

12. Write a program for Transaction.

Theory:

Transaction represents a single unit of work.

The ACID properties describes the transaction management well. ACID stands for Atomicity, Consistency, isolation and durability.

Atomicity: means either all successful or none.

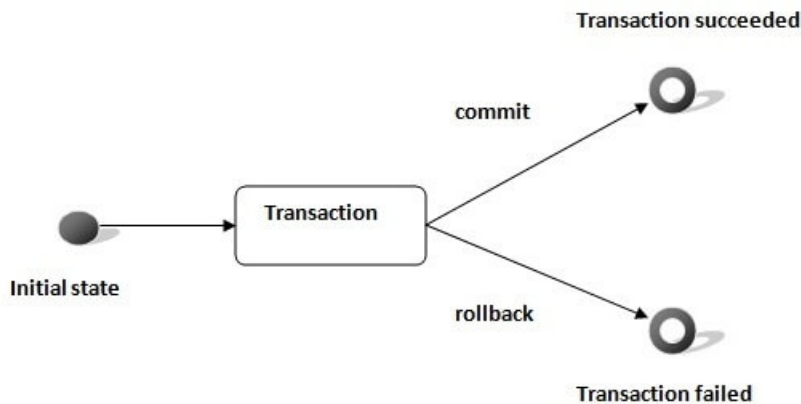
Consistency: ensures bringing the database from one consistent state to another consistent state.

Isolation: ensures that transaction is isolated from other transaction.

Durability: means once a transaction has been committed, it will remain so, even in the event of errors, power loss etc.

Advantage of Transaction Management

fast performance. It makes the performance fast because database is hit at the time of commit.



Source Code:

```
package q10;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class TransactionDemo {
    public static Connection con;
    public static ResultSet rsltset;
    public static Statement statement;

    public static void main(String[] args) throws SQLException {
```

```

try {
    Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
    System.out.println("Driver loaded");
    String server = "//Obs";
    String database = "CollegeDb";
    int port = 1433;

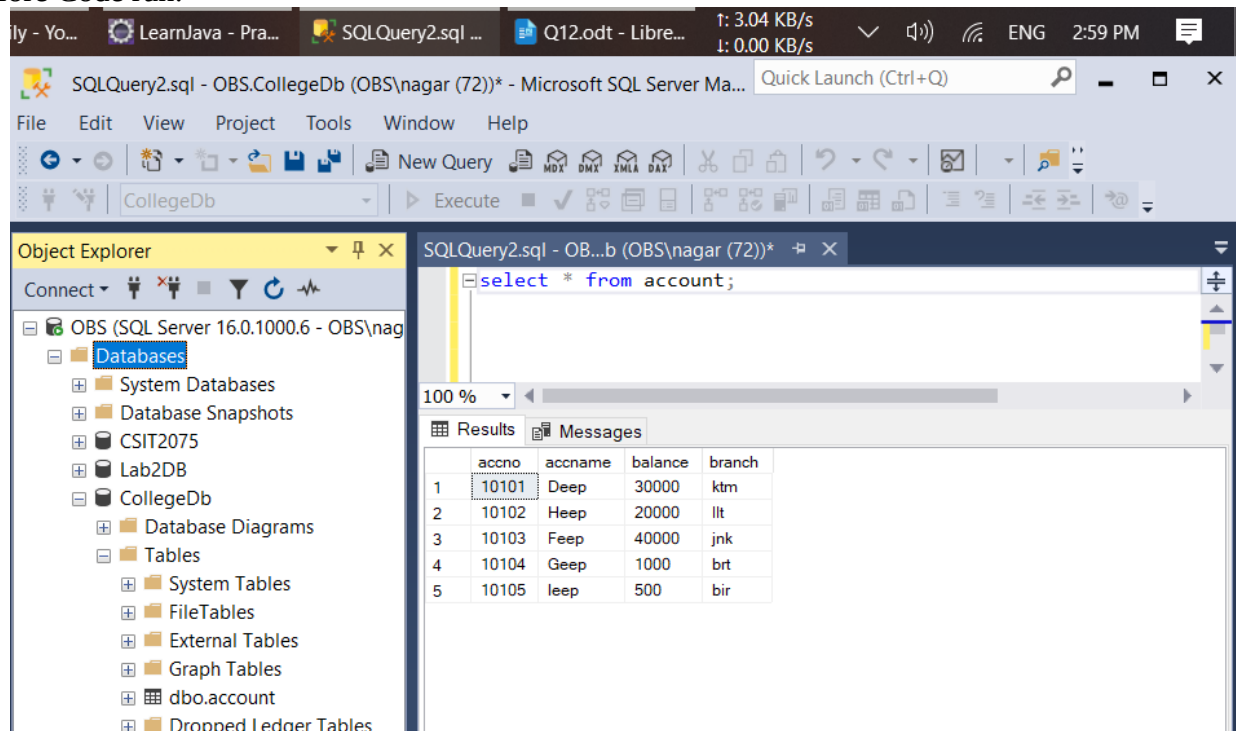
    String jdbcUrl = "jdbc:sqlserver:" + server + ":" + port + ";databaseName=" +
database
        + ";integratedSecurity=true";
    con = DriverManager.getConnection(jdbcUrl);
    System.out.println("Connection obtained");
    statement = con.createStatement();
    System.out.println("Statement is created");
    con.setAutoCommit(false);
    String sql1 = "UPDATE account SET " + "balance=balance-10000 " + " WHERE
accname='Deep'";
    String sql2 = "UPDATE account SET " + "balance=balance+10000 " + " WHERE
accname='Geep'";

    statement.executeUpdate(sql1);
    statement.executeUpdate(sql2);
    con.commit();// Explicit Method for executing transactions.
    ResultSet rsltset;
    rsltset = statement.executeQuery("SELECT * from account;");
    System.out.println("After Transaction Complete");
    while (rsltset.next()) {
        int ano = rsltset.getInt("accno");
        String aname = rsltset.getString("accname");
        float bal = rsltset.getFloat("balance");
        String bran = rsltset.getString("branch");
        System.out.print("Account Number: " + ano);
        System.out.print(", " + " ");
        System.out.print("Account Name: " + aname);
        System.out.print(", " + " ");
        System.out.print("Account Balance: " + bal);
        System.out.print(", " + " ");
        System.out.print("Account Branch: " + bran);
        System.out.println(",");
    }
    rsltset.close();
    statement.close();
} catch (Exception e) {
    con.rollback();
}
}
}

```

Output:

Before Code run:



SQLQuery2.sql - OBS.CollegeDb (OBS\nagar (72))* - Microsoft SQL Server Ma... Quick Launch (Ctrl+Q)

File Edit View Project Tools Window Help

CollegeDb Execute

Object Explorer

Connect

OBS (SQL Server 16.0.1000.6 - OBS\nag

Databases

System Databases

Database Snapshots

CSIT2075

Lab2DB

CollegeDb

Database Diagrams

Tables

System Tables

FileTables

External Tables

Graph Tables

dbo.account

Dropped Ledger Tables

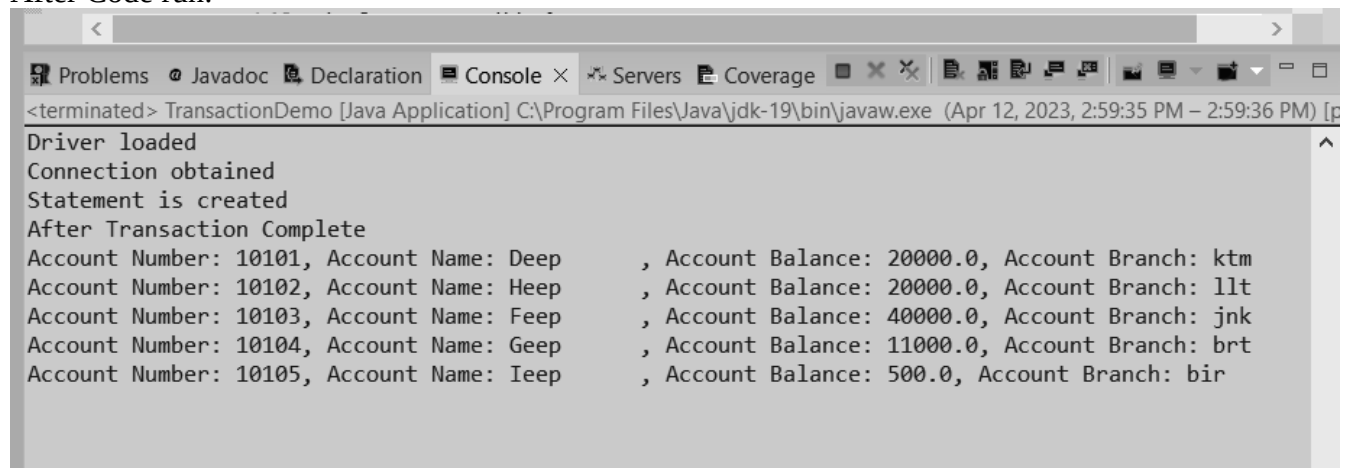
SQLQuery2.sql - OB...b (OBS\nagar (72))*

select * from account;

Results

	accno	accname	balance	branch
1	10101	Deep	30000	ktm
2	10102	Heep	20000	llt
3	10103	Feep	40000	jnk
4	10104	Geep	1000	brt
5	10105	leep	500	bir

After Code run:



Problems Javadoc Declaration Console X Servers Coverage

<terminated> TransactionDemo [Java Application] C:\Program Files\Java\jdk-19\bin\javaw.exe (Apr 12, 2023, 2:59:35 PM - 2:59:36 PM) [p

Driver loaded

Connection obtained

Statement is created

After Transaction Complete

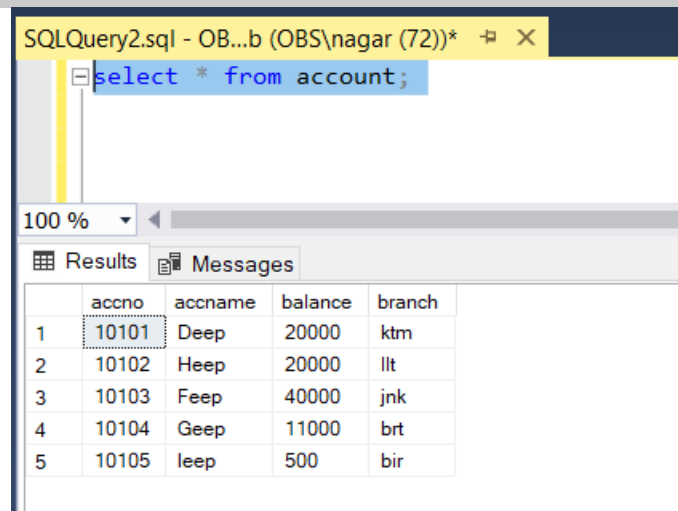
Account Number: 10101, Account Name: Deep , Account Balance: 20000.0, Account Branch: ktm

Account Number: 10102, Account Name: Heep , Account Balance: 20000.0, Account Branch: llt

Account Number: 10103, Account Name: Feep , Account Balance: 40000.0, Account Branch: jnk

Account Number: 10104, Account Name: Geep , Account Balance: 11000.0, Account Branch: brt

Account Number: 10105, Account Name: Ieep , Account Balance: 500.0, Account Branch: bir



SQLQuery2.sql - OB...b (OBS\nagar (72))*

select * from account;

Results

	accno	accname	balance	branch
1	10101	Deep	20000	ktm
2	10102	Heep	20000	llt
3	10103	Feep	40000	jnk
4	10104	Geep	11000	brt
5	10105	leep	500	bir