

# **JAVA AWT BASED- REGISTRATION FORM- SQL CONNECTIVITY USING JDBC**

**A**

*Report*

*Submitted in partial fulfilment of the  
Requirements for the award of the Degree of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

**I.V.PRANAV KUMAR <1602-18-737-116>**



**Department of Information Technology  
Vasavi College of Engineering (Autonomous)  
(Affiliated to Osmania University)  
Ibrahimbagh, Hyderabad-31**

**2020**

## BONAFIDE CERTIFICATE

This is to certify that this project report titled “AUTOMATED FINGERPRINT BASED ATM(LINKING AADHARCARD)” is the bona fide mini project work of Mr. I.V. PRANAV KUMAR bearing hall ticket number 1602-18-737-116 under the guidance of B. Leelavathy during 4th semester B.E for the academic year 2019-2020.

External Examiner

Internal Examiner

B.LEELAVATHY

Assistant Professor

Department Of Information Technology

# Automated Fingerprint Based Atm.

## **AIM:**

To create a Java GUI for ATM which is based on fingerprint by linking Aadhar card. We have created different tables related to banks, ATM, and Aadhar card. We a database of the tables where we perform update, delete and insert.

These values are to be updated in the database using JDBC connectivity.

## **Abstract:**

The growth in electronic transactions has resulted in a greater demand for fast and accurate user identification and authentication. Access codes for buildings, banks accounts and computer systems often use personal identification numbers (PIN's) for identification and security clearances. Conventional method of identification based on possession of ID cards or exclusive knowledge like a social security number or a password are not all together reliable. An embedded fingerprint biometric authentication scheme for automated teller machine (ATM) banking systems is proposed in this paper. In this scheme, a fingerprint biometric technique is fused with the ATM for person authentication to ameliorate the security level.

## **Introduction:**

### REQUIREMENT ANALYSIS

List of tables:

- BANK
- AADHAR
- VERIFICATION DATA
- ATM
- TRANSACTIONS

List of attributes with their domain types:

BANK:

Bank Name:bname –Varchar2(20)

Name:username-varchar2(20)

Amount:amount-number(20)

Account Pin: accpin-number()

Account Id:accid- number()

AADHAR:

Aadhar number:anum-number()

Name: name-varchar2(20)

Gender:gender-varchar(10)

Address:address-varchar2(20)

Father name:fname-varchar2(20)

Verification Data:

FingerPrints: fingerprints-varchar()

Name:name-varchar2(20)

Fingertype:fingertype-varchar2(20)

ATM:

Bank      Name:aname-varchar2(20)

Atm Card Number-acnum-number()

TRANSACTIONS:

Date: tdate-number()

Transaction Amount: transamount-number()

Time: time-number()

Account id: acid-number().

### **SOFTWARE USED:**

Java Eclipse, Oracle 11g Database, Java SE version 7, SQL\*Plus.

**Eclipse:** It is an integrated development environment (IDE) used in computer programming. It contains a base workspace and an extensible plug in system for customizing the environment. The Eclipse software development kit (SDK), which include java development tools is meant for java developers.

**SQL \*plus:** SQL \*plus is a command line tool proprietary to oracle. You can send SQL Queries to the server using the tool. It can also help you format the result of query. SQL is the query language that is used to communicate with the oracle server to access and modify data.

## **Java AWT:**

Java AWT (Abstract Window Toolkit) is an API to develop GUI or window-based

applications in java.

Java AWT components are platform-dependent i.e. components are displayed according

to the view of operating system. AWT is heavyweight i.e. its components are using the

resources of OS.

The java.awt package provides classes for AWT API such as TextField, Label, TextArea,

RadioButton, CheckBox, Choice, List etc.

## **Java Swing:**

Java Swing is a part of Java Foundation Classes (JFC) which was designed for enabling large-scale enterprise development of Java applications. Java Swing is a set of APIs that provides graphical user interface (GUI) for Java programs. Java Swing is also known as Java GUI widget toolkit.

Java Swing or Swing was developed based on earlier APIs called Abstract Windows Toolkit (AWT). Swing provides richer and more sophisticated GUI components than AWT. The GUI components are ranging from a simple label to complex tree and table. Besides emulating look and feel of various platforms, Swing also provides *the pluggable look and feel* to allow look and feel of Java programs independent from the underlying platform.

## **SQL:**

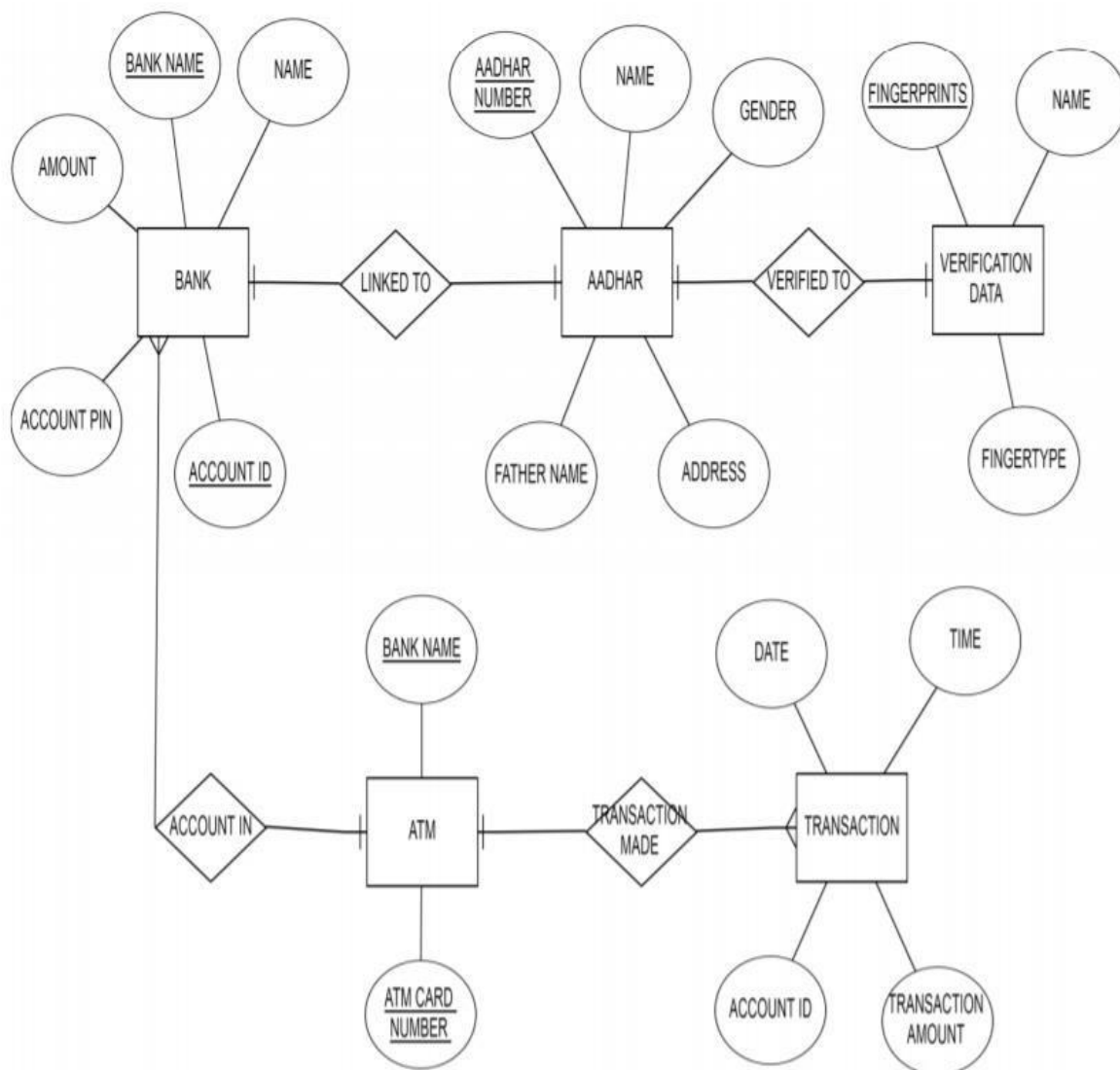
Structure Query Language(SQL) is a database query language used for storing and

managing data in Relational DBMS. SQL was the first commercial language introduced

for E.F Codd's Relational model of database. Today almost all RDBMS (MySQL , Oracle, Infomix , Sybase, MS Access) use SQL as the standard database query language. SQL is

used to perform all types of data operations in RDBMS.

### **ER DIAGRAM:**



### **CREATION OF TABLE:**

### **DDL AND DML COMMANDS :**

```
SQL> CREATE TABLE BANK (BNAME NUMBER, USERNAME VARCHAR2(20), ACCID
```

```
NUMBER,ACCPIN NUMBER,AMOUNT NUMBER);
```

Table created.

```
SQL> DESC BANK
```

Name	Null? Type
BNAME	NUMBER
USERNAME	VARCHAR2(20)
ACCID	NUMBER
ACCPIN	NUMBER
AMOUNT	NUMBER

```
SQL> ALTER TABLE BANK ADD(PRIMARY KEY(BNAME));
```

Table altered.

```
SQL> DESC BANK
```

Name	Null? Type
BNAME	NOT NULL NUMBER
USERNAME	VARCHAR2(20)
ACCID	NUMBER
ACCPIN	NUMBER
AMOUNT	NUMBER

```
SQL> ALTER TABLE BANK MODIFY(BNAME VARCHAR2(20));
```

Table altered.

```
SQL> DESC BANK
```

Name	Null? Type
BNAME	NOT NULL VARCHAR2(20)



USERNAME	VARCHAR2(20)
ACCID	NUMBER
ACCPIN	NUMBER
AMOUNT	NUMBER

SQL> DESC BANK

Name	Null?	Type
BNAME	NOT NULL	VARCHAR2(20)
USERNAME		VARCHAR2(20)
ACCID		NUMBER
ACCPIN		NUMBER
AMOUNT		NUMBER

SQL> INSERT INTO BANK

VALUES('&BNAME','&USERNAME',&ACCID,&ACCPIN,&AMOUNT);

Enter value for bname: SBI

Enter value for username: PRANAV

Enter value for accid: 1602116

Enter value for accpin: 116

Entervalueforamount: 10000

old 1: INSERT INTO BANK

VALUES('&BNAME','&USERNAME',&ACCID,&ACCPIN,&AMOUNT)

new 1: INSERT INTO BANK VALUES('SBI','PRANAV',1602116,116,10000)

SQL> INSERT INTO BANK

VALUES('&BNAME','&USERNAME',&ACCID,&ACCPIN,&AMOUNT);

Enter value for bname: HDFC

Enter value for username: ABHI

Entervalueforaccid: 1602061

Enter value for accpin: 061

Enter value for amount: 20000

old 1: INSERT INTO BANK

VALUES('&BNAME','&USERNAME',&ACCID,&ACCPIN,&AMOUNT)

new 1: INSERT INTO BANK VALUES('HDFC','ABHI',1602061,061,20000)

1 row created.

```
SQL> /
Enter value for bname: ICIC
Enter value for username: BADRINATH
Enter value for accid: 1602066 Enter
value for accpin: 066
Enter value for amount: 30000
old 1:INSERTINTOBANK
VALUES('&BNAME','&USERNAME','&ACCID','&ACCPIN','&AMOUNT')
new 1: INSERT INTO BANK VALUES('ICIC','BADRINATH',1602066,066,30000)
```

1row created.

```
SQL> /
Enter value for bname: PNB
Enter value for username: YASHO
Enter value for accid: 1602120
Enter value for accpin: 120
Enter value for amount: 40000
old 1:INSERTINTOBANK
VALUES('&BNAME','&USERNAME','&ACCID','&ACCPIN','&AMOUNT')
new 1: INSERT INTO BANK VALUES('PNB','YASHO',1602120,120,40000)
```

1 row created.

```
SQL>          INSERT          INTO          BANK
VALUES('&BNAME','&USERNAME','&ACCID','&ACCPIN','&AMOUNT');
Enter value for bname: SBI
Enter value for username: VARUN
Enter value for accid: 1602115
Enter value for accpin: 115
Enter value for amount: 12000
old 1:INSERT INTOBANK
VALUES('&BNAME','&USERNAME','&ACCID','&ACCPIN','&AMOUNT')
new 1: INSERT INTO BANK VALUES('SBI','VARUN',1602115,115,12000)
```

1 row created.

```
SQL> SELECT * FROM BANK;
```

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
SBI	PRANAV	1602116	116	10000
HDFC	ABHI	1602061	61	20000
ICIC	BADRINATH	1602066	66	30000
PNB	YASHO	1602120	120	40000
SBI	VARUN	1602115	115	12000

1 row created.

```
SQL> CREATE TABLE AADHARCARD(ANUM NUMBER PRIMARY KEY,NAME
VARCHAR2(20),GENDER VARCHAR(10),ADDRESS VARCHAR2(50),FNAME
VARCHAR2(20));
```

Table created.

```
SQL>          INSERT          INTO          AADHARCARD
VALUES(&ANUM,&NAME','&GENDER','&ADDRESS','&FNAME');
```

Enter value for anum: 1861

Enter value for name: ABHI

Enter value for gender: M

Enter value for address: SUNCITY

Enter value for fname: SRINU

```
old          1:          INSERT          INTO          AADHARCARD
VALUES(&ANUM,&NAME','&GENDER','&ADDRESS','&FNAME')
```

```
new 1: INSERT INTO AADHARCARD VALUES(1861,'ABHI','M','SUNCITY','SRINU')
```

1 row created.

```
SQL> /
```

Enter value for anum: 1866 Enter

value for name: BADRINATH Enter

value for gender: M

Enter value for address: GAJWEL

Enter value for fname: VENU

```
old    1:    INSERT    INTO    AADHARCARD
VALUES(&ANUM,&NAME','&GENDER','&ADDRESS','&FNAME')
new    1:    INSERT    INTO    AADHARCARD
VALUES(1866,'BADRINATH','M','GAJWEL','VENU')
```

1row created.

SQL> /

Enter value for anum:18115

Enter value for name: VARUN

Enter value for gender: M

Enter value for address: JANGON

Entervalueforfname: RAMU

```
old    1:    INSERT    INTO    AADHARCARD
VALUES(&ANUM,&NAME','&GENDER','&ADDRESS','&FNAME')
new    1:    INSERT    INTO    AADHARCARD
VALUES(18115,'VARUN','M','JANGON','RAMU')
```

1row created.

SQL> /

Enter value for anum: 18116

Enter value for name: PRANAV

Enter value for gender: M

Entervalueforaddress: AMEERPET

Entervalueforfname: SRINIVAS

```
old1:INSERTINTOAADHARCARD
VALUES(&ANUM,&NAME','&GENDER','&ADDRESS','&FNAME')
new    1:    INSERT    INTO    AADHARCARD
VALUES(18116,'PRANAV','M','AMEERPET','SRINIVAS')
```

1row created.

SQL> /

Enter value for anum: 118120

Enter value for name: YASHO

Enter value for gender: M

Enter value for address: NAMPALLY

Enter value for fname: RAMU

old 1: INSERT INTO AADHARCARD  
VALUES(&ANUM,&NAME,&GENDER,&ADDRESS,&FNAME)

new 1: INSERT INTO AADHARCARD  
VALUES(118120,'YASHO','M','NAMPALLY','RAMU')

1 row created.

SQL> SELECT \* FROM AADHARCARD;

ANUM	NAME	GENDER	ADDRESS	FNAME
1861	ABHI	M	SUNCITY	SRINU
1866	BADRINATH	M	GAJWEL	VENU
18115	VARUN	M	JANGON	RAMU

ANUM	NAME	GENDER	ADDRESS	FNAME
18116	PRANAV	M	AMEERPET	SRINIVAS
118120	YASHO	M	NAMPALLY	RAMU

SQL> DESC VERIFICATIONDATA

Name	Null? Type
FINGERPRINT	NOT NULL VARCHAR2(20)
NAME	VARCHAR2(20)
FINGERTYPE	VARCHAR2(20)

```
SQL> INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE');
Enter value for fingerprint: =
Enter value for name: PRANAV
Enter value for fingertype: INDEX
old 1: INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE')
new 1: INSERT INTO VERIFICATIONDATA VALUES('=', 'PRANAV', 'INDEX')
```

1 row created.

```
SQL> /
Enter value for fingerprint: )
Enter value for name: ABHI
Enter value for fingertype: INDEX
old 1: INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE')
new 1: INSERT INTO VERIFICATIONDATA VALUES(')', 'ABHI', 'INDEX')
```

1 row created.

```
SQL> /
Enter value for fingerprint: @
Enter value for name: BADRINATH
Enter value for fingertype: THUMB
old 1: INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE')
new 1: INSERT INTO VERIFICATIONDATA VALUES('@', 'BADRINATH', 'THUMB')
```

1 row created.

```
SQL> /
Enter value for fingerprint: ?
Enter value for name: YASHO
Enter value for fingertype: RING FINGER
old 1: INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE')
new 1: INSERT INTO VERIFICATIONDATA VALUES('?', 'YASHO', 'RING FINGER')

1 row created.
```

```
SQL> /
Enter value for fingerprint: >
Enter value for name: YASHO
Enter value for fingertype: THUMB
old 1: INSERT INTO VERIFICATIONDATA
VALUES('&FINGERPRINT','&NAME','&FINGERTYPE')
new 1: INSERT INTO VERIFICATIONDATA VALUES('>', 'YASHO', 'THUMB')

1 row created.
```

```
SQL> SELECT * FROM VERIFICATIONDATA;
```

	FINGERPRINT	NAME	FINGERTYPE
=		PRANAV	INDEX
)		ABHI	INDEX
@		BADRINATH	THUMB
?		YASHO	RING FINGER
>		YASHO	THUMB

```
SQL> ALTER TABLE ATM DROP(AMOUNT);
```

Table altered.

```
SQL> INSERT INTO ATM VALUES('&ANAME',&ACNUM);
Enter value for aname: SBI
```

Entervalue for acnum: 116  
old 1:INSERTINTOATMVALUES('&ANAME',&ACNUM)  
new 1:INSERTINTOATMVALUES('SBI',116)

1row created.

SQL> /  
Enter value for aname: HDFC  
Enter value for acnum:61  
old 1:INSERTINTOATMVALUES('&ANAME',&ACNUM)  
new 1:INSERTINTOATMVALUES('HDFC',61)

1row created.

SQL> /  
Enter value for aname: ICIC  
Enter value for acnum:66  
old 1:INSERTINTOATMVALUES('&ANAME',&ACNUM)  
new 1:INSERTINTOATMVALUES('ICIC',66)

1row created.

SQL> /  
Enter value for aname: PNB  
Entervalue for acnum: 120  
old 1:INSERTINTOATMVALUES('&ANAME',&ACNUM)  
new 1:INSERTINTOATMVALUES('PNB',120)

1row created.

SQL> /  
Enter value for aname: SBH  
Entervalue for acnum: 115  
old 1:INSERTINTOATMVALUES('&ANAME',&ACNUM)  
new 1:INSERTINTOATMVALUES('SBH',115)

1 row created.



SQL> SELECT \* FROM ATM;

ANAME	ACNUM
-------	-------

SBI	116
-----	-----

HDFC	61
------	----

ICIC	66
------	----

PNB	120
-----	-----

SBH	115
-----	-----

SQL> UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=12;

Enter value for accid: 1602116

old 1: UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=12

new 1: UPDATE TRANSACTION SET ACCID=1602116 WHERE TIME=12

1 row updated.

SQL> /

Enter value for accid:

old 1: UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=12

new 1: UPDATE TRANSACTION SET ACCID= WHERE TIME=12

UPDATE TRANSACTION SET ACCID= WHERE TIME=12

\*

ERROR at line 1:

ORA-00936: missing expression

SQL> UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=13;

Enter value for accid: 1602115

old 1: UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=13

new 1: UPDATE TRANSACTION SET ACCID=1602115 WHERE TIME=13

1 row updated.

SQL> UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=14;

Enter value for accid: 1602061

old 1: UPDATE TRANSACTION SET ACCID=&ACCID WHERE TIME=14

new 1: UPDATE TRANSACTION SET ACCID=1602061 WHERE TIME=14

1 row updated.

SQL> SELECT \* FROM TRANSACTION;

TDATE	TIME	TRANSAMONUT	ACCID
23012000	12	10000	1602116
24012000	13	10000	1602115
25012000	14	200000	1602061

### **Java-SQL Connectivity using JDBC:**

Java Database Connectivity (JDBC) is an application programming interface (API) for

the programming language Java, which defines how a client may access a database. It is a

Java-based data access technology used for Java database connectivity. It is part of

the Java Standard Edition platform, from Oracle Corporation. It provides methods to

query and update data in a database and is oriented towards relational databases.

The connection to the database can be performed using Java programming (JDBC API)

as:

```
private void connToDb()
```

```
{
```

```

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

    connection=DriverManger.getConnection("jdbc:oracle:thin:@localhost:1522:xe",
"pranav","pranav23");

    statement=connection.createStatement();
}
catch(SQLException connectException)
{
    System.out.println(connectException.getMessage());
    System.out.println(connectException.getSQLState());
    System.out.println(connectException.getErrorCode());
    System.exit(1);
}
catch(Exception e)
{
    System.err.println("Unable to find and load driver");
    System.exit(1);
}
}

```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

## Program For Main GUI:

```
package      dbms;

import javax.swing.*;

import java.awt.*;

import  java.awt.event.ActionEvent;

import java.awt.event.ActionListener;


public class home extends JFrame{

private JPanel panel1;

private JButton btnMycity; private

JButton  btnLandmarks; private

JButton  btnServices; private

JButton  btnHotels; private

JButton btnMetrostations; private

JButton btnTouristplaces; private

JButton btnHas_Hotels;

private JButton btnHas_metrostations;

private JButton btnHas_touristplaces;

public home()

{

panel1=new JPanel(new GridLayout(3,3,15,15));

btnMetrostations=new JButton("Bank");
```

```
btnHas_Hotels=new JButton("Linkedto");
btnHas_metrostations=new JButton("AadharCard");
btnHas_touristplaces=new JButton("VerifiedTo");
btnHotels=new JButton("VerificationData");
btnLandmarks=new JButton("AccountIn");
btnMycity=new JButton("Atm");
btnServices=new JButton("TransactionMade");
btnTouristplaces=new JButton("Transaction");
panel1.setBackground(Color.BLACK);
panel1.add(btnMycity);
panel1.add(btnServices);
panel1.add(btnLandmarks);
panel1.add(btnHotels);
panel1.add(btnMetrostations);
panel1.add(btnTouristplaces);
panel1.add(btnHas_Hotels);
panel1.add(btnHas_metrostations);
panel1.add(btnHas_touristplaces);
this.setVisible(true);
this.setSize(600,400);
this.setDefaultCloseOperation(3);
this.add(panel1,BorderLayout.CENTER);
btnHas_Hotels.addActionListener(new ActionListener() {

@Override
```

```
public void actionPerformed(ActionEvent e) {  
    // TODO Auto-generated method stub  
    new LinkedToView();  
    dispose();  
}  
});  
btnHas_metrostations.addActionListener(new ActionListener() {
```

```
@Override  
public void actionPerformed(ActionEvent e) {  
    // TODO Auto-generated method stub  
    new AadharCardView();  
    dispose();  
}  
});  
btnHas_touristplaces.addActionListener(new ActionListener() {
```

```
@Override  
public void actionPerformed(ActionEvent e) {  
    // TODO Auto-generated method stub  
    new VerifiedToView();  
    dispose();  
}  
});  
btnHotels.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new      VerificationDataView();

    dispose();

}

});

btnLandmarks.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new AccountInView();

    dispose();

}

});

btnMetrostations.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new BankView();

    dispose();

}
```

```

});

btnMycity.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {
// TODO Auto-generated method stub
new AtmView();
dispose();
}

});

btnServices.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {
// TODO Auto-generated method stub
new TransactionMadeView();
dispose();
}

});

btnTouristplaces.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {
// TODO Auto-generated method stub
new TransactionView();

```



```
dispose();  
  
}  
  
});  
  
}  
  
public static void main(String args[])  
{  
  
new home();  
  
}  
  
}
```

## **Program For Insert BANK:**

```
package      dbms;  
  
import javax.swing.*;  
  
import      java.sql.*;  
  
import java.awt.*;  
  
import      java.awt.event.ActionEvent;  
  
import java.awt.event.ActionListener;  
  
  
public class BankInsert extends JFrame{  
  
    private JPanel pn;  
  
    private      JLabel      bn;  
  
    private      JLabel      nn;  
  
    private      JLabel      amt;  
  
    private      JLabel      ap;  
  
    private      JLabel      aid;  
  
    private JTextField tpn;
```

```
private JTextField tbn;
private JTextField tnn;
private JTextField tamt;
private JTextField tap;
private JTextField taid;
private JButton btn; private
JMenu Transaction;
private JMenu TransactionMade;
private JMenu ATM;
private JMenu AccountIn;
private JMenu Bank; private
JMenu LinkedTo; private
JMenu Aadhar; private
JMenu VerifiedTo; private
JMenu VerifiedData; private
JMenuBar menubar; private
JMenuItem insert1; private
JMenuItem delete1; private
JMenuItem view1; private
JMenuItem insert2; private
JMenuItem delete2; private
JMenuItem view2; private
JMenuItem insert3; private
JMenuItem delete3; private
JMenuItem view3; private
JMenuItem insert4; private
JMenuItem delete4; private
JMenuItem view4;
```

```
private JMenuItem insert5;

private JMenuItem delete5;

private JMenuItem view5;

private JMenuItem insert6;

private JMenuItem delete6;

private JMenuItem view6;

private JMenuItem insert7;

private JMenuItem delete7;

private JMenuItem view7;

private JMenuItem insert8;

private JMenuItem delete8;

private JMenuItem view8;

private JMenuItem insert9;

private JMenuItem view9;

private JMenuItem delete9;
```

```
public BankInsert() {

    pn = new JPanel(new FlowLayout());

    bn = new JLabel("Bank Name");

    tbn = new JTextField(20);

    nn = new JLabel("Name");

    tnn = new JTextField(20);

    amt = new JLabel("Amount");

    tamt = new JTextField(20);

    ap = new JLabel("Account Pin");

    tap = new JTextField(20);

    aid = new JLabel("Account Id");

    taid = new JTextField(20);
```

```
btn      =      new      JButton("Insert");

menubar=new          JMenuBar();

Transaction=new JMenu("Transaction");

insert1=new      JMenuItem("Insert");

view1=new      JMenuItem("View");

delete1=new      JMenuItem("Delete");

Transaction.add(insert1);

Transaction.add(view1);

Transaction.add(delete1);

TransactionMade=new JMenu("TransactionMade");

insert2=new JMenuItem("Insert");

view2=new      JMenuItem("View");

delete2=new JMenuItem("Delete");

TransactionMade.add(insert2);

TransactionMade.add(view2);

TransactionMade.add(delete2);

ATM=new      JMenu("ATM");

insert3=new JMenuItem("Insert");

view3=new      JMenuItem("View");

delete3=new JMenuItem("Delete");

ATM.add(insert3);

ATM.add(view3);

ATM.add(delete3);

AccountIn=new JMenu("AccountIn");

insert4=new      JMenuItem("Insert");

view4=new      JMenuItem("View");

delete4=new      JMenuItem("Delete");

AccountIn.add(insert4);
```

```
AccountIn.add(view4);

AccountIn.add(delete4);

Bank=new      JMenu("Bank");

insert5=new JMenuItem("Insert");

view5=new    JMenuItem("View");

delete5=new JMenuItem("Delete");

Bank.add(insert5);

Bank.add(view5);

Bank.add(delete5);

LinkedTo=new JMenu("LinkedTo");

insert6=new JMenuItem("Insert");

view6=new    JMenuItem("View");

delete6=new JMenuItem("Delete");

LinkedTo.add(insert6);

LinkedTo.add(view6);

LinkedTo.add(delete6);

Aadhar=new    JMenu("Aadhar");

insert7=new JMenuItem("Insert");

view7=new    JMenuItem("View");

delete7=new JMenuItem("Delete");

Aadhar.add(insert7);

Aadhar.add(view7);

Aadhar.add(delete7);

VerifiedTo=new JMenu("VerifiedTo");

insert8=new    JMenuItem("Insert");

view8=new      JMenuItem("View");

delete8=new    JMenuItem("Delete");

VerifiedTo.add(insert8);
```

```
VerifiedTo.add(view8);

VerifiedTo.add(delete8);

VerifiedData=new JMenu("VerifiedData");

insert9=new JMenuItem("Insert");

view9=new JMenuItem("View");

delete9=new JMenuItem("Delete");

VerifiedData.add(insert9);

VerifiedData.add(view9);

VerifiedData.add(delete9);

this.setSize(600,600);

this.setVisible(true);

this.setDefaultCloseOperation(3);

this.add(pn);

//pn.add(lbl);

pn.add(bn);

pn.add(tbn);

pn.add(nn);

pn.add(tnn);

pn.add(amt);

pn.add(tamt);

pn.add(ap);

pn.add(tap);

pn.add(aid);

pn.add(taid);

pn.add(btn);

this.setJMenuBar(menubar);

menubar.add(Transaction);

menubar.add(TransactionMade);
```

```

menubar.add(ATM);

menubar.add(AccountIn);

menubar.add(Bank);

menubar.add(LinkedTo);

menubar.add(Aadhar);

menubar.add(VerifiedTo);

menubar.add(VerifiedData);

insert1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TransactionInsert();

        dispose();

    }

});

view1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TransactionView();

        dispose();

    }

});

delete1.addActionListener(new ActionListener() {

    @Override

```

```

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new TransactionDelete();

            dispose();

        }

    });

    insert2.addActionListener(new ActionListener() {

```

```

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new TransactionMadeInsert();

            dispose();

        }

    });

    view2.addActionListener(new ActionListener() {

```

```

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new TransactionMadeView();

            dispose();

        }

    });

    delete2.addActionListener(new ActionListener() {

```

```

        @Override

        public void actionPerformed(ActionEvent e) {

```



```

        // TODO Auto-generated method stub
        new TransactionMadeDelete();
        dispose();
    }

});

insert3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new AtmInsert();

        dispose();
    }

});

view3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new AtmView();
        dispose();
    }

});

delete3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

```

```

        // TODO Auto-generated method stub

        new AtmDelete();

        dispose();

    }

});

insert4.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AccountInInsert();

        dispose();

    }

});

view4.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AccountInView();

        dispose();

    }

});

delete4.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

```

```

        new AccountInDelete();

        dispose();
    }

});

insert5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new BankInsert();

        dispose();

    }

});

view5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new BankView();

        dispose();

    }

});

delete5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new BankDelete();

```

```

        dispose();
    }
});

insert6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new LinkedToInsert();

        dispose();

    }

});

view6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new LinkedToView();

        dispose();

    }

});

delete6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new LinkedToDelete();

        dispose();

    }

});

```

```

        }

    });

    insert7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AadharCardInsert();

            dispose();

        }

    });

    view7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AadharCardView();

            dispose();

        }

    });

    delete7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AadharCardDelete();

            dispose();

        }

    });

```

```
});

insert8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new VerifiedToInsert();

        dispose();

    }

});

view8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new VerifiedToView();

        dispose();

    }

});

delete8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new VerifiedToDelete();

        dispose();

    }

});
```

```

insert9.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent arg0) {

        // TODO Auto-generated method stub

        new      VerificationDataInsert();

        dispose();

    }

});

view9.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent arg0) {

        // TODO Auto-generated method stub

        new      VerificationDataView();

        dispose();

    }

});

delete9.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent arg0) {

        // TODO Auto-generated method stub

        new      VerificationDataDelete();

        dispose();

    }

});

```

```

});

btn.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        //      TODO      Auto-generated      method      stub

        if(taid.getText().compareTo("")==0 || tnn.getText().compareTo("")==0
|| tamt.getText().compareTo("")==0 || tap.getText().compareTo("")==0 ||
taid.getText().compareTo("")==0)

        {

            JOptionPane.showMessageDialog(null, "Enter All fieldS");

        }

        else

        {

            try{

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

                Connection con=DriverManager.getConnection(

                    "jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

                Statement stmt=con.createStatement();

                String bname=tbn.getText();

                String username=tnn.getText();

                String      accid=taid.getText();

                if(checkaccountid(accid))

                {

                    JOptionPane.showMessageDialog(null,"Enter
Vaild AccountId");

```



```

        throw new Exception();
    }

    String accpin=tap.getText();

    String amount=tamt.getText();

    int x=stmt.executeUpdate("Insert into bank
values("+bname+"",""+username+"",""+accid+"",""+accpin+"",""+amount+"");

    con.commit();

    System.out.println("Inserted rows:"+x);

    tbn.setText("");

    tnn.setText("");

    taid.setText("");

    tap.setText("");

    tamt.setText("");


    con.close();

}

}

});

}

public boolean checkaccountid(String accountid)
{
    try {

        Integer.parseInt(accountid);

        return false;

    }

```

```

        catch(Exception e)
        {
            return true;
        }
    }

    public static void main(String[] args)
    {
        new BankInsert();
    }
}

```

## Program For Update Of Hotels Table:

```

package        dbms;

import javax.swing.*;

import        java.sql.*;

import java.awt.*;

import        java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.ResultSet;


public class BankView extends JFrame {

    private JPanel pn;

    private JLabel bn;

    private JLabel nn;

    private JLabel amt;

    private JLabel ap;

```

```
private JLabel aid; private
TextField tpn; private
TextField tbn; private
TextField tnn; private
TextField tamt; private
TextField tap; private
TextField taidd; private
JLabel lblsac; private
TextField txtsac; private
JButton btnvw; private
JButton btn; private
JTextArea txtmes;

private JMenu Transaction;
private JMenu TransactionMade;
private JMenu ATM;
private JMenu AccountIn;
private JMenu Bank; private
JMenu LinkedTo; private
JMenu Aadhar; private
JMenu VerifiedTo; private
JMenu VerifiedData; private
JMenuBar menubar; private
JMenuItem insert1; private
JMenuItem delete1; private
JMenuItem view1; private
JMenuItem insert2;
```

```
private JMenuItem delete2;
private JMenuItem view2;
private JMenuItem insert3;
private JMenuItem delete3;
private JMenuItem view3;
private JMenuItem insert4;
private JMenuItem delete4;
private JMenuItem view4;
private JMenuItem insert5;
private JMenuItem delete5;
private JMenuItem view5;
private JMenuItem insert6;
private JMenuItem delete6;
private JMenuItem view6;
private JMenuItem insert7;
private JMenuItem delete7;
private JMenuItem view7;
private JMenuItem insert8;
private JMenuItem delete8;
private JMenuItem view8;
private JMenuItem insert9;
private JMenuItem view9;
private JMenuItem delete9;
public BankView() {
    pn = new JPanel(new FlowLayout());
    txtmes = new JTextArea(10,20);
```

```
lblsac = new JLabel("Enter Account Id:");

txtsac = new JTextField(20);

btnvw = new JButton("View");

bn = new JLabel("Bank Name");

tbn = new JTextField(20);

nn = new JLabel("Name"); tnn
= new JTextField(20); amt =
new JLabel("Amount"); tamt
= new JTextField(20);

ap = new JLabel("Account Pin");

tap = new JTextField(20);

aid = new JLabel("Account Id");

taid = new JTextField(20);

btn    =    new    JButton("Modify");

menubar=new          JMenuBar();

Transaction=new JMenu("Transaction");

insert1=new          JMenuItem("Insert");

view1=new            JMenuItem("View");

delete1=new          JMenuItem("Delete");

Transaction.add(insert1);

Transaction.add(view1);

Transaction.add(delete1);

TransactionMade=new JMenu("TransactionMade");

insert2=new JMenuItem("Insert");

view2=new    JMenuItem("View");

delete2=new JMenuItem("Delete");
```

```
TransactionMade.add(insert2);

TransactionMade.add(view2);

TransactionMade.add(delete2);

ATM=new          JMenu("ATM");

insert3=new JMenuItem("Insert");

view3=new    JMenuItem("View");

delete3=new JMenuItem("Delete");

ATM.add(insert3);

ATM.add(view3);

ATM.add(delete3);

AccountIn=new JMenu("AccountIn");

insert4=new    JMenuItem("Insert");

view4=new      JMenuItem("View");

delete4=new    JMenuItem("Delete");

AccountIn.add(insert4);

AccountIn.add(view4);

AccountIn.add(delete4);

Bank=new          JMenu("Bank");

insert5=new JMenuItem("Insert");

view5=new    JMenuItem("View");

delete5=new JMenuItem("Delete");

Bank.add(insert5);

Bank.add(view5);

Bank.add(delete5);

LinkedTo=new JMenu("LinkedTo");

insert6=new JMenuItem("Insert");
```

```
view6=new JMenuItem("View");
delete6=new JMenuItem("Delete");
LinkedTo.add(insert6);
LinkedTo.add(view6);
LinkedTo.add(delete6);

Aadhar=new JMenu("Aadhar");
insert7=new JMenuItem("Insert");
view7=new JMenuItem("View");
delete7=new JMenuItem("Delete");

Aadhar.add(insert7);
Aadhar.add(view7);
Aadhar.add(delete7);

VerifiedTo=new JMenu("VerifiedTo");
insert8=new JMenuItem("Insert");
view8=new JMenuItem("View");
delete8=new JMenuItem("Delete");

VerifiedTo.add(insert8);
VerifiedTo.add(view8);
VerifiedTo.add(delete8);

VerifiedData=new JMenu("VerifiedData");
insert9=new JMenuItem("Insert");
view9=new JMenuItem("View");
delete9=new JMenuItem("Delete");

VerifiedData.add(insert9);
VerifiedData.add(view9);
VerifiedData.add(delete9);
```

```
this.setSize(400,400);

this.setTitle("Bank      View");

this.setVisible(true);

this.setDefaultCloseOperation(3);

this.add(pn);

pn.add(txtmes);

pn.add(lblsac);

pn.add(txtsac);

pn.add(btnvw);

pn.add(bn);

pn.add(tbn);

pn.add(nn);

pn.add(tnn);

pn.add(amt);

pn.add(tamt);

pn.add(ap);

pn.add(tap);

pn.add(aid);

pn.add(taid);

pn.add(btn);

this.setJMenuBar(menubar);

menubar.add(Transaction);

menubar.add(TransactionMade);

menubar.add(ATM);

menubar.add(AccountIn);

menubar.add(Bank);
```



```
menubar.add(LinkedTo);
menubar.add(Aadhar);
menubar.add(VerifiedTo);
menubar.add(VerifiedData);
insert1.addActionListener(new ActionListener() {
```

```
    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new TransactionInsert();
        dispose();
    }
```

```
});
```

```
view1.addActionListener(new ActionListener() {
```

```
    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new TransactionView();
        dispose();
    }
```

```
});
```

```
delete1.addActionListener(new ActionListener() {
```

```
    @Override
    public void actionPerformed(ActionEvent e) {
```

```
        // TODO Auto-generated method stub
        new TransactionDelete();
        dispose();
    }

    });
```

```
insert2.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new TransactionMadeInsert();
        dispose();
    }

});
```

```
view2.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new TransactionMadeView();
        dispose();
    }

});
```

```
delete2.addActionListener(new ActionListener() {
```

```
    @Override
```

```
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new TransactionMadeDelete();  
            dispose();  
        }  
    });  
    insert3.addActionListener(new ActionListener() {
```

```
        @Override  
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new AtmInsert();  
  
            dispose();  
        }  
    });  
    view3.addActionListener(new ActionListener() {
```

```
        @Override  
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new AtmView();  
            dispose();  
        }  
    });  
    delete3.addActionListener(new ActionListener() {
```

```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AtmDelete();

            dispose();

        }

    });

    insert4.addActionListener(new ActionListener() {
```

```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AccountInInsert();

            dispose();

        }

    });

    view4.addActionListener(new ActionListener() {
```

```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AccountInView();

            dispose();

        }

    });
```

```
delete4.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AccountInDelete();

        dispose();

    }

});

insert5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new BankInsert();

        dispose();

    }

});

view5.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new BankView();

        dispose();

    }

});
```

```
});  
  
delete5.addActionListener(new ActionListener() {  
  
    @Override  
  
    public void actionPerformed(ActionEvent e) {  
        // TODO Auto-generated method stub  
        new BankDelete();  
        dispose();  
    }  
});
```

```
insert6.addActionListener(new ActionListener() {  
  
    @Override  
  
    public void actionPerformed(ActionEvent e) {  
        // TODO Auto-generated method stub  
        new LinkedToInsert();  
        dispose();  
    }  
});
```

```
view6.addActionListener(new ActionListener() {  
  
    @Override  
  
    public void actionPerformed(ActionEvent e) {  
        // TODO Auto-generated method stub  
        new LinkedToView();  
        dispose();  
    }  
});
```

```

        }

    });

    delete6.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new LinkedToDelete();

            dispose();

        }

    });

    insert7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AadharCardInsert();

            dispose();

        }

    });

    view7.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AadharCardView();

```

```

        dispose();
    }
});

delete7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new AadharCardDelete();

        dispose();

    }

});

insert8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new VerifiedToInsert();

        dispose();

    }

});

view8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

```



```

        new VerifiedToView();
        dispose();
    }

});

delete8.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new VerifiedToDelete();
        dispose();
    }

});

insert9.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub
        new      VerificationDataInsert();
        dispose();
    }

});

view9.addActionListener(new ActionListener() {

    @Override

```

```

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new      VerificationDataView();

            dispose();

        }

    });

    delete9.addActionListener(new ActionListener() {

```

```

        @Override

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new      VerificationDataDelete();

            dispose();

        }

    });

    try{

```

```

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

```

```

        Connection      con=DriverManager.getConnection(
        "jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

```

```

        Statement stmt=con.createStatement();

        ResultSet rs=stmt.executeQuery("select accid from bank ");

        while(rs.next())

```

```

    {

        txtmes.append(rs.getString(1)+"\n");

    }

    con.close();

    }catch(Exception ex){ System.out.println(ex);}
btnvw.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {

        //      TODO      Auto-generated      method      stub
        if(txtsac.getText().compareTo("")==0)

            JOptionPane.showMessageDialog(null, "Enter Account
id");

        else
        {

            try{

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

                Connection con=DriverManager.getConnection(

"jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

                Statement stmt=con.createStatement();

```

```

        ResultSet rs=stmt.executeQuery("select * from
bank where accid="+txtsac.getText()+"");

        while(rs.next())
        {

            tbn.setText(rs.getString(1));

            tnn.setText(rs.getString(2));

            taid.setText(rs.getString(3));

            tap.setText(rs.getString(4));

            tamt.setText(rs.getString(5));

        }

        con.close();

    }catch(Exception ex){ System.out.println(ex);}

    }

}

});

btn.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        if(taid.getText().compareTo("")==0 ||

tnn.getText().compareTo("")==0 || tamt.getText().compareTo("")==0 ||

tap.getText().compareTo("")==0 || taid.getText().compareTo("")==0)

        {

            JOptionPane.showMessageDialog(null, "Enter All fieldS");

```

```

    }
    else
    {
        try{

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

            Connection con=DriverManager.getConnection(

                "jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

            Statement stmt=con.createStatement();

            String bname=tbn.getText();

            String username=tnn.getText();

            String    accid=taid.getText();

            String    accpin=tap.getText();

            String amount=tamt.getText();

            int  x=stmt.executeUpdate("update bank set
bname='"+bname+"',username='"+username+"',accid='"+accid+"',accpin='"+accpin+"',amount='"+
amount+" where accid='"+txtsac.getText());

            con.commit();

            System.out.println("Updated rows:"+x);

            tbn.setText("");

            tnn.setText("");

            taid.setText("");

            tap.setText("");

            tamt.setText("");

            txtmes.setText("");

            txtsac.setText("");

            ResultSet rs=stmt.executeQuery("select accid
from bank ");

```

```

        while(rs.next())
        {
            txtmes.append(rs.getString(1)+"\n");
        }

        con.close();

    }catch(Exception ex){ System.out.println(ex);}

    }

    });

}

public static void main(String[] args)
{
    new BankView();

}

}

```

## Program for Delete BANK:

```

package      dbms;

import      java.sql.*;

import javax.swing.*;

import java.awt.*;

import      java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

```

```
public class BankDelete extends JFrame {  
    private JPanel pn;  
    private JLabel bn;  
    private JLabel nn;  
    private JLabel amt;  
    private JLabel ap;  
    private JLabel aid;  
    private JTextField tpn;  
    private JTextField tbn;  
    private JTextField tnn;  
    private JTextField tamt;  
    private JTextField tap;  
    private JTextField taid;  
    private JButton btn;  
    private JTextArea txtmes;  
    private JLabel lblsac;  
    private JTextField txtsac;  
    private JButton btnvw;  
    private JMenu Transaction;  
    private JMenu TransactionMade;  
    private JMenu ATM;  
    private JMenu AccountIn;  
    private JMenu Bank;  
    private JMenu LinkedTo;  
    private JMenu Aadhar;
```

```
private JMenu VerifiedTo;
private JMenu VerifiedData;
private JMenuBar menubar;
private JMenuItem insert1;
private JMenuItem delete1;
private JMenuItem view1;
private JMenuItem insert2;
private JMenuItem delete2;
private JMenuItem view2;
private JMenuItem insert3;
private JMenuItem delete3;
private JMenuItem view3;
private JMenuItem insert4;
private JMenuItem delete4;
private JMenuItem view4;
private JMenuItem insert5;
private JMenuItem delete5;
private JMenuItem view5;
private JMenuItem insert6;
private JMenuItem delete6;
private JMenuItem view6;
private JMenuItem insert7;
private JMenuItem delete7;
private JMenuItem view7;
private JMenuItem insert8;
private JMenuItem delete8;
```



```

private JMenuItem view8;

private JMenuItem insert9;

private JMenuItem view9;

private JMenuItem delete9;

public BankDelete() {

    pn = new JPanel(new FlowLayout());

    txtmes = new JTextArea(10,20);

    lblsac = new JLabel("Enter Account Id:");

    txtsac = new JTextField(20);

    btnvw = new JButton("View");

    bn = new JLabel("Bank Name");

    tbn = new JTextField(20);

    nn = new JLabel("Name"); tnn
= new JTextField(20); amt =
new JLabel("Amount"); tamt
= new JTextField(20);

    ap = new JLabel("Account Pin");

    tap = new JTextField(20);

    aid = new JLabel("Account Id");

    taid = new JTextField(20);

    btn    =    new    JButton("Delete");

    menubar=new                JMenuBar();

    Transaction=new JMenu("Transaction");

    insert1=new        JMenuItem("Insert");

    view1=new          JMenuItem("View");

    delete1=new JMenuItem("Delete");

```

```
Transaction.add(insert1);

Transaction.add(view1);

Transaction.add(delete1);

TransactionMade=new JMenu("TransactionMade");

insert2=new JMenuItem("Insert");

view2=new JMenuItem("View");

delete2=new JMenuItem("Delete");

TransactionMade.add(insert2);

TransactionMade.add(view2);

TransactionMade.add(delete2);

ATM=new JMenu("ATM");

insert3=new JMenuItem("Insert");

view3=new JMenuItem("View");

delete3=new JMenuItem("Delete");

ATM.add(insert3);

ATM.add(view3);

ATM.add(delete3);

AccountIn=new JMenu("AccountIn");

insert4=new JMenuItem("Insert");

view4=new JMenuItem("View");

delete4=new JMenuItem("Delete");

AccountIn.add(insert4);

AccountIn.add(view4);

AccountIn.add(delete4);

Bank=new JMenu("Bank");

insert5=new JMenuItem("Insert");
```

```
view5=new JMenuItem("View");
delete5=new JMenuItem("Delete");
Bank.add(insert5);
Bank.add(view5);
Bank.add(delete5);
LinkedTo=new JMenu("LinkedTo");
insert6=new JMenuItem("Insert");
view6=new JMenuItem("View");
delete6=new JMenuItem("Delete");
LinkedTo.add(insert6);
LinkedTo.add(view6);
LinkedTo.add(delete6);
Aadhar=new JMenu("Aadhar");
insert7=new JMenuItem("Insert");
view7=new JMenuItem("View");
delete7=new JMenuItem("Delete");
Aadhar.add(insert7);
Aadhar.add(view7);
Aadhar.add(delete7);
VerifiedTo=new JMenu("VerifiedTo");
insert8=new JMenuItem("Insert");
view8=new JMenuItem("View");
delete8=new JMenuItem("Delete");
VerifiedTo.add(insert8);
VerifiedTo.add(view8);
VerifiedTo.add(delete8);
```

```
VerifiedData=new JMenu("VerifiedData");

insert9=new      JMenuItem("Insert");

view9=new        JMenuItem("View");

delete9=new      JMenuItem("Delete");

VerifiedData.add(insert9);

VerifiedData.add(view9);

VerifiedData.add(delete9);

this.setSize(400,400);

this.setTitle("Bank      View");

this.setVisible(true);

this.setDefaultCloseOperation(3);

this.add(pn);

pn.add(txtmes);

pn.add(lblsac);

pn.add(txtsac);

pn.add(btnvw);

pn.add(bn);

pn.add(tbn);

pn.add(nn);

pn.add(tnn);

pn.add(amt);

pn.add(tamt);

pn.add(ap);

pn.add(tap);

pn.add(aid);

pn.add(taid);
```

```
pn.add(btn);

this.setJMenuBar(menubar);

menubar.add(Transaction);

menubar.add(TransactionMade);

menubar.add(ATM);

menubar.add(AccountIn);

menubar.add(Bank);

menubar.add(LinkedTo);

menubar.add(Aadhar);

menubar.add(VerifiedTo);

menubar.add(VerifiedData);

insert1.addActionListener(new ActionListener() {
```

```
    @Override
```

```
    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TransactionInsert();

        dispose();

    }

```

```
});
```

```
view1.addActionListener(new ActionListener() {
```

```
    @Override
```

```
    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new TransactionView();

    }

```

```

        dispose();
    }
});

delete1.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new TransactionDelete();

        dispose();

    }

});

insert2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub
        new TransactionMadeInsert();

        dispose();

    }

});

view2.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

```

```

        new TransactionMadeView();
        dispose();
    }

});

delete2.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new TransactionMadeDelete();
        dispose();
    }
});

insert3.addActionListener(new ActionListener() {

    @Override
    public void actionPerformed(ActionEvent e) {
        // TODO Auto-generated method stub
        new AtmInsert();

        dispose();
    }
});

view3.addActionListener(new ActionListener() {

    @Override

```

```
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new AtmView();  
            dispose();  
        }  
    }  
});  
delete3.addActionListener(new ActionListener() {
```

```
        @Override  
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new AtmDelete();  
            dispose();  
        }  
    }  
});  
insert4.addActionListener(new ActionListener() {
```

```
        @Override  
        public void actionPerformed(ActionEvent e) {  
            // TODO Auto-generated method stub  
            new AccountInInsert();  
            dispose();  
        }  
    }  
});  
view4.addActionListener(new ActionListener() {
```



```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AccountInView();

            dispose();

        }

    });

    delete4.addActionListener(new ActionListener() {
```

```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new AccountInDelete();

            dispose();

        }

    });

    insert5.addActionListener(new ActionListener() {
```

```
        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new BankInsert();

            dispose();

        }

    });

    view5.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new BankView();

    dispose();

}

});

delete5.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new BankDelete();

    dispose();

}

});

insert6.addActionListener(new ActionListener() {
```

```
@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    new LinkedToInsert();

    dispose();

}

});
```

```
view6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new LinkedToView();

        dispose();

    }

});

delete6.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new LinkedToDelete();

        dispose();

    }

});

insert7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AadharCardInsert();

        dispose();

    }

});
```

```
});

view7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AadharCardView();

        dispose();

    }

});

delete7.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new AadharCardDelete();

        dispose();

    }

});

insert8.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent e) {

        // TODO Auto-generated method stub

        new VerifiedToInsert();

        dispose();

    }

});
```

```

        }

    });

    view8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new VerifiedToView();

            dispose();

        }

    });

    delete8.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent e) {

            // TODO Auto-generated method stub

            new VerifiedToDelete();

            dispose();

        }

    });

    insert9.addActionListener(new ActionListener() {

        @Override

        public void actionPerformed(ActionEvent arg0) {

            // TODO Auto-generated method stub

            new VerificationDataInsert();

```

```

        dispose();

    }

});

view9.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent arg0) {

        // TODO Auto-generated method stub

        new      VerificationDataView();

        dispose();

    }

});

delete9.addActionListener(new ActionListener() {

    @Override

    public void actionPerformed(ActionEvent arg0) {

        // TODO Auto-generated method stub

        new      VerificationDataDelete();

        dispose();

    }

});

try{

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

```

```

Connection          con=DriverManager.getConnection(
"jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select accid from bank ");
while(rs.next())
{
    txtmes.append(rs.getString(1)+"\n");

}

con.close();

}catch(Exception ex){ System.out.println(ex);}
btnvw.addActionListener(new ActionListener() {

@Override
public void actionPerformed(ActionEvent e) {
    //          TODO          Auto-generated          method          stub
    if(txtsac.getText().compareTo("")==0)
        JOptionPane.showMessageDialog(null, "Enter Account
id");

    else
    {
        try{

```

```

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

        Connection con=DriverManager.getConnection(

            "jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

        Statement      stmt=con.createStatement();
        ResultSet rs=stmt.executeQuery("select * from
bank where accid="+txtsac.getText()+"");

        while(rs.next())
        {

            tbn.setText(rs.getString(1));
            tnn.setText(rs.getString(2));
            taid.setText(rs.getString(3));
            tap.setText(rs.getString(4));
            tamt.setText(rs.getString(5));

        }

        con.close();

    }catch(Exception ex){ System.out.println(ex);}

    }

}

});

btn.addActionListener(new ActionListener() {

```



```

@Override

public void actionPerformed(ActionEvent e) {

    // TODO Auto-generated method stub

    if(taid.getText().compareTo("")==0 ||
tnn.getText().compareTo("")==0 || tamt.getText().compareTo("")==0 ||
tap.getText().compareTo("")==0 || taid.getText().compareTo("")==0)

    {

        JOptionPane.showMessageDialog(null, "Enter All fieldS");

    }

    else

    {

        try{

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

            Connection con=DriverManager.getConnection(

                "jdbc:oracle:thin:@localhost:1521:xe","pranav","pranav23");

            Statement stmt=con.createStatement();

            String bname=tbn.getText();

            String username=tnn.getText();

            String accid=taid.getText();

            String accpin=tap.getText();

            String amount=tamt.getText();

            int x=stmt.executeUpdate("delete from bank

where accid="+txtsac.getText());

            con.commit();

```

```

        System.out.println("Deleted rows:"+x);

        tbn.setText("");
        tnn.setText("");
        taid.setText("");
        tap.setText("");
        tamt.setText("");
        txtmes.setText("");
        txtsac.setText("");

        ResultSet rs=stmt.executeQuery("select accid
from bank ");

        while(rs.next())
        {
            txtmes.append(rs.getString(1)+"\n")
        }

        con.close();

    }catch(Exception ex){ System.out.println(ex);}

    }

}

});

}

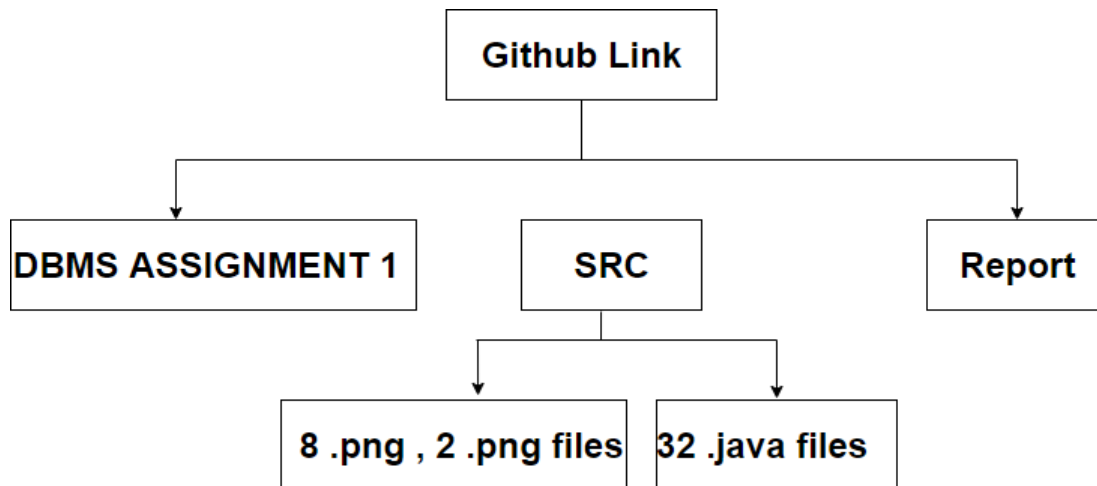
public static void main(String[] args)
{
    new BankDelete();
}

}

```

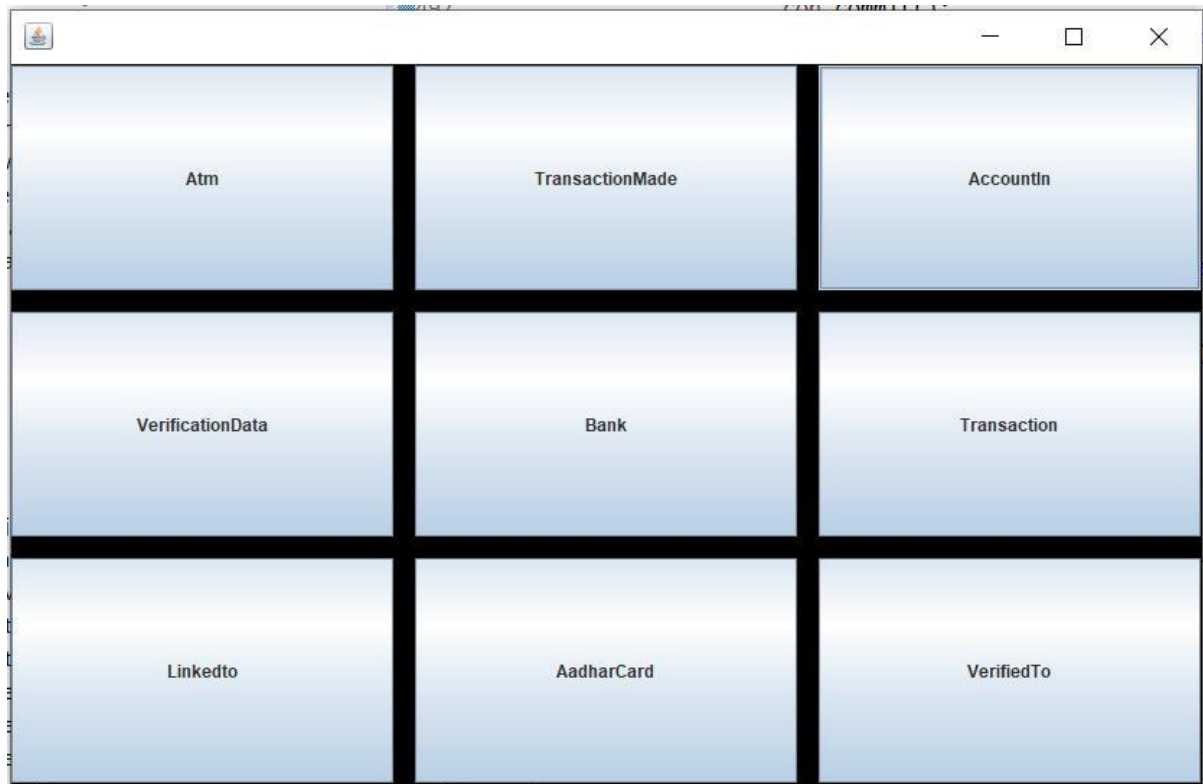
## GITHUB LINK AND FOLDER STRUCTURE:

<https://github.com/Pranav2323/dbms-project>

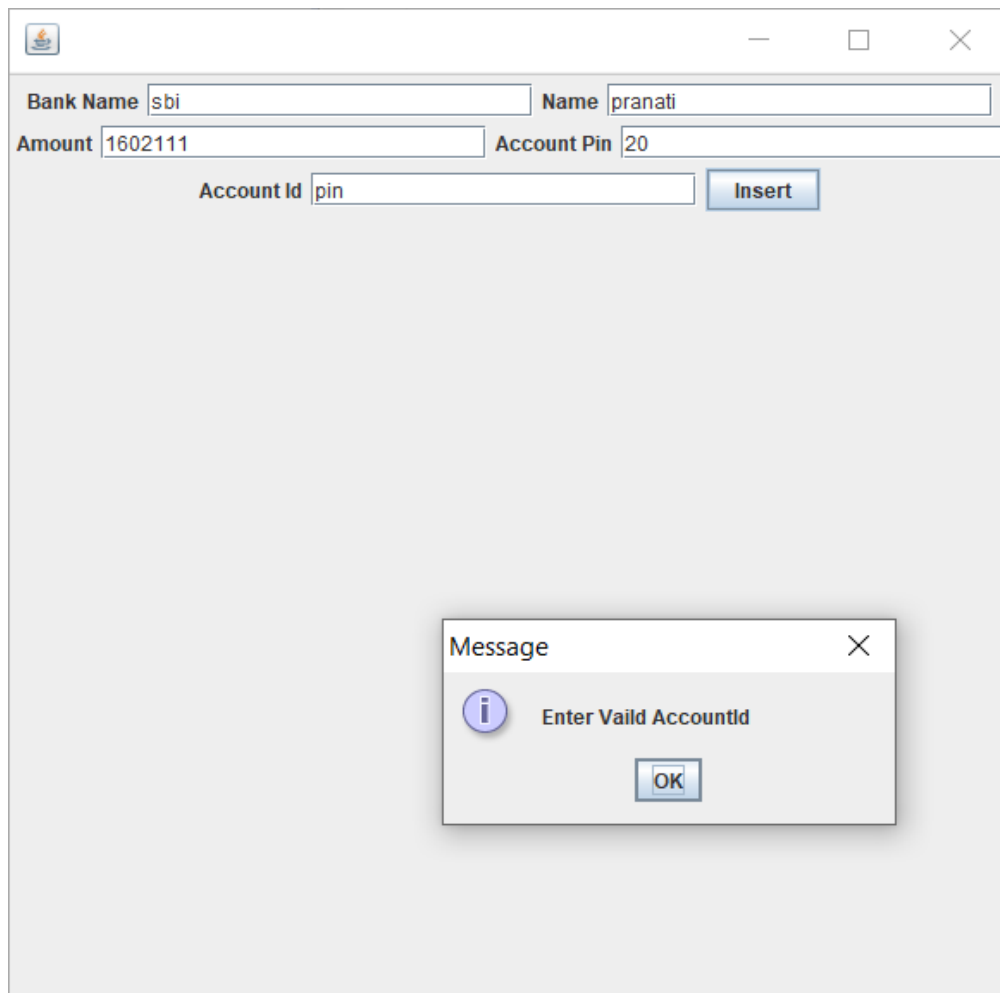


**OUTPUT :**

**GUI For Main Frame:**



## ERROR MESSAGES:




The screenshot displays a web application interface with a light gray background. At the top, there is a header bar with a small logo on the left and standard window controls (minimize, maximize, close) on the right. Below the header, the main content area contains a form with several input fields and a button. The form fields are labeled "Bank Name", "Name", "Amount", "Account Pin", and "Account Id". The "Bank Name" field contains the text "sbi", the "Name" field contains "pranati", the "Amount" field contains "1602111", the "Account Pin" field contains "20", and the "Account Id" field contains "pin". To the right of the "Account Id" field is a blue button labeled "Insert". In the center of the page, there is a modal dialog box titled "Message" with a close button (X) in the top right corner. The dialog box has a light gray background and contains a blue circular icon with a white lowercase "i" on the left. To the right of the icon, the text "Enter Vaild AccountId" is displayed. At the bottom center of the dialog box is a blue button labeled "OK".

Bank Name  Name

Amount  Account Pin

Account Id

**Message** [X]

 Enter Vaild AccountId



Bank Name  Name   
Amount  Account Pin

Account Id

Message

i

Enter All fieldS

OK

# Java GUI Insert in Bank:

Transaction TransactionMade ATM AccountIn Bank LinkedTo Aadhar VerifiedTo VerifiedData

Bank Name  Name

Amount  Account Pin

Account Id

Run SQL Command Line

SQL\*Plus: Release 11.2.0.2.0 Production on Wed Apr 29 21:45:09 2020

Copyright (c) 1982, 2010, Oracle. All rights reserved.

SQL> conn pranav/pranav23

Connected.

SQL> select \* from bank;

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
sbi	pranav	1602116	116	10000
hdfc	abhi	1602061	61	20000
sbi	varun	1602115	115	12000
pnb	yasho	1602120	120	40000
icic	badrinath	1602066	66	30000

SQL> select \* from bank;

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
sbi	pranav	1602116	116	10000
hdfc	abhi	1602061	61	20000
sbi	varun	1602115	115	12000
sbi	pranati	1602111	11	20000
pnb	yasho	1602120	120	40000
icic	badrinath	1602066	66	30000

6 rows selected.

SQL>

## Update GUI for BANK Table:

Bank View

Transac	Transaction	ATI	Accou	Bar	Linker	Aadh	Verifie	Verified
1602061								
1602066								
1602111								
1602115								
1602116								
1602120								

Enter Account Id:

1602111

View Bank Name

sbi Name

pranati Amount

20000 Account Pin

111 Account Id

1602111

Modify

Run SQL Command Line

```
pnb yasho 1602120 120 40000
icic badrinath 1602066 66 30000

SQL> select * from bank;

BNAME      USERNAME      ACCID  ACCPIN  AMOUNT
-----
sbi         pranav        1602116  116    10000
hdfc        abhi          1602061  61     20000
sbi         varun         1602115  115    12000
sbi         pranati       1602111  11     20000
pnb         yasho         1602120  120    40000
icic        badrinath     1602066  66     30000

6 rows selected.

SQL> select * from bank;

BNAME      USERNAME      ACCID  ACCPIN  AMOUNT
-----
sbi         pranav        1602116  116    10000
hdfc        abhi          1602061  61     20000
sbi         varun         1602115  115    12000
sbi         pranati       1602111  111    20000
pnb         yasho         1602120  120    40000
icic        badrinath     1602066  66     30000

6 rows selected.

SQL>
```



## GUI for Before Delete operation on BANK Table:

Bank View

Transac Transaction ATI Accou Bar Linker Aadh Verifie Verified

1602061  
1602066  
1602115  
1602116  
1602120

Enter Account Id:

1602115

View

Bank Name

sbi Name

varun Amount

12000 Account Pin

115 Account Id

1602115

Delete

Run SQL Command Line

6 rows selected.

```
SQL> select * from;
```

```
select * from
```

```
*
```

```
ERROR at line 1:
```

```
ORA-00903: invalid table name
```

```
SQL> select * from bank;
```

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
sbi	pranav	1602116	116	10000
hdfc	abhi	1602061	61	20000
sbi	varun	1602115	115	12000
pnb	yasho	1602120	120	40000
icic	badrinath	1602066	66	30000

```
SQL>
```

## DataBase After Delete:

Run SQL Command Line

6 rows selected.

```
SQL> select * from;  
select * from  
      *
```

ERROR at line 1:  
ORA-00903: invalid table name

```
SQL> select * from bank;
```

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
sbi	pranav	1602116	116	10000
hdfc	abhi	1602061	61	20000
sbi	varun	1602115	115	12000
pnb	yasho	1602120	120	40000
icic	badrinath	1602066	66	30000

```
SQL> select * from bank;
```

BNAME	USERNAME	ACCID	ACCPIN	AMOUNT
sbi	pranav	1602116	116	10000
hdfc	abhi	1602061	61	20000
pnb	yasho	1602120	120	40000
icic	badrinath	1602066	66	30000

```
SQL> 
```

## GUI INSERT IN AADHARCARD TABLE:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing the results of a SQL query. The query is 'select \* from aadharcard;'. The results are presented in a table format with columns: ADDRESS, ANUM, NAME, GENDER, and FNAME. The data is as follows:

ADDRESS	ANUM	NAME	GENDER	FNAME
gajwel	1866	badrinath	m	venu
jangon	18115	varun	m	ramu
ameerpeth	18116	pranav	m	srinivas
hampally	118120	yasho	m	ramu
srinu	1861	abhiraj	m	suncity
srinu	1870	harsha	m	gutta

Below the table, it states '6 rows selected.' and the prompt 'SQL>' is visible.

On the right, a GUI application window is open. It has a title bar with standard Windows window controls. The application contains several input fields and an 'Insert' button. The fields are labeled: 'Aadhar Number' (with the value '1870'), 'Aadhar Name' (with the value 'harsha'), 'Gender' (with the value 'm'), 'Father Name' (with the value 'srinu'), and 'Address' (with the value 'gutta'). The 'Insert' button is located below the input fields.

### UPDATE GUI FOR ADDHARCARD TABLE:

The screenshot shows a Windows desktop environment. On the left, a 'Select Run SQL Command Line' window displays a table of user data. The table has columns: ANUM, NAME, GENDER, and FNAME. The data is as follows:

ANUM	NAME	GENDER	FNAME
1866	badrinath	m	venu
18115	varun	m	ramu
18116	pranav	m	srinivas
1861	abhiraj	m	suncity
1870	harsha	m	gutta

Below the table, it says '6 rows selected.' and 'SQL> /'.

On the right, a web browser window is open, displaying a form for adding or updating a user. The form has a header with tabs: Transaction, TransactionMade, ATM, AccountIn, Bank, LinkedTo, Aadhar, VerifiedTo, and VerifiedData. The 'Aadhar' tab is selected. The form contains the following fields:

- Transaction: 1861
- TransactionMade: 1866
- ATM: 1870
- AccountIn: 18115
- Bank: 18116
- LinkedTo: 118120
- Aadhar: (empty)
- VerifiedTo: (empty)
- VerifiedData: (empty)

Below the form, there are buttons for 'View', 'Add', 'Update', and 'Delete'. The 'View' button is highlighted. Below the buttons, there are fields for 'Gender', 'Father Name', and 'Address', each with a corresponding button: 'Gender' (m), 'Father Name' (langer houze), and 'Address' (srinu). The 'Modify' button is highlighted.

## GUI for Before Delete operation on AADHARCARD Table:

The screenshot shows a Windows desktop environment. On the left, a 'Select Run SQL Command Line' window displays the following SQL query and its results:

```
SQL> select * from aadharcard;
```

ANUM	NAME	GENDER	FNAME
1866	badrinath	m	venu
18115	varun	m	ramu
18116	pranav	m	srinivas
118120	yasho	m	ramu
1861	abhiraj	m	srinu
1870	harsha	m	gutta

On the right, a GUI application window titled 'Transaction' is open. It contains a list of Aadhaar numbers: 1801, 1895, 1870, 18115, 18116, 118120. Below the list is a search bar with the text 'Enter Aadhar Number:'. A 'View' button is next to the search bar. Below the search bar, there are input fields for 'Aadhar Number' (containing 1870), 'Aadhar Name' (containing Harsha), 'Gender' (containing m), 'Father Name' (containing gutta), and 'Address' (containing srinu). A 'Delete' button is located at the bottom right of the GUI window.

### DATABASE AFTER DELETE:

The screenshot shows a Windows desktop with two windows open. The background window is a 'Run SQL Command Line' application, which is a terminal window with a dark background. It displays SQL commands and their results. The first command is 'select \* from aadharcard;', which returns a list of 8 rows of Aadhar card data. The data is presented in a table format with columns for ANUM, NAME, GENDER, ADDRESS, and FNAME. The second command is 'select /', which returns the same 8 rows of data. The foreground window is a web application titled 'TransactionMade ATM Account Bank LinkedTo Aadhar VerifiedTo VerifiedData'. It has a light gray background and a white header bar. The header bar contains a list of tabs: Transaction, TransactionMade, ATM, Account, Bank, LinkedTo, Aadhar, VerifiedTo, and VerifiedData. The 'Aadhar' tab is currently selected. Below the header bar, there is a search bar with the text 'Enter Aadhar Number:'. To the right of the search bar is a 'View' button. Below the search bar, there are three input fields: 'Aadhar Name', 'Father Name', and 'Address'. To the right of the 'Aadhar Name' field is a 'Delete' button. The taskbar at the bottom of the screen shows various icons, including the Start button, a search bar, and several application icons. The system clock in the bottom right corner shows the time as 12:45 and the date as 30-05-2020.

## GUI INSERT FOR VERIFIEDDATA TABLE:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing the execution of SQL commands and the resulting data from a table named 'verificationdata'.

The SQL commands executed are:

```
SQL> select * from verificationdata;
```

The output shows a table with three columns: FINGERPRINT, NAME, and FINGERTYPE. The data is as follows:

FINGERPRINT	NAME	FINGERTYPE
0	abhi	index
?	yasho	ring finger
>	varun	thumb
!	pranav	thumb

The SQL commands executed are:

```
SQL> commit;
```

The output shows 'Commit complete.'

The SQL commands executed are:

```
SQL> select * from verificationdata;
```

The output shows a table with three columns: FINGERPRINT, NAME, and FINGERTYPE. The data is as follows:

FINGERPRINT	NAME	FINGERTYPE
0	abhi	index
@	badrinath	index
?	yasho	ring finger
>	varun	thumb
!	pranav	thumb

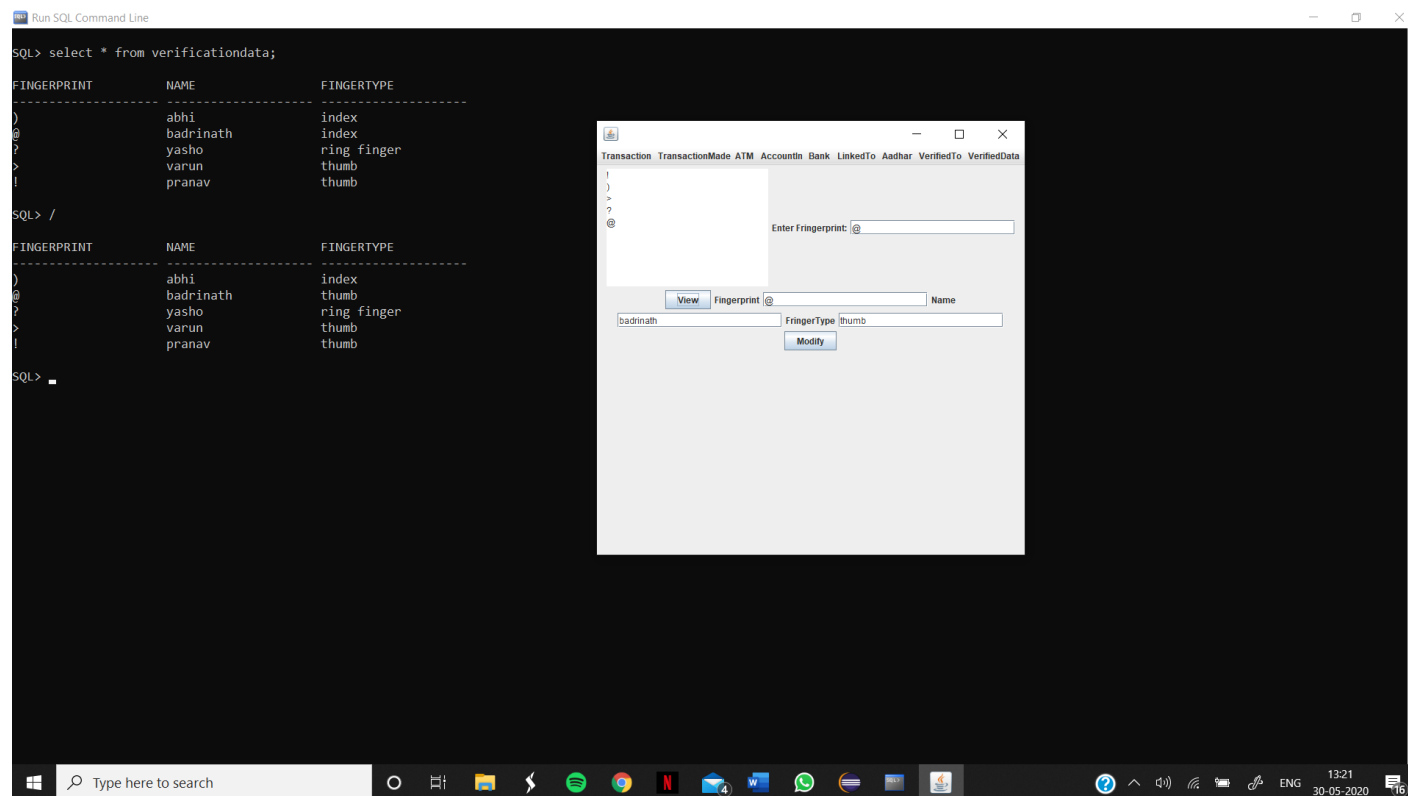
The SQL command executed is:

```
SQL>
```

On the right, a GUI application window is open. It has a title bar with a minus, maximize, and close button. The window contains two text input fields: 'Fingerprint' and 'Name'. The 'Fingerprint' field contains the value '0' and the 'Name' field contains the value 'badrinath'. Below these fields is a label 'Fingertype' followed by a dropdown menu showing 'index'. To the right of the dropdown is an 'Insert' button.

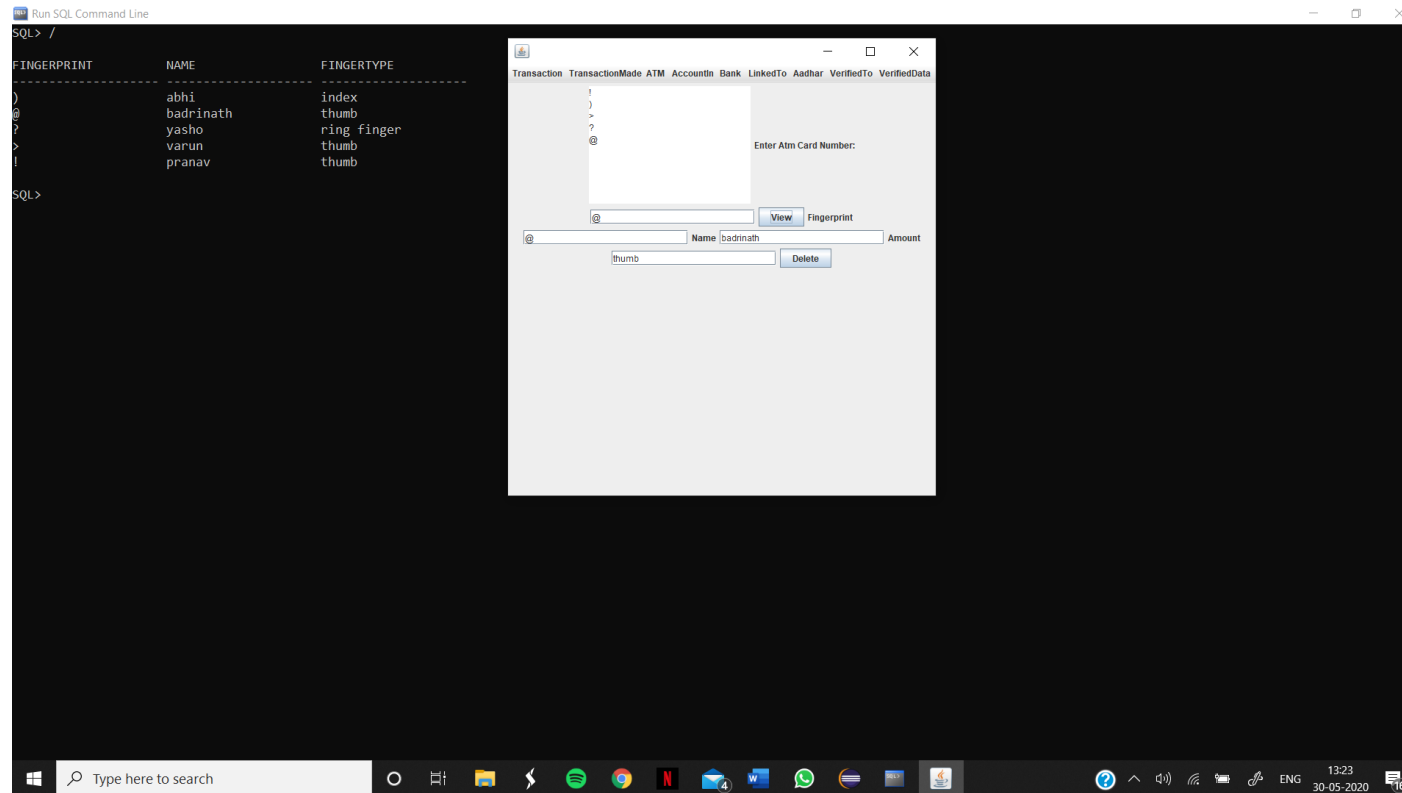
The Windows taskbar at the bottom shows the search bar with the text 'Type here to search', several application icons (including File Explorer, Edge, Spotify, Chrome, and others), and the system tray with the date and time '13:04 30-05-2020'.

# UPDATE GUI FOR VERIFICATIONDATA TABLE:





## GUI for Before Delete operation on VERIFICATIONDATA Table:



## DATABASE AFTER DELETE:

Run SQL Command Line

SQL> /

FINGERPRINT	NAME	FINGERTYPE
1	abhi	index
2	badrinath	thumb
3	yasho	ring finger
4	varun	thumb
5	pranav	thumb

SQL> /

FINGERPRINT	NAME	FINGERTYPE
1	abhi	index
2	yasho	ring finger
3	varun	thumb
4	pranav	thumb

SQL>

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
1								
2								
3								
4								
5								

Enter Atm Card Number:

Fingerprint

Name  Amount



Type here to search



ENG

13:24

30-05-2020

16

# INSERT FOR ATM TABLE:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing the results of a SQL query. The query is 'select \* from atm ;'. The results are displayed in a table with two columns: 'ANAME' and 'ACNUM'. The data is as follows:

ANAME	ACNUM
sbi	116
icic	66
pnb	120
sbh	115
hdfc	61

Below the first table, the command 'SQL> /' is entered. The results are repeated:

ANAME	ACNUM
sbi	116
icic	70
icic	66
pnb	120
sbh	115
hdfc	61

Below the second table, the message '6 rows selected.' is displayed. The command line prompt 'SQL>' is shown at the bottom.

On the right, an application window titled 'ATM' is open. It contains two input fields: 'Atm Name' with the value 'icic' and 'Atm Card Number' with the value '70'. An 'Insert' button is located to the right of the 'Atm Card Number' field.

The Windows taskbar at the bottom shows the search bar with the text 'Type here to search', several application icons (including File Explorer, Edge, Chrome, and various office and communication apps), and system tray icons on the right indicating the time as 13:27 and the date as 30-05-2020.

**UPDATE FOR ATM TABLE:**

The screenshot shows a terminal window on the left and a web application interface on the right.

**Terminal Window:**

```

Run SQL Command Line

ANAME                               ACNUM
-----
sbi                                  116
icic                                  70
icic                                  66
pnb                                   120
sbh                                   115
hdFc                                  61

6 rows selected.

SQL> /

ANAME                               ACNUM
-----
sbi                                  116
axis                                  70
icic                                  66
pnb                                   120
sbh                                   115
hdFc                                  61

6 rows selected.

SQL>

```

**Web Application Interface:**

The interface has a title bar with a minus, maximize, and close button. Below the title bar is a header row with the following labels: Transaction, TransactionMade, ATM, Accountn, Bank, LinkedTo, Aadhar, VerifiedTo, VerifiedData.

The main content area is divided into two sections:

- Left Section:** A list of bank names and their corresponding ACNUM values:
  - 61
  - 66
  - 70
  - 115
  - 115
  - 120
- Right Section:** A form titled "Enter Atm Card Number:". It contains a text input field with the value "70", a "View" button, and a label "Atm Name".

Below the main content area, there is a search bar with the text "axis" and a "Modify" button.

## GUI OF ATM TABLE BEFORE DELETION:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing a list of bank names and their corresponding ACNUM values. The list is as follows:

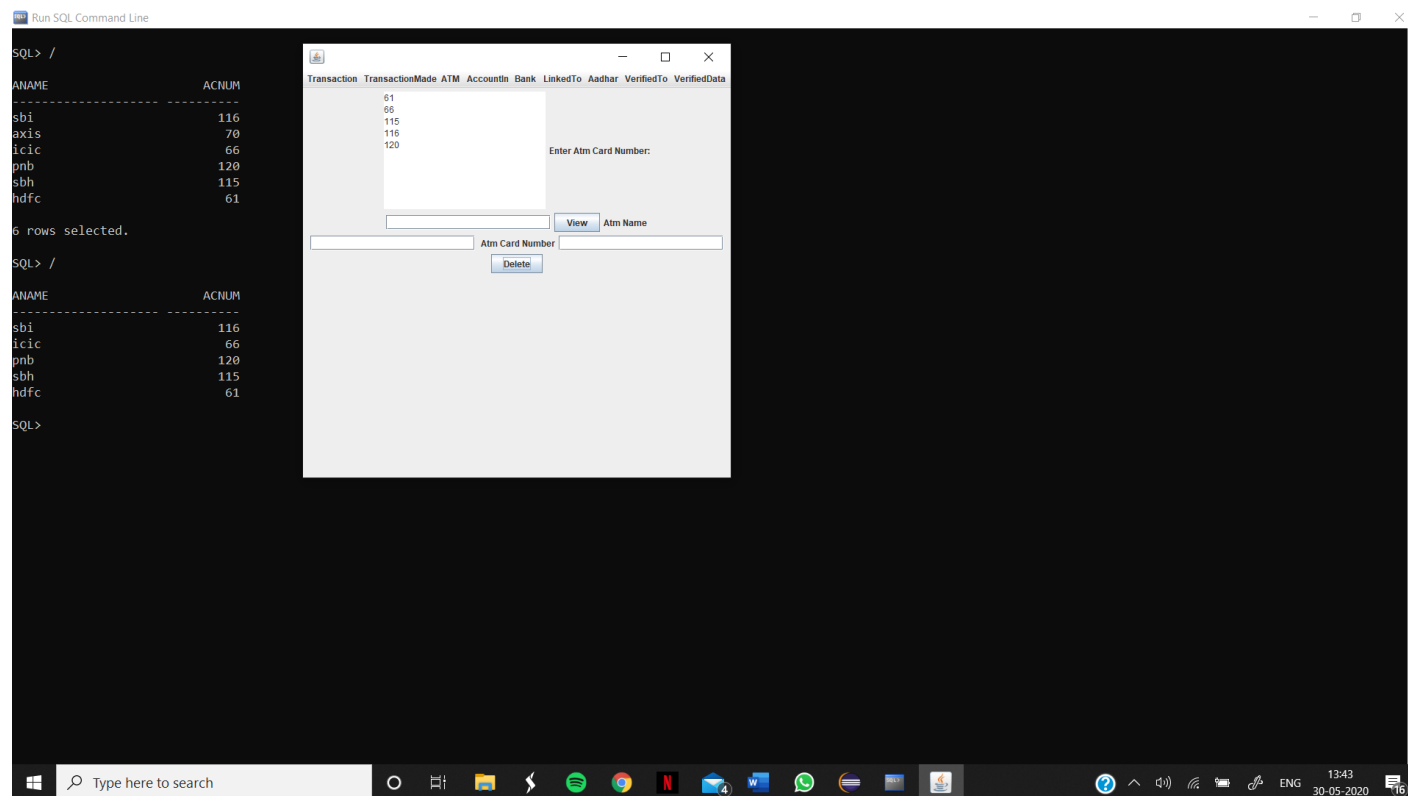
ANAME	ACNUM
sbi	116
axis	70
icic	66
pnb	120
sbh	115
hdfc	61

Below the list, it indicates '6 rows selected.' and shows the SQL prompt 'SQL> /'.

Overlaid on the right side of the SQL window is an application window titled 'Transaction'. This window contains a list of transaction IDs (81, 95, 70, 115, 115, 120) and a section for entering an ATM card number. The 'Enter Atm Card Number:' field is currently empty. Below this, there is a 'View' button and a label 'Atm Name'. At the bottom of the application window, there is a 'Modify' button and a label 'Atm Card Number' followed by the value '70'.

The Windows taskbar at the bottom shows various icons including the Start button, search bar, and several application icons. The system clock in the bottom right corner indicates the time is 13:41 on 30-05-2020.

## GUI AFTER DELETE:



### INSERT GUI FOR TRANSACTION TABLE:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing a SQL query and its results. The query is 'select \* from transaction;'. The results are presented in a table with four columns: TDATE, TIME, TRANSACTION, and ACCID. The data is as follows:

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061

Below the first table, the command 'SQL> /' is entered, and a second table is displayed with the same columns and data, including an additional row:

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061
20-05-2020	01:45 pm	20000	1602111

Below the second table, the message '6 rows selected.' is displayed, followed by the command 'SQL> \_'.

On the right, a web application window is open, showing a form for a transaction. The form has the following fields and values:

- Transaction: TransactionMade
- ATM: ATM
- AccountNo: AccountNo
- Bank: Bank
- LinkedTo: LinkedTo
- Aadhar: Aadhar
- VerifiedTo: VerifiedTo
- VerifiedData: VerifiedData
- Transaction Amount: 20000
- Date: 20-05-2020
- Time: 01:45 pm
- Account Id: 1602111

The 'Insert' button is highlighted.

The Windows taskbar at the bottom shows the search bar with the text 'Type here to search', and several application icons including File Explorer, Edge, Spotify, Chrome, Notepad, Mail, Word, WhatsApp, and the Start button.

# UPADTE GUI FOR TRANSACTION TABLE:

Run SQL Command Line

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061
20-05-2020	01:45 pm	20000	1602111

6 rows selected.

SQL> /

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061
21-05-2020	11:45 pm	20000	1602111

6 rows selected.

SQL>

Transaction TransactionMade ATM AccountIn Bank LinkedTo Aadhar VerifiedTo VerifiedData

1602061  
1602066  
1602111  
1602115  
1602116  
1602120

Enter Account Id: 1602111

View Transaction Amount 20000 Date

21-05-2020 Time 11:45 pm Account Id

1602111 Modify

Type here to search

13:50  
30-05-2020



# GUI OF TRANSACTION TABLE BEFORE DELETE:

The screenshot displays a Windows desktop environment. On the left, a 'Run SQL Command Line' window is open, showing a list of transactions with columns: DATE, TIME, TRANSACTION, and ACCID. The data is as follows:

DATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061
21-05-2020	11:45 pm	20000	1602111

Below the table, it states '6 rows selected.' and the prompt 'SQL>' is visible.

On the right, a GUI application window is open. It features a list of transaction IDs on the left: 1602061, 1602066, 1602111, 1602115, 1602116, and 1602120. A search bar labeled 'Enter Account Id:' contains the value '1602111'. Below this, there are input fields for 'Transaction Amount' (set to 20000) and 'Date' (set to 21-05-2020). A 'View' button is positioned between these fields. At the bottom, there is a 'Time' field (set to 11:45 pm) and an 'Account Id' field (set to 1602111), with a 'Delete' button to the right.

The Windows taskbar at the bottom shows the search bar, task view button, and several application icons. The system tray on the right indicates the time as 13:51 on 30-05-2020, with language set to ENG.

# DATABASE AFTER DELETE:

Run SQL Command Line

```
SQL> /
```

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061
21-05-2020	11:45 pm	20000	1602111

6 rows selected.

```
SQL> /
```

TDATE	TIME	TRANSACTION	ACCID
23-01-2000	12:00 pm	1000	1602116
24-01-2000	03:11 am	1000	1602115
01-01-2001	12:00 am	3000	1602066
06-10-2004	10:30 pm	1500	1602120
20-01-2001	03:04 pm	12500	1602061

```
SQL>
```

Transaction TransactionMade ATM AccountIn Bank LinkedTo Aadhar VerifiedTo VerifiedData

1602061  
1602066  
1602115  
1602116  
1602120

Enter Account Id:

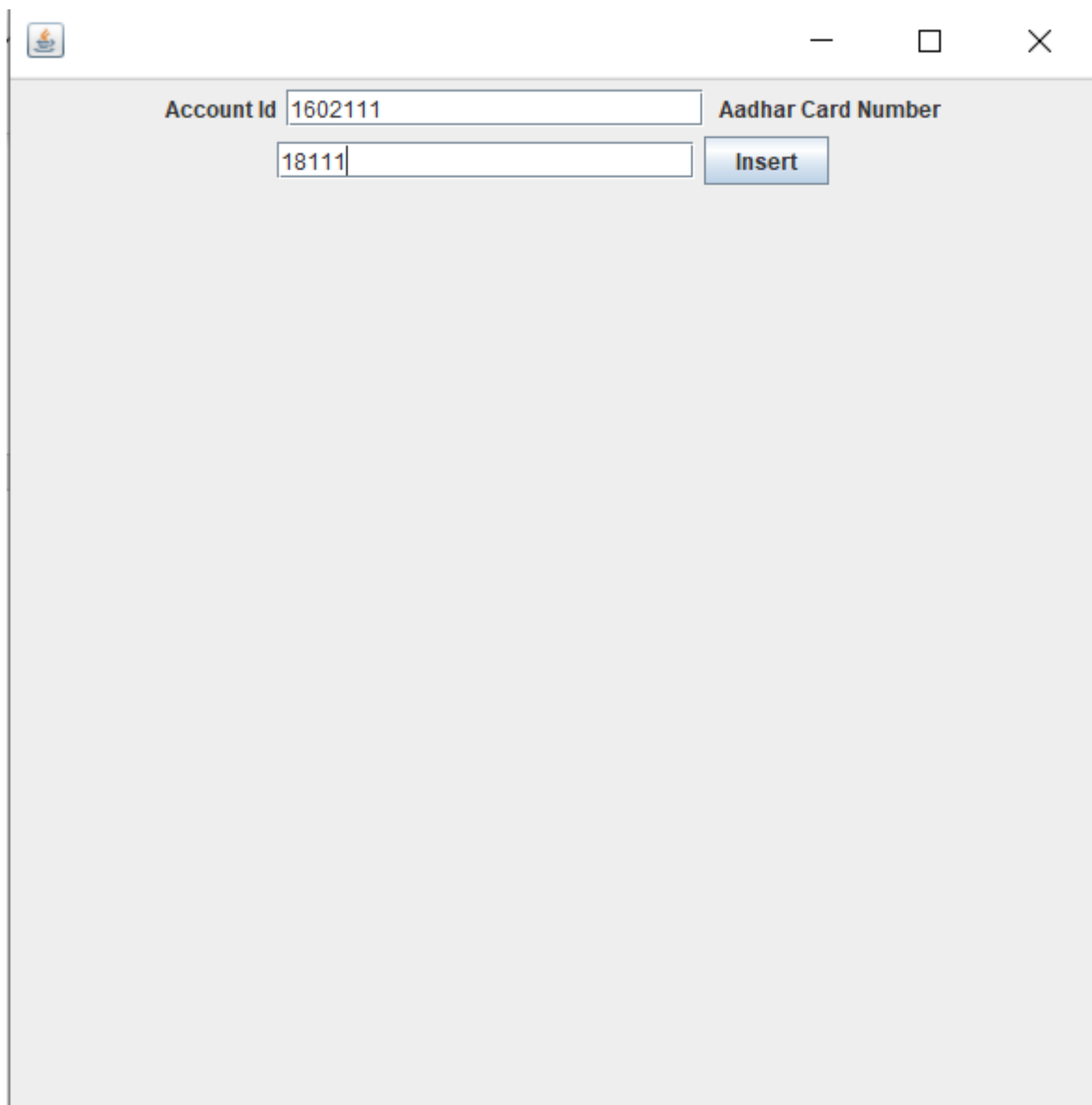
Transaction Amount  Date

Time  Account Id

Type here to search

13:52  
30-05-2020

## INSERT GUI PAGE FOR LINKEDTO TABLE:



Account Id 1602111 Aadhar Card Number 18111

Insert

## UPDATE GUI PAGE FOR LINKEDTO TABLE:

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData								
<table><thead><tr><th>AccountID</th><th>AadharNumber</th></tr></thead><tbody><tr><td>1602066</td><td>1866</td></tr><tr><td>1602116</td><td>18116</td></tr><tr><td>1602120</td><td>118120</td></tr></tbody></table>									AccountID	AadharNumber	1602066	1866	1602116	18116	1602120	118120
AccountID	AadharNumber															
1602066	1866															
1602116	18116															
1602120	118120															


Enter New Account Id:

160266 Enter New AadharCard Number:

1866 Account Id: 1602066

AadharCard Number: 18066

## DELETE GUI PAGE FOR LINKEDTO TABLE:

—□×

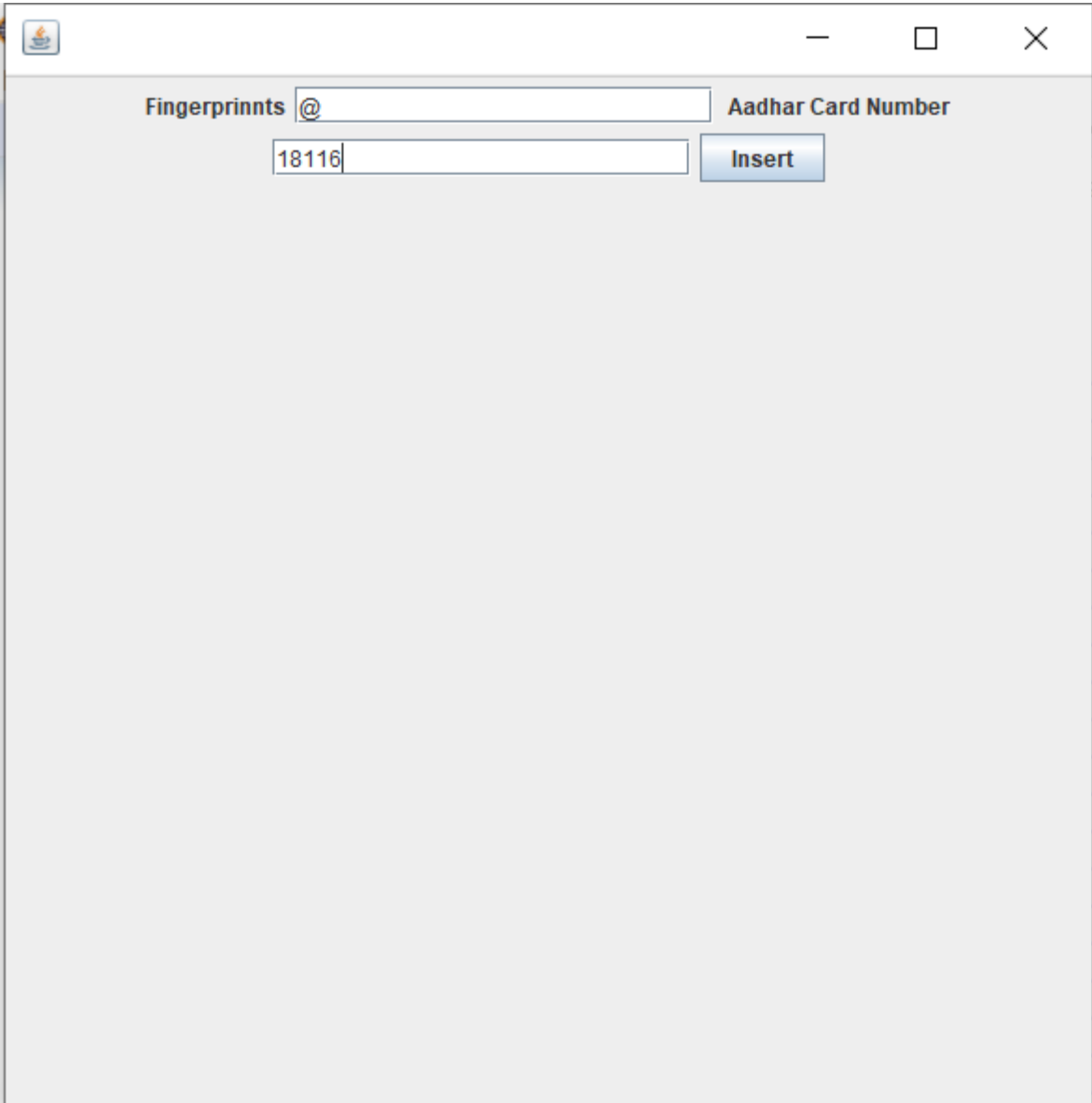
Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
AccountID	AadharNumber							
1602066	1866							
1602116	18116							
1602120	118120							

Account Id:

AadharCard Number:

Delete

## INSERT GUI PAGE FOR VERIFIEDTO TABLE:




The image shows a Java Swing window with a title bar containing a small icon, a minus sign, a maximize button, and a close button. The window's content area has a light gray background. At the top left, the text "Fingerprints @" is displayed. To its right is a text input field. Below this field is another text input field containing the number "18116". To the right of the "18116" field is a blue button with the text "Insert". The label "Aadhar Card Number" is positioned above the "Insert" button.

Fingerprints	Aadhar Card Number
@	18116

## UPDATE GUI PAGE FOR VERIFIEDTO TABLE:

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
AadharCardNumber    Fingerprint								
18115    >								
18116    !								
Enter New FringerPrint:								
@    Enter New AadharCard Number								
180116    Fingerprint: !								
AadharCard: 18116    Modify								

## DELETE GUI PAGE FOR VERIFIEDTO TABLE:

— □ ×

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
AadharNumber Fingerprint								
18115 >								
18116 !								

FringerPrint:

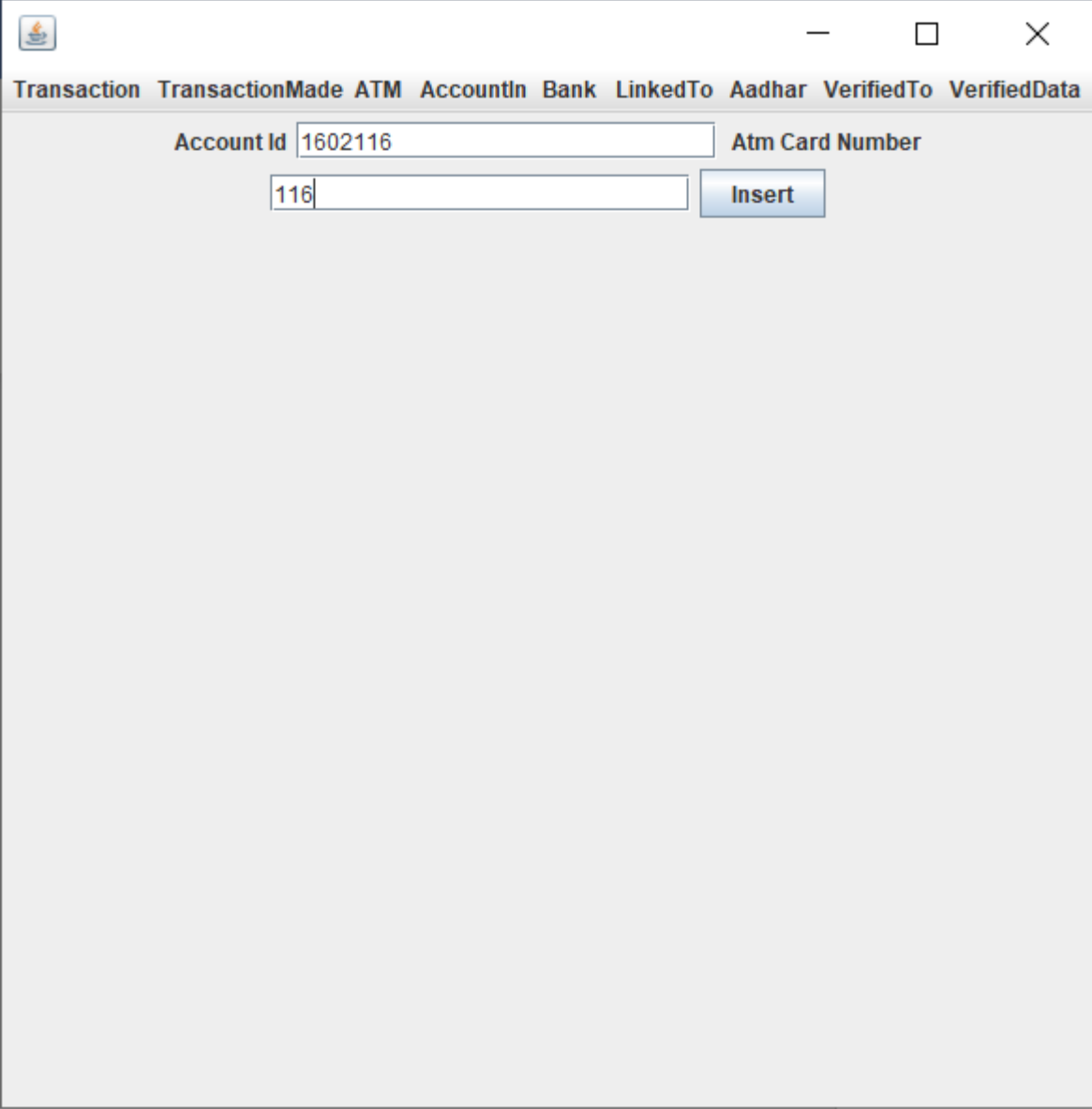
18116

AadharCard:

Delete



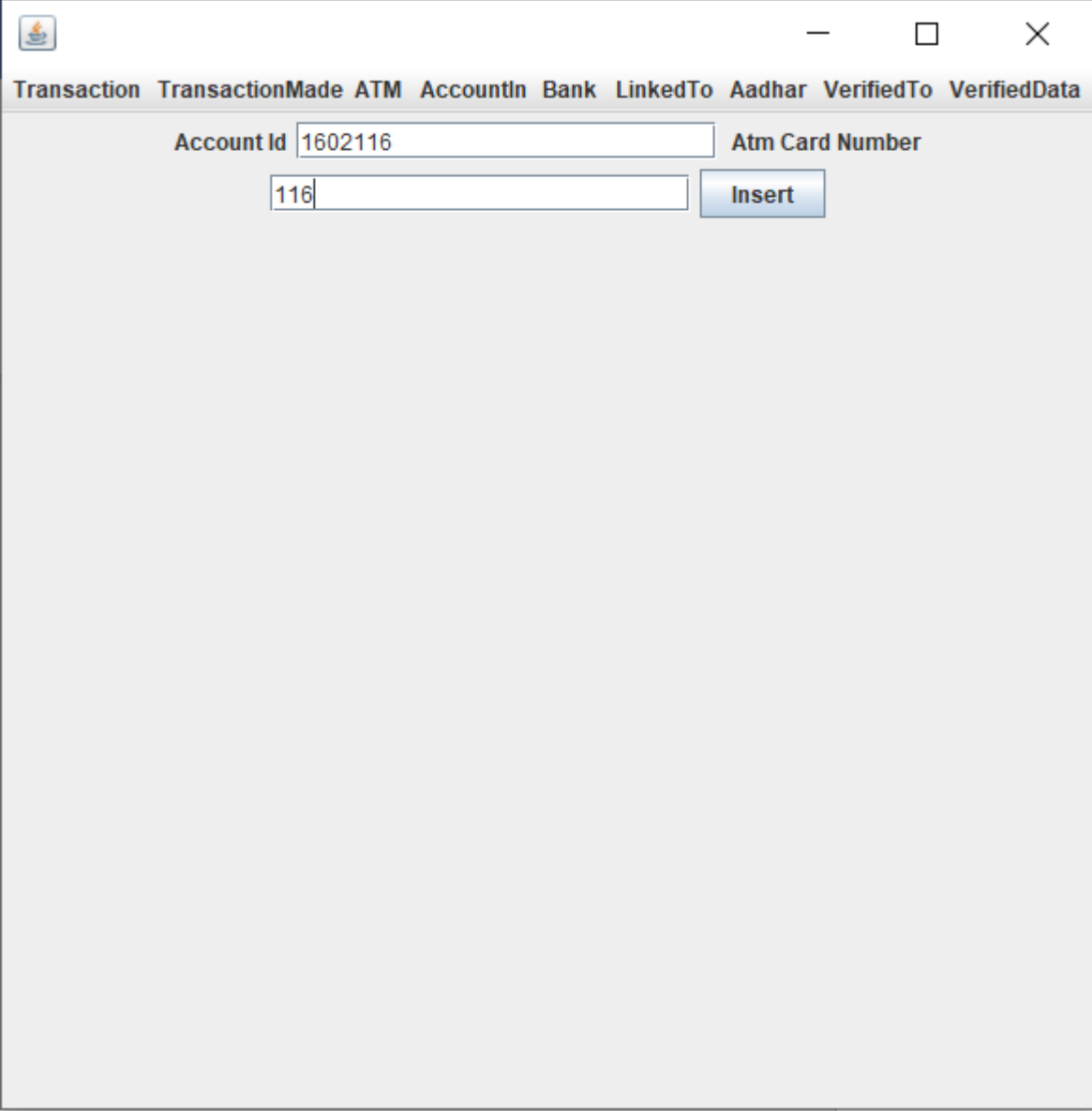
## INSERT GUI PAGE FOR ACCOUNTIN TABLE:



The image shows a Java Swing window titled "Transaction". The window has a standard title bar with a minimize button, a maximize button, and a close button. Below the title bar is a table header with the following columns: Transaction, TransactionMade, ATM, AccountIn, Bank, LinkedTo, Aadhar, VerifiedTo, and VerifiedData. The main area of the window contains two input fields. The first input field is labeled "Account Id" and contains the text "1602116". The second input field is labeled "Atm Card Number" and contains the text "116". To the right of the second input field is a button labeled "Insert".

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
<div>Account Id <input type="text" value="1602116"/></div> <div>Atm Card Number <input type="text" value="116"/></div> <div><input type="button" value="Insert"/></div>								

## INSERT GUI PAGE FOR ACCOUNTIN TABLE:



The image shows a Java Swing window titled "Transaction". The window has a standard title bar with a minimize button, a maximize button, and a close button. Below the title bar is a table header with the following columns: Transaction, TransactionMade, ATM, AccountIn, Bank, LinkedTo, Aadhar, VerifiedTo, and VerifiedData. The main area of the window is a light gray panel. It contains two input fields: "Account Id" with the value "1602116" and "Atm Card Number" with the value "116". There is an "Insert" button to the right of the "Atm Card Number" field.

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
<p>Account Id <input type="text" value="1602116"/> Atm Card Number <input type="text" value="116"/></p> <p><input type="button" value="Insert"/></p>								

UPDATE GUI PAGE FOR ACCOUNTIN TABLE:

—□×

TransactionTransactionMade ATM AccountIn Bank LinkedTo Aadhar VerifiedTo VerifiedData

AccountIDAtmCardNumber

160206666

1602116116

1602120120

Enter New AccountId:

160266

Enter New AtmCard Number

61

AccountId:


1602066

AtmCard Number:

66

Modify

**DELETE GUI PAGE FOR ACCOUNTIN TABLE:**



—

□

×


Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
AccountID	AtmCardNumber							
1602066	66							
1602116	116							
1602120	120							

Accountid:

AtmCard Number:

Delete

**INSERT GUI FOR TRANSACTIONMADE TABLE:**




—

□

×

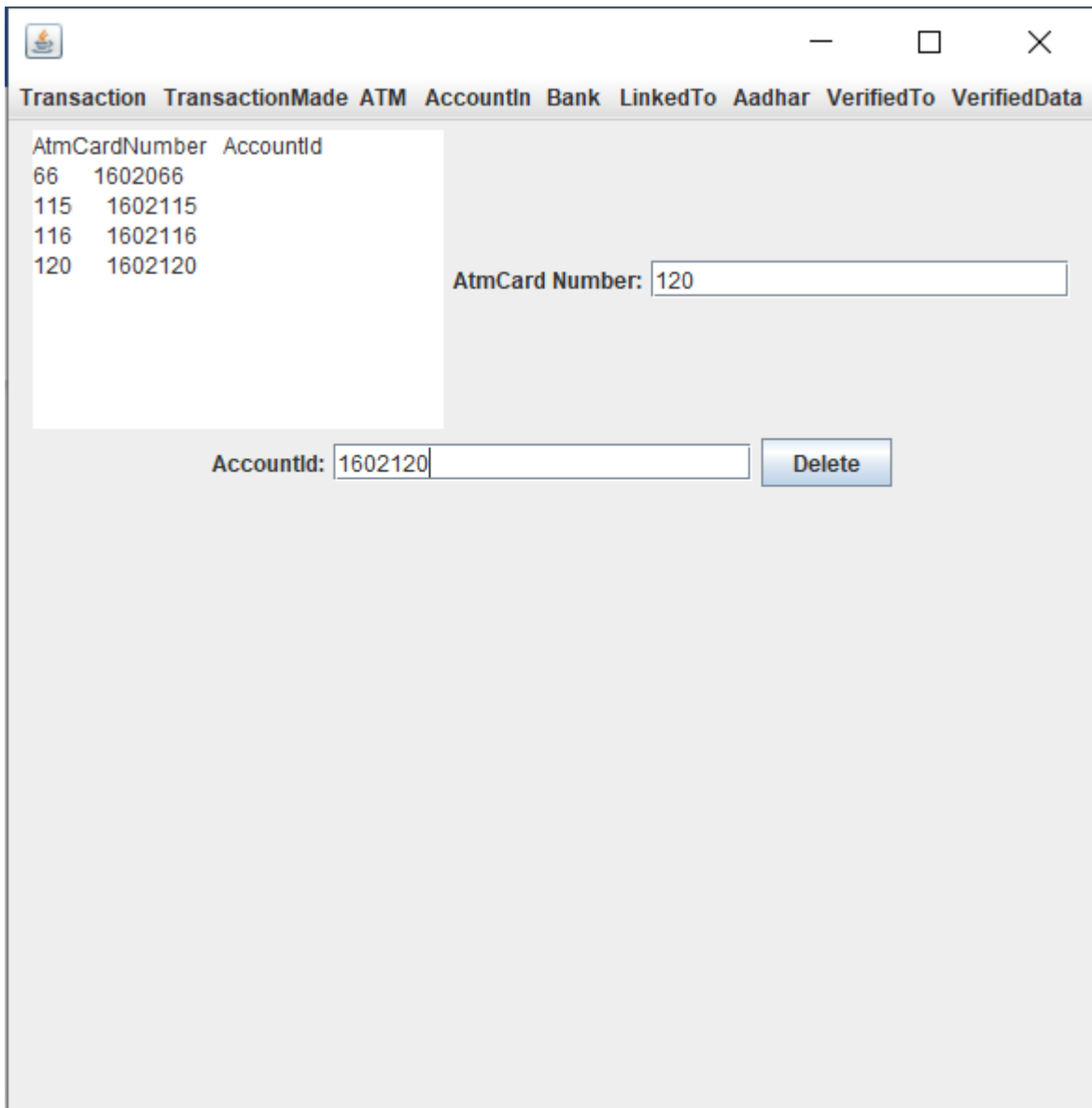
Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
Account Id	<input type="text" value="1602111"/>	Atm Card Number	<input type="text" value="111"/>	<input type="button" value="Insert"/>				

## UPDATE GUI PAGE FOR TRANSACTIONMADE:

— □ ×

Transaction	TransactionMade	ATM	AccountIn	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
<div><div>AtmCardNumber AccountId</div><div>66 1602066</div><div>115 1602115</div><div>116 1602116</div><div>120 1602120</div></div> <div>Enter New AtmCard Number:</div> <div><input type="text" value="066"/></div> <div>Enter New AccountId:</div> <div><input type="text" value="160266"/> AtmCard Number: <input type="text" value="66"/></div> <div>AccountId: <input type="text" value="1602066"/> <input type="button" value="Modify"/></div>								

## DELETE GUI PAGE FOR TRANSACTIONMADE TABLE:



Transaction	TransactionMade	ATM	AccountId	Bank	LinkedTo	Aadhar	VerifiedTo	VerifiedData
66	1602066							
115	1602115							
116	1602116							
120	1602120							

AtmCard Number:

AccountId:

## **DISCUSSION AND FUTURE WORK!**

This project tells us about we can access ATM by using our fingerprint by linking Aadhar card.

This make the more secured will using withdrawing or depositing money at ATM centers.

Future our government can plan these kinds of projects to be implemented.

## **REFERENCES:**

<https://www.decodejava.com/what-is-jdbc.htm>

<https://docs.oracle.com/javase/8/docs/api/>

<https://www.tutorialspoint.com/swing/index.htm>