year from dtm)='2021' and
extract(month from dtm)>6";
    System.out.println("Executing the query: " + sql);
    ResultSet rs=stmt.executeQuery(sql);
    while(rs.next()){
        int transid=rs.getInt("t_id");
        System.out.println(transid);
    }
    rs.close();
    stmt.close();
    conn.close();
    System.out.println("Thank you for banking with us!");
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
}
}

```

Task-4:

Count the number of CIP, VIP, and OPs. Also show the number of people that do not fall into any of the categories.

Java Code :

```
import java.sql.*;
public class task4 {
static final String JDBC_DRIVER = "oracle.jdbc.driver.OracleDriver";
static final String DB_URL= "jdbc:oracle:thin:@localhost:1521:xe";
static final String USER="mukit";
static final String PASS="7445";
public static void main (String args[]){
    Connection conn=null;
    Statement stmt=null;

    try{
        Class.forName(JDBC_DRIVER);
        System.out.println("Connecting to database");
        conn=DriverManager.getConnection(DB_URL, USER, PASS);
        System.out.println("Creating statement");
        stmt=conn.createStatement();

        String main_sql="FROM (SELECT A_ID, (RECEIVED-SPENT) AS
                                BALANCE,(RECEIVED+SPENT) AS
TOTAL_TRANSACTIONS FROM((SELECT SUM(AMOUNT) AS RECEIVED,A_ID FROM
ACCOUNT NATURAL JOIN
TRANSACTIONS WHERE TYPE=0 GROUP BY A_ID) T1 NATURAL JOIN (SELECT
SUM(AMOUNT) AS SPENT ,A_ID FROM ACCOUNT NATURAL JOIN TRANSACTIONS
WHERE TYPE=1 GROUP BY A_ID) T2)))";

        String CIPs=main_sql+" WHERE BALANCE ≥ 1000000 AND
TOTAL_TRANSACTIONS ≥ 5000000";
        String sql1="SELECT COUNT(A_ID) AS CIP_NUM " +CIPs;
        String forlastsql1="SELECT A_ID " +CIPs;
        System.out.println("Executing the query: " + sql1);
        ResultSet rs=stmt.executeQuery(sql1);
        while(rs.next()){
            int cip_num=rs.getInt("CIP_NUM");
            System.out.println("Number of CIP : "+ cip_num);
        }

        String VIPs=main_sql+" WHERE BALANCE ≥ 500000 AND
BALANCE<1000000 AND
                                TOTAL_TRANSACTIONS ≥ 2500000 AND
TOTAL_TRANSACTIONS ≤ 4500000 ";
        String sql2="SELECT COUNT(A_ID) AS VIP_NUM " + VIPs;
```

```

String forlastsql2="SELECT A_ID " +VIPs;
System.out.println("Executing the query: "+sql2);
ResultSet rs1=stmt.executeQuery(sql2);

while(rs1.next()){
    int vip_num=rs1.getInt("VIP_NUM");
    System.out.println("Number of VIP : "+ vip_num);
}
String OIPs=main_sql+" WHERE BALANCE ≥ 0 AND BALANCE
≤ 1000000 AND TOTAL_TRANSACTIONS<1000000";
String sql3="SELECT COUNT(A_ID) AS OIP_NUM " + OIPs;
String forlastsql3 = "SELECT A_ID " +OIPs;
System.out.println("Executing the query: "+sql3);
ResultSet rs2=stmt.executeQuery(sql3);
while(rs2.next()){
    int oip_num=rs2.getInt("OIP_NUM");
    System.out.println("Number of OIP : "+ oip_num);
}
String sql4="SELECT A_ID FROM ACCOUNT WHERE A_ID NOT IN (
("+forlastsql1+") UNION ( "+forlastsql2+" ) UNION
("+forlastsql3+"))";
System.out.println("Executing the query: "+sql4);
ResultSet rs3=stmt.executeQuery(sql4);
System.out.println("A_ID:(DON'T FALL INTO ANY CATEGORY) \n");
while(rs3.next()){
    int account_no=rs3.getInt("A_ID");
    System.out.println(account_no);
}
rs.close();
rs1.close();
rs2.close();
rs3.close();
stmt.close();
conn.close();
System.out.println("Thank you for banking with us!");
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
}

```

```
}  
}
```

Description :

It is necessary to overcome the problem of setting up the environment and connecting to the database server using a Java IDE. An application programming interface called JDBC creates a SQL query as a character string at runtime, performs the query, and retrieves the results. On the other hand, configuring it with a Java IDE like Visual Studio, IntelliJ, etc. Code must first be added as an external JAR file before using JDBC. In this situation, when developing code, "java.sql.*" must be included. A connection with a database server must be made at the beginning. There is a built-in DriverManager Class that contains a getConnection method that essentially takes three parameters: JDBC driver (for instance, Oracle), Address of host, and Connection type (In this case localhost).

The getConnection method of the built-in DriverManager Class includes three fundamental parameters: the JDBC driver (for instance, Oracle), the host address (in this case, localhost), the user ID, and the password to connect with. Then it starts generating statements for the database server using the Statement Class and its methods. Despite providing addresses and other information required to connect to the database, we neglected to handle exceptions. If the given values in the parameter were invalid, connection cannot be formed. Therefore, the codebase must include try catch blocks to deal with exceptions.

In Task - 4, The ResultSet class is called to store the results of the SQL statements. Then we can access results row by row from the ResultSet's next function and store them in the variables defined inside the codebase.

Problems Faced :

1. It is difficult to determine if an error or exception occurred because the SQL code was written in the incorrect format or because of some other problem because JDBC validates the queries during runtime rather than at compile time.
2. When writing code inside a string variable, one of the questions that arise is whether to use a semicolon or not. Every time we use the command line, we always terminate the query with a semicolon. In this instance, the executeQuery function of the Statement class appends a semicolon. Therefore, entering a semicolon will result in database errors.
3. Finally, the final task was difficult due to the large number of subqueries involved. As a result, while the code was not running, it became difficult to read and debug in a single string. To solve the problem, we declare multiple string variables and store subparts of the queries in different variables. Then, combine the variables to obtain the desired query and results. Multiple types of results had to be displayed in this task.

