

Archim Paul C. Pameroyan

DICT 2-5

Information Form

Output:

Applet Viewer: AppletJAppletMain.class

Applet

Information Form

NAME:

MIDDLE NAME:

LAST NAME:

BIRTHDATE:

GENDER: ☒ Male ☐ Female

OCCUPATION:

ADDRESS:

STREET:

UNIT:

DISTRICT:

CITY/PROVINCE:

☒ Show Stored Information

| | | | | | | |
|----------------|-----------|-----------|--------------------|--------|----------------------|-------------------------|
| Aldrin Jhan | Germidia | Celedonio | September 26, 1998 | Male | Network Manager | Blk 39, Rd. 53 We... |
| Archim Paul | Carpio | Pameroyan | February 16, 2000 | Male | Game Developer | 83 Umbel Roxas D... |
| Chin | Ruado | Ching | June 19, 1999 | Female | Game Designer | 136 Don Pedro Ma... |
| James Jimmuel | Tolentino | Sanbrano | August 31, 2000 | Male | System Administra... | 45-C Banahaw Cu... |
| Jhon Lawrence | San Luis | Manalad | September 6, 2000 | Male | Software Engineer | 21 D Fabella " " Di... |
| Lenard | Santiago | Selda | April 10, 1999 | Male | Network Manager | 666 Bagong Nayan... |
| Maria Venus | Cutamora | Jemina | November 7, 1999 | Female | IT Manager | 223 Quintina Sta. ... |
| Michaela Joy | Baltazar | Reyes | January 30, 1999 | Female | Web Developer | 11-A West Village ... |
| Nikki Anderson | Penos | Daluz | July 26, 2000 | Female | Web Developer | 287 Alfonso Isidro "... |
| Razelle Ann | Manuel | Apas | December 9, 1999 | Female | Computer Progra... | Lot 11 Blk, 9 Karan... |

JApplet Code:

```
package Applet;

import ConnectionDB.ConnectionDB;
import net.proteanit.sql.DbUtils;

import javax.swing.JApplet;

import java.awt.Color;
import javax.swing.JLabel;
import javax.swing.JOptionPane;

import java.awt.Font;
import javax.swing.SwingConstants;

import java.awt.Dimension;
import javax.swing.JTextField;
import javax.swing.JComboBox;
import javax.swing.ButtonGroup;
import javax.swing.JTextArea;
import javax.swing.JRadioButton;
import javax.swing.JScrollPane;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
```

```
import java.awt.event.ItemListener;
```

```
import javax.swing.JButton;
```

```
import java.sql.Connection;
```

```
import java.sql.PreparedStatement;
```

```
import java.sql.ResultSet;
```

```
import java.sql.ResultSetMetaData;
```

```
import java.sql.SQLException;
```

```
import java.sql.Statement;
```

```
import javax.swing.JTable;
```

```
import javax.swing.JCheckBox;
```

```
import javax.swing.table.DefaultTableModel;
```

```
import javax.swing.ScrollPaneConstants;
```

```
import javax.swing.JList;
```

```
import javax.swing.ListSelectionModel;
```

```
import java.awt.event.KeyAdapter;
```

```
import java.awt.event.KeyEvent;
```

```
public class JAppletMain extends JApplet implements ActionListener, ItemListener {
```

```
    public JAppletMain() throws ClassNotFoundException, InstantiationException,  
    SQLException, IllegalAccessException {
```

```
        conn = ConnectionDB.getConnection();
```

```

    }

//Declaration Variable//

    JTextField TFName, TFMiddlename, TFLastname, TFStreet, TFUnit, TFDistrict, TFCity;

    JButton BClear, BAdd;

    JTextArea TAAddress;

    JLabel LInfo, LName, LMiddlename, LLastname, LStreet, LCity, LUnit, LDistrict, LAddress,
    LGender, LBirthdate,

        LOccupation;

    JRadioButton RBMale, RBFemale;

    JComboBox<String> CBYear, CBMonth, CBDate;

    ButtonGroup Gender;

    JScrollPane SPane1, SPane2, SPane3;

    JList<String> List;

    JTable TInformation;

    JCheckBox CBox;

    public ResultSetMetaData rsmd = null;

    public Connection conn = null;

    public Statement st = null;

    public PreparedStatement ps = null;

    public ResultSet rs = null;

// Applet Initialization//

    public void init() {

        this.setSize(new Dimension(1500, 1000));

        this.repaint();

        getContentPane().setBackground(new Color(224, 255, 255));

```

```
getContentPane().setLayout(null);
```

```
// Label//
```

```
LInfo = new JLabel("Information Form");
```

```
LInfo.setHorizontalAlignment(SwingConstants.CENTER);
```

```
LInfo.setVerticalAlignment(SwingConstants.TOP);
```

```
LInfo.setFont(new Font("Times New Roman", Font.BOLD, 35));
```

```
LInfo.setBounds(32, 12, 385, 40);
```

```
getContentPane().add(LInfo);
```

```
LName = new JLabel("Name:");
```

```
LName.setBackground(Color.WHITE);
```

```
LName.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));
```

```
LName.setHorizontalAlignment(SwingConstants.LEFT);
```

```
LName.setBounds(10, 68, 146, 31);
```

```
getContentPane().add(LName);
```

```
LMiddlename = new JLabel("Middle Name:");
```

```
LMiddlename.setHorizontalAlignment(SwingConstants.LEFT);
```

```
LMiddlename.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));
```

```
LMiddlename.setBackground(Color.WHITE);
```

```
LMiddlename.setBounds(10, 109, 150, 31);
```

```
getContentPane().add(LMiddlename);
```

```
LLastname = new JLabel("Last Name:");
```

```
LLastname.setHorizontalAlignment(SwingConstants.LEFT);
```

```
LLastname.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));
```

```
LLastname.setBackground(Color.WHITE);  
LLastname.setBounds(10, 150, 150, 31);  
getContentPane().add(LLastname);
```

```
LStreet = new JLabel("Street:");  
LStreet.setHorizontalAlignment(SwingConstants.RIGHT);  
LStreet.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 15));  
LStreet.setBackground(Color.WHITE);  
LStreet.setBounds(10, 390, 146, 31);  
getContentPane().add(LStreet);
```

```
LUnit = new JLabel("Unit:");  
LUnit.setHorizontalAlignment(SwingConstants.RIGHT);  
LUnit.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 15));  
LUnit.setBackground(Color.WHITE);  
LUnit.setBounds(10, 433, 146, 31);  
getContentPane().add(LUnit);
```

```
LDistrict = new JLabel("District:");  
LDistrict.setHorizontalAlignment(SwingConstants.RIGHT);  
LDistrict.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 15));  
LDistrict.setBackground(Color.WHITE);  
LDistrict.setBounds(10, 476, 146, 31);  
getContentPane().add(LDistrict);
```

```
LCity = new JLabel("City/Province:");  
LCity.setHorizontalAlignment(SwingConstants.RIGHT);
```

```
LCity.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 15));  
LCity.setBackground(Color.WHITE);  
LCity.setBounds(10, 519, 146, 31);  
getContentPane().add(LCity);
```

```
LGender = new JLabel("Gender:");  
LGender.setHorizontalAlignment(SwingConstants.LEFT);  
LGender.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));  
LGender.setBackground(Color.WHITE);  
LGender.setBounds(6, 248, 150, 31);  
getContentPane().add(LGender);
```

```
LBirthdate = new JLabel("Birthdate:");  
LBirthdate.setHorizontalAlignment(SwingConstants.LEFT);  
LBirthdate.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));  
LBirthdate.setBackground(Color.WHITE);  
LBirthdate.setBounds(6, 205, 150, 31);  
getContentPane().add(LBirthdate);
```

```
LOccupation = new JLabel("Occupation:");  
LOccupation.setHorizontalAlignment(SwingConstants.LEFT);  
LOccupation.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));  
LOccupation.setBackground(Color.WHITE);  
LOccupation.setBounds(6, 304, 150, 31);  
getContentPane().add(LOccupation);
```

```
LAddress = new JLabel("Address:");
```

```

LAddress.setHorizontalAlignment(SwingConstants.LEFT);

LAddress.setFont(new Font("Copperplate Gothic Bold", Font.PLAIN, 20));

LAddress.setBackground(Color.WHITE);

LAddress.setBounds(10, 347, 150, 31);

getContentPane().add(LAddress);


// Text Field//

TFName = new JTextField();

TFName.addKeyListener(new KeyAdapter() {

    public void keyPressed(KeyEvent e) {

        char c = e.getKeyChar();

        if (Character.isLetter(c) || Character.isWhitespace(c) ||
Character.isISOControl(c)) {

            TFName.setEditable(true);

        } else {

            TFName.setEditable(false);

        }

    }

});

TFName.setBackground(new Color(102, 205, 170));

TFName.setFont(new Font("Cooper Black", Font.PLAIN, 15));

TFName.setBounds(166, 68, 254, 31);

TFName.setColumns(10);

getContentPane().add(TFName);


TFMiddlename = new JTextField();

TFMiddlename.addKeyListener(new KeyAdapter() {

```



```

        public void keyPressed(KeyEvent e) {
            char c = e.getKeyChar();
            if (Character.isLetter(c) || Character.isWhitespace(c) ||
Character.isISOControl(c)) {
                TFMiddlename.setEditable(true);
            } else {
                TFMiddlename.setEditable(false);
            }
        }
    });
    TFMiddlename.setFont(new Font("Cooper Black", Font.PLAIN, 15));
    TFMiddlename.setColumns(10);
    TFMiddlename.setBackground(new Color(102, 205, 170));
    TFMiddlename.setBounds(166, 109, 254, 31);
    getContentPane().add(TFMiddlename);

    TFLastname = new JTextField();
    TFLastname.addKeyListener(new KeyAdapter() {
        public void keyPressed(KeyEvent e) {
            char c = e.getKeyChar();
            if (Character.isLetter(c) || Character.isWhitespace(c) ||
Character.isISOControl(c)) {
                TFLastname.setEditable(true);
            } else {
                TFLastname.setEditable(false);
            }
        }
    });

```

```

TFLastname.setFont(new Font("Cooper Black", Font.PLAIN, 15));
TFLastname.setColumns(10);
TFLastname.setBackground(new Color(102, 205, 170));
TFLastname.setBounds(166, 150, 254, 31);
getContentPane().add(TFLastname);

TFStreet = new JTextField();
TFStreet.addKeyListener(new KeyAdapter() {
    public void keyPressed(KeyEvent e) {
        char c = e.getKeyChar();
        if (Character.isLetter(c) || Character.isWhitespace(c) ||
Character.isISOControl(c)) {
            TFStreet.setEditable(true);
        } else {
            TFStreet.setEditable(false);
        }
    }
});
TFStreet.setFont(new Font("Cooper Black", Font.PLAIN, 15));
TFStreet.setColumns(10);
TFStreet.setBackground(new Color(102, 205, 170));
TFStreet.setBounds(166, 390, 254, 31);
getContentPane().add(TFStreet);

TFUnit = new JTextField();
TFUnit.setFont(new Font("Cooper Black", Font.PLAIN, 15));
TFUnit.setColumns(10);

```

```
TFUnit.setBackground(new Color(102, 205, 170));
```

```
TFUnit.setBounds(166, 433, 254, 31);
```

```
getContentPane().add(TFUnit);
```

```
TFDistrict = new JTextField();
```

```
TFDistrict.setFont(new Font("Cooper Black", Font.PLAIN, 15));
```

```
TFDistrict.setColumns(10);
```

```
TFDistrict.setBackground(new Color(102, 205, 170));
```

```
TFDistrict.setBounds(166, 476, 254, 31);
```

```
getContentPane().add(TFDistrict);
```

```
TFCity = new JTextField();
```

```
TFCity.addKeyListener(new KeyAdapter() {
```

```
    public void keyPressed(KeyEvent e) {
```

```
        char c = e.getKeyChar();
```

```
        if (Character.isLetter(c) || Character.isWhitespace(c) ||  
Character.isISOControl(c)) {
```

```
            TFCity.setEditable(true);
```

```
        } else {
```

```
            TFCity.setEditable(false);
```

```
        }
```

```
    }
```

```
});
```

```
TFCity.setFont(new Font("Cooper Black", Font.PLAIN, 15));
```

```
TFCity.setColumns(10);
```

```
TFCity.setBackground(new Color(102, 205, 170));
```

```
TFCity.setBounds(166, 519, 254, 31);
```

```
getContentPane().add(TFCity);
```

```
// Combo Box//
```

```
String month[] = { " ", "January", "February", "March", "April", "May", "June",  
"July", "August", "September",
```

```
    "October", "November", "December" };
```

```
CBMonth = new JComboBox(month);
```

```
CBMonth.setForeground(Color.BLACK);
```

```
CBMonth.setBackground(new Color(102, 205, 170));
```

```
CBMonth.setFont(new Font("Copperplate Gothic Light", Font.PLAIN, 20));
```

```
CBMonth.setBounds(142, 213, 146, 21);
```

```
CBMonth.setActionCommand((String) CBMonth.getSelectedItem());
```

```
getContentPane().add(CBMonth);
```

```
String date[] = { " ", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12",  
"13", "14", "15", "16",
```

```
    "17", "18", "19", "20", "21", "22", "23", "24", "25", "26", "27",  
"28", "29", "30", "31" };
```

```
CBDate = new JComboBox(date);
```

```
CBDate.setBackground(new Color(102, 205, 170));
```

```
CBDate.setName("Date");
```

```
CBDate.setFont(new Font("Copperplate Gothic Light", Font.PLAIN, 20));
```

```
CBDate.setBounds(289, 213, 66, 21);
```

```
CBDate.setActionCommand((String) CBDate.getSelectedItem());
```

```
getContentPane().add(CBDate);
```

```

        String year[] = { " ", "2020", "2019", "2018", "2017", "2016", "2015", "2014",
"2013", "2012", "2011", "2010",
        "2009", "2008", "2007", "2006", "2005", "2004", "2003", "2002",
"2001", "2000", "1999", "1998", "1997",
        "1996", "1995", "1994", "1993", "1992", "1991", "1990", "1989",
"1988", "1987", "1986", "1985", "1984",
        "1983", "1982", "1981", "1980", "1979", "1978", "1977", "1976",
"1975", "1974", "1973", "1972", "1971",
        "1970", "1969", "1968", "1967", "1966", "1965", "1964", "1963",
"1962", "1961", "1960", "1959", "1958",
        "1957", "1956", "1955", "1954", "1953", "1952", "1951", "1950",
"1949", "1948", "1947", "1946", "1945",
        "1944", "1943", "1942", "1941", "1940", "1939", "1938", "1937",
"1936", "1935", "1934", "1933", "1932",
        "1931", "1930", "1929", "1928", "1927", "1926", "1925", "1924",
"1923", "1922", "1921", "1920" };

```

```

        CBYear = new JComboBox(year);

```

```

        CBYear.setBackground(new Color(102, 205, 170));

```

```

        CBYear.setName("Year");

```

```

        CBYear.setFont(new Font("Copperplate Gothic Light", Font.PLAIN, 20));

```

```

        CBYear.setBounds(356, 213, 122, 21);

```

```

        CBDate.setActionCommand((String) CBDate.getSelectedItem());

```

```

        getContentPane().add(CBYear);

```

```

        // Text Area//

```

```

        SPane2 = new JScrollPane();

```

```

        SPane2.setHorizontalScrollBarPolicy(ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AL
WAYS);

```

```

        SPane2.setBounds(46, 562, 410, 194);

```

```
getContentPane().add(SPane2);
TAAddress = new JTextArea();
SPane2.setViewportView(TAAddress);
TAAddress.setEditable(false);
TAAddress.setWrapStyleWord(false);
TAAddress.setFont(new Font("Cooper Black", Font.PLAIN, 20));
TAAddress.setLineWrap(false);
TAAddress.setBackground(new Color(102, 205, 170));
```

```
// Radio Button//
```

```
RBMale = new JRadioButton("Male");
RBMale.setBackground(new Color(102, 205, 170));
RBMale.setFont(new Font("Dialog", Font.PLAIN, 20));
RBMale.setBounds(164, 252, 91, 24);
RBMale.setActionCommand("Male");
getContentPane().add(RBMale);
```

```
RBFemale = new JRadioButton("Female");
RBFemale.setBackground(new Color(102, 205, 170));
RBFemale.setFont(new Font("Dialog", Font.PLAIN, 20));
RBFemale.setBounds(259, 250, 110, 24);
RBFemale.setActionCommand("Female");
getContentPane().add(RBFemale);
```

```
Gender = new ButtonGroup();
Gender.add(RBMale);
Gender.add(RBFemale);
```

```

// Button//
BAdd = new JButton("Add");
BAdd.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        if (TFName.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFMiddlename.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFLastname.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFUnit.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFStreet.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFCity.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (TFDistrict.getText().trim().length() == 0) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (List.getSelectedValue() == null) {
            JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
        } else if (Gender.isSelected(null)) {

```

```

JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
    } else if (CBMonth.getSelectedItem() == " ") {
        JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
    } else if (CBDate.getSelectedItem() == " ") {
        JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
    } else if (CByear.getSelectedItem() == " ") {
        JOptionPane.showMessageDialog(null, "Please Fill Up The
Form First!");
    }

    else {
        String space = " ";
        String comma = ",";
        TAAAddress.setText(TFUnit.getText() + space +
TFStreet.getText() + space + TFDistrict.getText()
+ space + "District" + comma + space +
TFCity.getText());

    }

    try {
        String query = "insert into information
(FName,MName,LName,Birthdate,Gender,Occupation,Address) values (?, ?, ?, ?, ?, ?) ";
        ps = conn.prepareStatement(query);
        ps.setString(1, TFName.getText());
        ps.setString(2, TFMiddlename.getText());

```



```

        ps.setString(3, TFLastname.getText());

        ps.setString(4, (String) CBMonth.getSelectedItem() + " " +
(String) CBDate.getSelectedItem() + ", "

                                + " " + (String) CBYear.getSelectedItem());

        ps.setString(5,
Gender.getSelection().getActionCommand());

        ps.setString(6, (String) List.getSelectedValue());

        ps.setString(7, TAAddress.getText());

        ps.execute();

        JOptionPane.showMessageDialog(null, "Successfully
Added");

        ps.close();

    } catch (Exception ex) {

        ex.printStackTrace();

    }

}

});

BAdd.setFont(new Font("Dialog", Font.BOLD, 20));

BAdd.setBounds(10, 768, 122, 45);

getContentPane().add(BAdd);

BClear = new JButton("Clear");

BClear.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {

```

```

        TFName.setText("");
        TFMiddlename.setText("");
        TFLastname.setText("");
        TFUnit.setText("");
        TFCity.setText("");
        TFStreet.setText("");
        TFDistrict.setText("");
        TAAddress.setText("");
        Gender.clearSelection();
        CBMonth.setSelectedItem(null);
        CBDate.setSelectedItem(null);
        CBYear.setSelectedItem(null);
        List.clearSelection();
    }

});

BClear.setFont(new Font("Dialog", Font.BOLD, 20));
BClear.setBounds(368, 768, 122, 45);
getContentPane().add(BClear);

// Table//
DefaultTableModel TModel = new DefaultTableModel();
TModel.addColumn("First Name");
TModel.addColumn("Middle Name");
TModel.addColumn("Last Name");
TModel.addColumn("Birthdate");
TModel.addColumn("Gender");

```

```

TModel.addColumn("Occupation");
TModel.addColumn("Address");
TInformation = new JTable(TModel);
TInformation.setRowHeight(60);
TInformation.setRowSelectionAllowed(false);
TInformation.setFocusable(false);
TInformation.setEditingRow(0);
TInformation.setEditingColumn(0);
TInformation.setAutoCreateRowSorter(true);
TInformation.setAutoCreateColumnsFromModel(false);
TInformation.setFont(new Font("Dialog",Font.PLAIN,15));
SPane1 = new JScrollPane();
SPane1.setBackground(new Color(135, 206, 235));
SPane1.setFont(new Font("Copperplate Gothic Light", Font.PLAIN, 10));

SPane1.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);

SPane1.setBounds(490, 68, 998, 650);
getContentPane().add(SPane1);
SPane1.setViewportView(TInformation);

// Checkbox//
CBox = new JCheckBox("Show Stored Information");
SPane1.setColumnHeaderView(CBox);
CBox.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {
        if (CBox.isSelected()) {

```

```

        ShowData();
    } else {
        TInformation.setModel(TModel);
    }
}

}

);

CBox.setBackground(new Color(102, 205, 170));
CBox.setFont(new Font("DialogInput", Font.BOLD, 20));

// List Box//
String list[] = { "Web Developer", "IT Manager", "Software Engineer", "Database
Administrator",
                "Game Developer", "Computer Programmer", "Network
Engineer", "Game Designer", "System Administrator",
                "Network Manager" };

SPane3 = new JScrollPane();

SPane3.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_ALWAYS);
SPane3.setBounds(166, 284, 250, 98);
getContentPane().add(SPane3);
List = new JList(list);
List.setFont(new Font("Dialog", Font.BOLD, 15));
List.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);

```

```

        List.getSelectedValue();

        SPane3.setViewportView(List);

        List.setBackground(new Color(102, 205, 170));

    }

//Displays Data from Database//
public void ShowData() {

    DefaultTableModel TModel = new DefaultTableModel();

    TModel.addColumn("First Name");
    TModel.addColumn("Middle Name");
    TModel.addColumn("Last Name");
    TModel.addColumn("Birthdate");
    TModel.addColumn("Gender");
    TModel.addColumn("Occupation");
    TModel.addColumn("Address");

    try {

        Class.forName("com.mysql.cj.jdbc.Driver").newInstance();
        Statement st = conn.createStatement();
        st.executeQuery("select * from informationdb.information");
        ResultSet rs = st.getResultSet();

        TInformation.setModel(DbUtils.resultSetToTableModel(rs));
    }
}

```

```
        } catch (Exception e) {  
            JOptionPane.showMessageDialog(null, "Error: Can't Display Data");  
        }  
    }  
  
    @Override  
    public void itemStateChanged(ItemEvent arg0) {  
        // TODO Auto-generated method stub  
  
    }  
  
    @Override  
    public void actionPerformed(ActionEvent arg0) {  
        // TODO Auto-generated method stub  
  
    }  
}
```

Database Connection Code:

```
package ConnectionDB;

import java.sql.*;

public class ConnectionDB {

    static String driver = "com.mysql.cj.jdbc.Driver";

    static String url ="jdbc:mysql://localhost:3306/informationdb";

    static String username = "root";

    static String password = "kakashijirachisohaer";

    public static Connection getConnection(){

        Connection conn=null;

        try{

            Class.forName(driver);

            conn=DriverManager.getConnection(url,username,password);

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

        return conn;

    }

}
```