

Dt : 18/3/2025

(a)getAutoCommit():

=>getAutoCommit()-method is used to know the status of Commit-Operation.

syntax:

boolean b = con.getAutoCommit();

(b)setAutoCommit():

=>setAutoCommit()-method is used to stop the commit-operation

syntax:

con.setAutoCommit(false);

(c)setSavepoint():

=>setSavepoint()-method is used to take one savepoint to perform rollback operation.

syntax:

Savepoint sp = con.setSavepoint();

(d)releaseSavepoint():

=>releaseSavepoint()-method is used to delete the savepoint.

syntax:

con.releaseSavepoint(sp);

(e)commit():

=>commit()-method is used to save the data permanently to Database.

syntax:

con.commit();

(f)rollback():

=>rollback()-method will reset the buufers and moves the control to savepoint,when the transaction is failed.

syntax:

con.rollback();

DB Table : Bank72(accNo,name,bal,accType)

Primary Key:accNo

create table Bank72(accno number(15),name varchar2(15),bal number(10,2),accType varchar2(15),

primary key(accno));

insert into Bank72 values(6123456,'Alex',12000,'Savings');

insert into Bank72 values(313131,'Ram',500,'Savings');

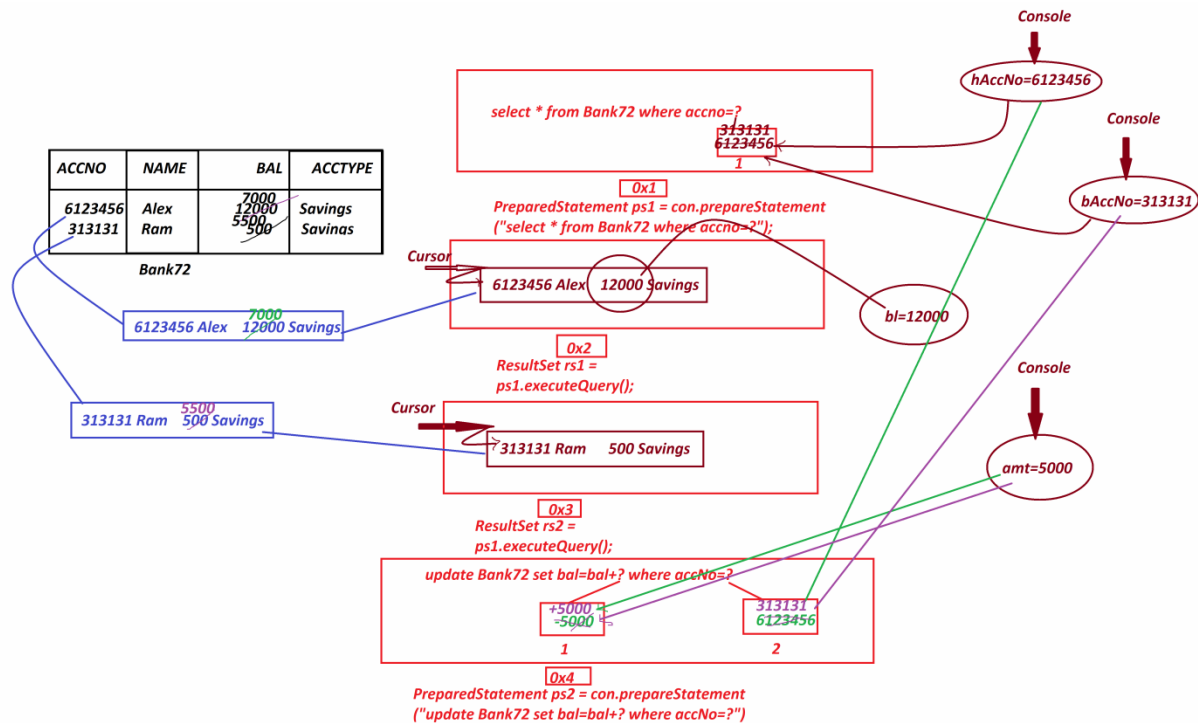
ACCNO NAME	BAL ACCTYPE
6123456 Alex	12000 Savings
313131 Ram	500 Savings

Transaction : Transfer amt:5000/- from accNo:6123456 to accNo:313131

Statement-1 : Subtract amt:5000/- from accNo:6123456

Statement-2 : Add amt:5000/- to accNo:313131

Layout:



Ex:

Program : DBCon14.java

package test;

import java.util.*;

import java.sql.*;

public class DBCon14 {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

try(s){

```

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con = DriverManager.getConnection

    ("jdbc:oracle:thin:@localhost:1521:xe","system","tiger");

PreparedStatement ps1 = con.prepareStatement

    ("select * from Bank72 where accno=?");

PreparedStatement ps2 = con.prepareStatement

    ("update Bank72 set bal=bal+? where accNo=?");

System.out.println("Commit-Status : "+con.getAutoCommit());

con.setAutoCommit(false);//Auto-Commit-Operation-Stopped

System.out.println("Commit-Status : "+con.getAutoCommit());

System.out.println("Enter the Home-AccNo:");

long hAccNo = s.nextLong();

ps1.setLong(1, hAccNo);

ResultSet rs1 = ps1.executeQuery();

if(rs1.next()) {

    float bl = rs1.getFloat(3);

    System.out.println("Enter the benificieryAccNo:");

    long bAccNo = s.nextLong();

    ps1.setLong(1, bAccNo);

    ResultSet rs2 = ps1.executeQuery();

    if(rs2.next()) {

        System.out.println("Enter the Amount to be Transferred:");

        float amt = s.nextFloat();

        if(amt<=bl) {

```

//Statement-1 : Subtract amt:5000/- from accNo:6123456

ps2.setFloat(1, -amt);

ps2.setLong(2, hAccNo);

int p = ps2.executeUpdate();//Updated in Buffer

//Statement-2 : Add amt:5000/- to accNo:313131

ps2.setFloat(1, +amt);

ps2.setLong(2, bAccNo);

int q = ps2.executeUpdate();//Updated in Buffer

if(p==1 && q==1) {

System.out.println("Transaction Successfull...");

con.commit();

}else {

System.out.println("Transaction failed....");

con.rollback();

}

}else {

System.out.println("Insufficient Fund...");

}

}else {

System.out.println("Invalid bAccNo...");

}

}else {

```
        System.out.println("Invalid HomeAccNo...");
    }
    con.close();
} catch (Exception e) {
    e.printStackTrace();
}
}
```

o/p:

Commit-Status : true

Commit-Status : false

Enter the Home-AccNo:

6123456

Enter the beneficiaryAccNo:

313131

Enter the Amount to be Transferred:

5000

Transaction Successfull...

ACCNO NAME	BAL ACCTYPE

6123456 Alex	7000 Savings
313131 Ram	5500 Savings

Venkatesh Maipathii