## PASSBOOK CLASS

```
package com.Bank.App;

import java.util.Iterator;
import java.util.List;

import com.Bank.DAO.TransactionDAO;
import com.Bank.DAO.TransactionDAOImpl;
import com.Bank.DTO.Customer;
import com.Bank.DTO.Transaction;

public class Passbook {
    public static void getPassbook(Customer c)
    {
        TransactionDAO tdao=new TransactionDAOImpl();
        List<Transaction>
passbook=tdao.getTransaction(c.getAccno());
        Iterator<Transaction> it=passbook.iterator();
        while(it.hasNext())
        {
            Transaction t=it.next();
            System.out.println("<------------------------------------
---->");
            System.out.println("Transaction 
ID:"+t.getTransactionId());
//+"||"+"Account_Number:"+t.getUser()+"||"+"Transaction Date");
            System.out.println("Account_Number:"+t.getRec_acc());
            System.out.println("Transaction 
Type:"+t.getTransaction());
            System.out.println("Transaction Date:"+t.getDate());
            if(t.getTransaction().equals("CREDITED"))
            {
                System.out.println("+"+t.getAmount());
                System.out.println("Current 
Balance:"+t.getBalance());
                System.out.println("<--------------------------------
-------->");
            }
            else
            {
                System.out.println("-"+t.getAmount());
                System.out.println("Current 
Balance:"+t.getBalance());
```

```
               System.out.println("<-----------------------------
-------->");
            }
        }
    }
}
```

TRANSACTIONDAOIMPL (TrnasactionDAOImpl) CLASS

```
package com.Bank.DAO;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

import com.Bank.Connection.Connector;
import com.Bank.DTO.Transaction;

public class TransactionDAOImpl implements TransactionDAO{
```

```java
    private Connection con;

    public TransactionDAOImpl()
    {
        this.con=Connector.requestConnection();
    }

    @Override
    public boolean insertTransaction(Transaction t) {
        return false;
    }

    @Override
    public List getTransaction(long user) {

        return null;
    }

}

//Override the below logic in TransactionDAOImpl class
```

```
@Override
    public List getTransaction(long user) {
        PreparedStatement ps=null;
        ResultSet rs=null;
        String query="SELECT * FROM PASSBOOK WHERE USER_ACC=? ORDER
BY TRAN_DATE desc";
        Transaction t=null;
        ArrayList<Transaction> passbook=new
ArrayList<Transaction>();
    try {
            ps=con.prepareStatement(query);
            ps.setLong(1, user);
            rs=ps.executeQuery();
            while(rs.next())
            {
                t=new Transaction();
                t.setTransactionId(rs.getLong(1));
                t.setUser(rs.getLong(2));
                t.setRec_acc(rs.getLong(3));
                t.setDate(rs.getDate(4));
                t.setTransaction(rs.getString(5));
                t.setAmount(rs.getDouble(6));
            t.setBalance(rs.getDouble(7));
            passbook.add(t);
        }
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return passbook;
}
```