**TECHNOLOGICAL INSTITUTE OF THE PHILIPPINES**

**QUEZON CITY**

**COLLEGE OF INFORMATION TECHNOLOGY EDUCATION (CITE)**

**CS 201-IT21S1 - Data Structures and Algorithms**

|  |  |
| --- | --- |
| **Name:** Kathlynne Joy Moraga | **Date: 09/29/2021** |
| **Program/Section: IT21S1** | **Instructor: Ms. Rosmina Joy M. Cabauatan** |
| **Assessment Task: Assignment 4. | Final Project Development** | |

**Code**

# **EmployeePos class (LinkedList)**

**package** groupThree\_FinalProject;

**class** EmployeePos {

**public** Node head = **null**;

**class** Node {

**private** String name;

**private** String positionName;

**private** Node next;

**public** Node(String name, String positionName) {

**this**.name = name;

**this**.positionName = positionName;

**this**.next = **null**;

}

}

**public** **int** prepend(String name, String positionName) {

Node newNode = **new** Node(name, positionName);

**if** (**this**.head == **null**) {

**this**.head = newNode;

} **else** {

newNode.next = **this**.head;

**this**.head = newNode;

}

**return** 0;

}

**public** **int** append(String name, String positionName) {

Node newNode = **new** Node(name, positionName);

**int** index = 0;

**if** (**this**.head == **null**) {

**this**.head = newNode;

} **else** {

Node current = **this**.head;

**while** (current.next != **null**) {

current = current.next;

index++;

}

index++;

current.next = newNode;

}

**return** index;

}

**public** **int** add(**int** position, String name, String positionName) {

Node newNode = **new** Node(name, positionName);

Node current = **this**.head, prev = **this**.head;

**if** (position == 0) {

// Point the new node's next to head

newNode.next = head;

// Make the new node as head

**this**.head = newNode;

**return** 0;

}

**int** index = 0;

**while** (--position > 0) {

prev = current;

current = current.next;

index++;

}

prev.next = newNode;

newNode.next = current;

**return** index;

}

**public** String display() {

String value = "";

**if** (**this**.head == **null**) {

System.***out***.println("The List is empty.");

} **else** {

Node current = **this**.head;

**while** (current != **null**) {

value += current.name + "-" + current.positionName + ", ";

current = current.next;

System.***out***.println("Current " + value);

}

}

**return** value;

}

}

# **EmployeeList class (Runner)**

package groupThree\_FinalProject;

import java.awt.EventQueue;

import java.awt.Image;

import java.awt.Toolkit;

import javax.swing.DefaultComboBoxModel;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.SwingConstants;

import javax.swing.border.EmptyBorder;

import javax.swing.table.DefaultTableModel;

import java.awt.Font;

import javax.swing.JTextField;

import javax.swing.JTable;

import javax.swing.JButton;

import javax.swing.JComboBox;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import javax.swing.JScrollPane;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

public class EmployeeList extends JFrame {

private JPanel contentPane;

private JTextField nameTextField;

private JTable table;

DefaultTableModel model;

private int managerCount=0;

private int tellerCount=0;

private EmployeePos list = new EmployeePos();

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

EmployeeList frame = new EmployeeList();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public EmployeeList() {

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setIconImage(Toolkit.getDefaultToolkit().getImage(Payroll\_JFrame.class.getResource("/res/3.png")));

setTitle("Employee Attendance");

setBounds(450, 190, 1014, 597);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(null);

setResizable(false);

JLabel logo = new JLabel("");

logo.setHorizontalAlignment(SwingConstants.CENTER);

logo.setBounds(289, 22, 457, 92);

logo.setIcon(new ImageIcon(new javax.swing.ImageIcon(getClass().getResource("/res/Horizontal Logo.png")).getImage().getScaledInstance(400, 400, Image.SCALE\_SMOOTH)));

contentPane.add(logo);

JLabel lblName = new JLabel("Name:");

lblName.setHorizontalAlignment(SwingConstants.RIGHT);

lblName.setFont(new Font("Tahoma", Font.PLAIN, 16));

lblName.setBounds(51, 126, 84, 17);

contentPane.add(lblName);

JLabel lblPosition = new JLabel("Position:");

lblPosition.setHorizontalAlignment(SwingConstants.RIGHT);

lblPosition.setFont(new Font("Tahoma", Font.PLAIN, 16));

lblPosition.setBounds(51, 175, 84, 17);

contentPane.add(lblPosition);

nameTextField = new JTextField();

nameTextField.setBounds(145, 121, 189, 31);

contentPane.add(nameTextField);

nameTextField.setColumns(10);

JComboBox<String> comboBox\_1 = new JComboBox<String>();

comboBox\_1.setModel(new DefaultComboBoxModel<String>(new String[] { "Manager", "Teller", "Technician", "Instructor"}));

comboBox\_1.setBounds(145, 175, 189, 29);

contentPane.add(comboBox\_1);

JScrollPane scrollPane = new JScrollPane();

scrollPane.addMouseListener(new MouseAdapter() {

@Override

public void mouseClicked(MouseEvent e) {

int i = table.getSelectedRow();

if(i>=0) {

comboBox\_1.setSelectedItem(model.getValueAt(i,0).toString());

nameTextField.setText(model.getValueAt(i,1).toString());

}else {

JFrame errorFrame = new JFrame();

Toolkit.getDefaultToolkit().beep();

JOptionPane.showMessageDialog(errorFrame, "Please Select a Row First","3MBC Payroll System", JOptionPane.ERROR\_MESSAGE);

}

}

});

scrollPane.setBounds(22, 226, 953, 324);

contentPane.add(scrollPane);

table = new JTable();

model = new DefaultTableModel();

Object[] column = {"POSITION", "NAME"};

final Object[] row = new Object[2];

model.setColumnIdentifiers(column);

table.setModel(model);

scrollPane.setViewportView(table);

JButton btnDelete = new JButton("DELETE");

btnDelete.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

int i = table.getSelectedRow();

if(i>=0) {

int n = JOptionPane.showConfirmDialog(

null,

"Are you sure you want to permanently remove this item?",

"3MBC Payroll System",

JOptionPane.YES\_NO\_OPTION);

if(n==JOptionPane.YES\_OPTION) {

model.removeRow(i);

ImageIcon logo = new ImageIcon(getClass().getResource("/res/3MBC Payroll System LOGO.png"));

Image img3MBC = logo.getImage();

Image modifiedLogo = img3MBC.getScaledInstance(100, 100, java.awt.Image.SCALE\_SMOOTH);

logo = new ImageIcon(modifiedLogo);

JOptionPane.showMessageDialog(null, "Deleted Successfully","3MBC Payroll System", JOptionPane.INFORMATION\_MESSAGE, logo);

}

}else {

JFrame errorFrame = new JFrame();

Toolkit.getDefaultToolkit().beep();

JOptionPane.showMessageDialog(errorFrame, "Please Select a Row First","3MBC Payroll System", JOptionPane.ERROR\_MESSAGE);

}

}

});

btnDelete.setBounds(855, 152, 85, 21);

contentPane.add(btnDelete);

JButton btnPrint = new JButton("DONE");

btnPrint.addMouseListener(new MouseAdapter() {

@Override

public void mouseClicked(MouseEvent e) {

//Will proceed to Payroll Page

Payroll\_JFrame payroll = new Payroll\_JFrame();

payroll.setVisible(true);

EmployeeList.this.dispose();

}

});

btnPrint.setBounds(855, 183, 85, 21);

contentPane.add(btnPrint);

JButton btnAdd = new JButton("ADD");

btnAdd.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if(nameTextField.getText().equals("")) {

JFrame errorFrame = new JFrame();

Toolkit.getDefaultToolkit().beep();

JOptionPane.showMessageDialog(errorFrame, "Please Fill Complete Information","3MBC Payroll System", JOptionPane.ERROR\_MESSAGE);

} else {

String position = comboBox\_1.getSelectedItem().toString();

String name = nameTextField.getText();

int rowCount = model.getRowCount();

model.setRowCount(rowCount);

int index = 0;

if (position.equals("Manager")) {

index = list.prepend(name, position);

++managerCount;

} else if (position.equals("Teller")) {

if(managerCount==0) {

index = list.prepend(name, position);

} else {

index = list.add(managerCount + 1, name, position);

}

++tellerCount;

} else if(position.equals("Technician")){

if(managerCount==0) {

index = list.prepend(name, position);

} else if(tellerCount ==0) {

index = list.add(managerCount +1, name, position);

} else {

index = list.add(managerCount + tellerCount + 1, name, position);

}

} else if(position.equals("Instructor")) {

index = list.append(name, position);

}

row[0] = position;

row[1] = name;

model.insertRow(index, row);

nameTextField.setText("");

}

}

});

btnAdd.setBounds(855, 121, 85, 21);

contentPane.add(btnAdd);

}

}

# **Queue class (Queue)**

**package** groupThree\_FinalProject;

**public** **class** Queue {

**private** Node front, rear;

**private** **int** currentSize; // number of items

//class to define linked node

**private** **class** Node

{

Object[] names;

Node next;

}

//Zero argument constructor

**public** Queue()

{

front = **null**;

rear = **null**;

currentSize = 0;

}

**public** **boolean** isEmpty()

{

**return** (currentSize == 0);

}

//Remove item from the beginning of the list.

**public** Object[] dequeue()

{

Object[] data = front.names;

front = front.next;

**if** (isEmpty())

{

rear = **null**;

}

currentSize--;

**return** data;

}

//Add data to the end of the list.

**public** **void** enqueue(Object[] data)

{

Node oldRear = rear;

rear = **new** Node();

rear.names = data;

rear.next = **null**;

**if** (isEmpty())

{

front = rear;

}

**else**

{

oldRear.next = rear;

}

currentSize++;

}

}

# **Payroll\_JFrame class (Runner)**

**package** groupThree\_FinalProject;

**import** java.awt.EventQueue;

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**import** javax.swing.JScrollPane;

**import** javax.swing.JTable;

**import** javax.swing.border.EmptyBorder;

**import** javax.swing.table.DefaultTableModel;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** java.awt.Font;

**import** java.awt.Image;

**import** javax.swing.SwingConstants;

**import** javax.swing.JTextField;

**import** javax.swing.ImageIcon;

**import** javax.swing.JButton;

**import** javax.swing.JComboBox;

**import** java.awt.event.ActionListener;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**import** java.util.Stack;

**import** java.awt.Toolkit;

**public** **class** Payroll\_JFrame **extends** JFrame {

**private** JPanel contentPane;

**private** JTextField textFieldID;

**private** JTextField textFieldPhilHealth;

**private** JTextField textFieldTax;

**private** JTextField textFieldGross;

**private** JTextField textFieldName;

**private** JTextField textFieldTotalDeduction;

**private** JTextField textFieldNet;

**private** JTextField textFieldSSS;

**private** JTextField textFieldOTP;

**private** JTextField textFieldPagIbig;

**private** JTextField textFieldWorkingHours;

DefaultTableModel model;

**private** JButton btnAdd;

**private** JButton btnDelete;

**private** JButton btnClear;

**private** JButton btnUpdate;

**private** JTable tablePayRoll;

**private** JScrollPane scrollPane;

**private** JButton btnOK;

**private** JTextField textFieldBasicPay;

**private** JLabel lblPagIbig;

Queue queue = **new** Queue();

/\*\*

\* Launch the application.

\*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

Payroll\_JFrame frame = **new** Payroll\_JFrame();

frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

**public** Payroll\_JFrame() {

setResizable(**false**);

setIconImage(Toolkit.*getDefaultToolkit*().getImage(Payroll\_JFrame.**class**.getResource("/res/3.png")));

setTitle("PayRoll");

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

setBounds(450, 190, 1014, 597);

contentPane = **new** JPanel();

contentPane.setBorder(**new** EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(**null**);

JLabel lblID = **new** JLabel("ID:");

lblID.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblID.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblID.setBounds(142, 15, 60, 21);

contentPane.add(lblID);

textFieldID = **new** JTextField();

textFieldID.setColumns(10);

textFieldID.setBounds(203, 15, 150, 21);

contentPane.add(textFieldID);

JLabel lblField = **new** JLabel("FIELD:");

lblField.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblField.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblField.setBounds(109, 42, 93, 21);

contentPane.add(lblField);

JComboBox <String> comboBox\_1 = **new** JComboBox <String>();

comboBox\_1.setBounds(203, 42, 150, 21);

contentPane.add(comboBox\_1);

comboBox\_1.addItem("Manager");

comboBox\_1.addItem("Accountant");

comboBox\_1.addItem("Technician");

comboBox\_1.addItem("Instructor");

JLabel lblBasicPay = **new** JLabel("BASIC PAY:");

lblBasicPay.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblBasicPay.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblBasicPay.setBounds(92, 69, 110, 21);

contentPane.add(lblBasicPay);

textFieldBasicPay = **new** JTextField();

textFieldBasicPay.setEditable(**false**);

textFieldBasicPay.setColumns(10);

textFieldBasicPay.setBounds(203, 69, 150, 21);

contentPane.add(textFieldBasicPay);

JLabel lblPhilHealth = **new** JLabel("PHILHEALTH:");

lblPhilHealth.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblPhilHealth.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblPhilHealth.setBounds(92, 96, 110, 21);

contentPane.add(lblPhilHealth);

textFieldPhilHealth = **new** JTextField();

textFieldPhilHealth.setEditable(**false**);

textFieldPhilHealth.setColumns(10);

textFieldPhilHealth.setBounds(203, 96, 150, 21);

contentPane.add(textFieldPhilHealth);

JLabel lblGross = **new** JLabel("GROSS:");

lblGross.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblGross.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblGross.setBounds(109, 123, 93, 21);

contentPane.add(lblGross);

textFieldGross = **new** JTextField();

textFieldGross.setEditable(**false**);

textFieldGross.setColumns(10);

textFieldGross.setBounds(203, 123, 150, 21);

contentPane.add(textFieldGross);

JLabel lblTax = **new** JLabel("TAX:");

lblTax.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblTax.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblTax.setBounds(129, 150, 73, 21);

contentPane.add(lblTax);

textFieldTax = **new** JTextField();

textFieldTax.setEditable(**false**);

textFieldTax.setColumns(10);

textFieldTax.setBounds(203, 150, 150, 21);

contentPane.add(textFieldTax);

JLabel lblName = **new** JLabel("NAME:");

lblName.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblName.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblName.setBounds(442, 15, 89, 21);

contentPane.add(lblName);

textFieldName = **new** JTextField();

textFieldName.setColumns(10);

textFieldName.setBounds(532, 15, 150, 21);

contentPane.add(textFieldName);

JLabel lblWorkingHours = **new** JLabel("WORKING HOURS:");

lblWorkingHours.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblWorkingHours.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblWorkingHours.setBounds(353, 42, 178, 21);

contentPane.add(lblWorkingHours);

textFieldWorkingHours = **new** JTextField();

textFieldWorkingHours.setColumns(10);

textFieldWorkingHours.setBounds(532, 42, 150, 21);

contentPane.add(textFieldWorkingHours);

JLabel lblOTP = **new** JLabel("OTP:");

lblOTP.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblOTP.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblOTP.setBounds(442, 69, 89, 21);

contentPane.add(lblOTP);

textFieldOTP = **new** JTextField();

textFieldOTP.setEditable(**false**);

textFieldOTP.setColumns(10);

textFieldOTP.setBounds(532, 69, 150, 21);

contentPane.add(textFieldOTP);

setLocationRelativeTo(**null**);

JLabel lblSSS = **new** JLabel("SSS:");

lblSSS.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblSSS.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblSSS.setBounds(477, 123, 54, 21);

contentPane.add(lblSSS);

textFieldSSS = **new** JTextField();

textFieldSSS.setEditable(**false**);

textFieldSSS.setColumns(10);

textFieldSSS.setBounds(532, 123, 150, 21);

contentPane.add(textFieldSSS);

JLabel lblTotalDeduction = **new** JLabel("TOTAL DEDUCTION:");

lblTotalDeduction.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblTotalDeduction.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblTotalDeduction.setBounds(42, 177, 160, 21);

contentPane.add(lblTotalDeduction);

textFieldTotalDeduction = **new** JTextField();

textFieldTotalDeduction.setEditable(**false**);

textFieldTotalDeduction.setColumns(10);

textFieldTotalDeduction.setBounds(203, 177, 150, 21);

contentPane.add(textFieldTotalDeduction);

JLabel lblNet = **new** JLabel("NET:");

lblNet.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblNet.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblNet.setBounds(471, 150, 60, 21);

contentPane.add(lblNet);

textFieldNet = **new** JTextField();

textFieldNet.setEditable(**false**);

textFieldNet.setColumns(10);

textFieldNet.setBounds(532, 150, 150, 21);

contentPane.add(textFieldNet);

lblPagIbig = **new** JLabel("PAG-IBIG:");

lblPagIbig.setHorizontalAlignment(SwingConstants.***RIGHT***);

lblPagIbig.setFont(**new** Font("Consolas", Font.***BOLD***, 18));

lblPagIbig.setBounds(421, 96, 110, 21);

contentPane.add(lblPagIbig);

textFieldPagIbig = **new** JTextField();

textFieldPagIbig.setEditable(**false**);

textFieldPagIbig.setColumns(10);

textFieldPagIbig.setBounds(532, 96, 150, 21);

contentPane.add(textFieldPagIbig);

btnDelete = **new** JButton("DELETE");

btnDelete.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**int** i = tablePayRoll.getSelectedRow();

**if**(i>=0) {

**int** n = JOptionPane.*showConfirmDialog*(

**null**,

"Are you sure you want to permanently remove this item?",

"3MBC Payroll System",

JOptionPane.***YES\_NO\_OPTION***);

**if**(n==JOptionPane.***YES\_OPTION***) {

model.removeRow(i);

queue.dequeue();

ImageIcon logo = **new** ImageIcon(getClass().getResource("/res/3MBC Payroll System LOGO.png"));

Image img3MBC = logo.getImage();

Image modifiedLogo = img3MBC.getScaledInstance(100, 100, java.awt.Image.***SCALE\_SMOOTH***);

logo = **new** ImageIcon(modifiedLogo);

JOptionPane.*showMessageDialog*(**null**, "Deleted Successfully","3MBC Payroll System", JOptionPane.***INFORMATION\_MESSAGE***, logo);

}

}**else** {

JFrame errorFrame = **new** JFrame();

Toolkit.*getDefaultToolkit*().beep();

JOptionPane.*showMessageDialog*(errorFrame, "Please Select a Row First","3MBC Payroll System", JOptionPane.***ERROR\_MESSAGE***);

}

}

});

btnDelete.setBounds(709, 52, 89, 23);

contentPane.add(btnDelete);

btnClear = **new** JButton("CLEAR");

btnClear.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

textFieldID.setText("");

textFieldName.setText("");

textFieldBasicPay.setText("");

textFieldPhilHealth.setText("");

textFieldPagIbig.setText("");

textFieldOTP.setText("");

textFieldSSS.setText("");

textFieldWorkingHours.setText("");

textFieldGross.setText("");

textFieldNet.setText("");

textFieldTax.setText("");

textFieldTotalDeduction.setText("");

}

});

btnClear.setBounds(709, 92, 89, 23);

contentPane.add(btnClear);

btnUpdate = **new** JButton("UPDATE");

btnUpdate.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**int** i = tablePayRoll.getSelectedRow();

**if**(i>=0) {

model.setValueAt(textFieldID.getText(), i, 0);

model.setValueAt(textFieldName.getText(), i, 1);

model.setValueAt(comboBox\_1.getSelectedItem(), i, 2);

model.setValueAt(textFieldWorkingHours.getText(), i, 3);

model.setValueAt(textFieldBasicPay.getText(), i, 4);

model.setValueAt(textFieldOTP.getText(), i, 5);

model.setValueAt(textFieldPhilHealth.getText(), i, 6);

model.setValueAt(textFieldPagIbig.getText(), i, 7);

model.setValueAt(textFieldGross.getText(), i, 8);

model.setValueAt(textFieldSSS.getText(), i, 9);

model.setValueAt(textFieldTax.getText(), i, 10);

model.setValueAt(textFieldTotalDeduction.getText(), i, 11);

model.setValueAt(textFieldNet.getText(), i, 12);

}**else** {

JFrame errorFrame = **new** JFrame();

Toolkit.*getDefaultToolkit*().beep();

JOptionPane.*showMessageDialog*(errorFrame, "Please Select a Row First","3MBC Payroll System", JOptionPane.***ERROR\_MESSAGE***);

}

}

});

btnUpdate.setBounds(709, 132, 89, 23);

contentPane.add(btnUpdate);

scrollPane = **new** JScrollPane();

scrollPane.setBounds(10, 208, 977, 342);

contentPane.add(scrollPane);

tablePayRoll = **new** JTable();

tablePayRoll.addMouseListener(**new** MouseAdapter() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

**int** i= tablePayRoll.getSelectedRow();

**if**(i>=0) {

textFieldID.setText(model.getValueAt(i,0).toString());

textFieldName.setText(model.getValueAt(i,1).toString());

comboBox\_1.setSelectedItem(model.getValueAt(i,2).toString());

textFieldWorkingHours.setText(model.getValueAt(i,3).toString());

textFieldBasicPay.setText(model.getValueAt(i,4).toString());

textFieldOTP.setText(model.getValueAt(i,5).toString());

textFieldPhilHealth.setText(model.getValueAt(i,6).toString());

textFieldPagIbig.setText(model.getValueAt(i,7).toString());

textFieldGross.setText(model.getValueAt(i,8).toString());

textFieldSSS.setText(model.getValueAt(i,9).toString());

textFieldTax.setText(model.getValueAt(i,10).toString());

textFieldTotalDeduction.setText(model.getValueAt(i,11).toString());

textFieldNet.setText(model.getValueAt(i,12).toString());

}**else** {

JFrame errorFrame = **new** JFrame();

Toolkit.*getDefaultToolkit*().beep();

JOptionPane.*showMessageDialog*(errorFrame, "Please Select a Row First","3MBC Payroll System", JOptionPane.***ERROR\_MESSAGE***);

}

}

});

model = **new** DefaultTableModel();

Object[] column = {"ID", "Name", "Field", "Working Hours", "Basic Pay", "OTP", "PhilHealth","Pag-ibig", "SSS", "Tax","Gross", "Deduction", "Net"};

Stack <Object> col = **new** Stack <Object>();

col.push(column);

**final** Object[] row = **new** Object[13];

Stack <Object> rows = **new** Stack <Object>();

rows.push(row);

model.setColumnIdentifiers(column);

tablePayRoll.setModel(model);

scrollPane.setViewportView(tablePayRoll);

btnAdd = **new** JButton("ADD");

btnAdd.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

//store to array list

//function refresh table()

//refresh loop arraylist

**if**(textFieldID.getText().equals("") || textFieldName.getText().equals("") || textFieldPhilHealth.getText().equals("")|| textFieldWorkingHours.getText().equals("") || textFieldGross.getText().equals("") || textFieldNet.getText().equals("") || textFieldTax.getText().equals("") || textFieldTotalDeduction.getText().equals("")

|| textFieldOTP.getText().equals("") || textFieldSSS.getText().equals("")) {

JFrame errorFrame = **new** JFrame();

Toolkit.*getDefaultToolkit*().beep();

JOptionPane.*showMessageDialog*(errorFrame, "Please Fill Complete Information","3MBC Payroll System", JOptionPane.***ERROR\_MESSAGE***);

} **else** {

row[0]= textFieldID.getText();

row[1]= textFieldName.getText();

row[2]= comboBox\_1.getSelectedItem().toString();

row[3]= textFieldWorkingHours.getText();

row[4]= textFieldBasicPay.getText();

row[5]= textFieldOTP.getText();

row[6]= textFieldPhilHealth.getText();

row[7]= textFieldPagIbig.getText();

row[8]= textFieldSSS.getText();

row[9]= textFieldTax.getText();

row[10]= textFieldGross.getText();

row[11]= textFieldTotalDeduction.getText();

row[12]= textFieldNet.getText();

model.addRow(row);

queue.enqueue(row);

textFieldID.setText("");

textFieldName.setText("");

textFieldPhilHealth.setText("");

textFieldOTP.setText("");

textFieldSSS.setText("");

textFieldWorkingHours.setText("");

textFieldGross.setText("");

textFieldNet.setText("");

textFieldTax.setText("");

textFieldTotalDeduction.setText("");

textFieldBasicPay.setText("");

textFieldPagIbig.setText("");

ImageIcon logo = **new** ImageIcon(getClass().getResource("/res/3MBC Payroll System LOGO.png"));

Image img3MBC = logo.getImage();

Image modifiedLogo = img3MBC.getScaledInstance(100, 100, java.awt.Image.***SCALE\_SMOOTH***);

logo = **new** ImageIcon(modifiedLogo);

JOptionPane.*showMessageDialog*(**null**, "Saved Successfully","3MBC Payroll System", JOptionPane.***INFORMATION\_MESSAGE***, logo);

}

}

});

btnAdd.setBounds(709, 16, 89, 23);

contentPane.add(btnAdd);

JButton btnBack = **new** JButton("Back");

btnBack.addMouseListener(**new** MouseAdapter() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

//Go back to Employee list

EmployeeList attendance = **new** EmployeeList();

attendance.setVisible(**true**);

Payroll\_JFrame.**this**.dispose();

}

});

btnBack.setBounds(922, 177, 65, 21);

contentPane.add(btnBack);

// JComboBox<String> comboBox = new JComboBox<String>();

// comboBox.setBounds(901, 15, 87, 21);

// contentPane.add(comboBox);

// comboBox.addItem("ID");

// comboBox.addItem("Name");

// comboBox.addItem("Field");

// comboBox.addItem("Working Hours");

// comboBox.addItem("Basic Pay");

// comboBox.addItem("OTP");

// comboBox.addItem("PhilHealth");

// comboBox.addItem("Pag-ibig");

// comboBox.addItem("Gross");

// comboBox.addItem("SSS");

// comboBox.addItem("Tax");

// comboBox.addItem("Deduction");

// comboBox.addItem("Net");

// JLabel lblSortBy = new JLabel("Sort by:");

// lblSortBy.setHorizontalAlignment(SwingConstants.RIGHT);

// lblSortBy.setFont(new Font("Consolas", Font.BOLD, 18));

// lblSortBy.setBounds(810, 16, 92, 21);

// contentPane.add(lblSortBy);

btnOK = **new** JButton("OK");

btnOK.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

**if**(textFieldID.getText().equals("") || textFieldName.getText().equals("") || textFieldWorkingHours.getText().equals("")){

JFrame errorFrame = **new** JFrame();

Toolkit.*getDefaultToolkit*().beep();

JOptionPane.*showMessageDialog*(errorFrame, "Please Fill Complete Information","3MBC Payroll System", JOptionPane.***ERROR\_MESSAGE***);

} **else** {

**int** numHrsWork = Integer.*parseInt*(textFieldWorkingHours.getText());

// Formulas //

**int** rateHour;

**if**(comboBox\_1.getSelectedItem().toString().equals("Manager")) {

rateHour=72;

} **else** **if**(comboBox\_1.getSelectedItem().toString().equals("Accountant")) {

rateHour=78;

} **else** **if**(comboBox\_1.getSelectedItem().toString().equals("Technician")) {

rateHour=65;

} **else** {

rateHour=24;

}

**int** SSS = 500;

**int** pagIbig = 100;

**int** NumOThrs;

**if**(numHrsWork >=40) {

NumOThrs = numHrsWork - 40;

} **else** {

NumOThrs = 0;

}

**int** OThrsPay = (**int**) (NumOThrs \*rateHour \*1.25);

**int** basicPay = rateHour \*40;

**int** grossPay = basicPay + OThrsPay;

**int** tax = (**int**) (grossPay \* 0.1);

**int** philHealth = (**int**) (grossPay \* 0.02);

**int** totDeduction = tax + philHealth + SSS + pagIbig;

**int** netPay = grossPay-totDeduction;

textFieldOTP.setText(Integer.*toString*(OThrsPay));

textFieldPhilHealth.setText(Integer.*toString*(philHealth));

textFieldSSS.setText(Integer.*toString*(SSS));

textFieldGross.setText(Integer.*toString*(grossPay));

textFieldTax.setText(Integer.*toString*(tax));

textFieldNet.setText(Integer.*toString*(netPay));

textFieldTotalDeduction.setText(Integer.*toString*(totDeduction));

textFieldBasicPay.setText(Integer.*toString*(basicPay));

textFieldPagIbig.setText(Integer.*toString*(pagIbig));

}

}

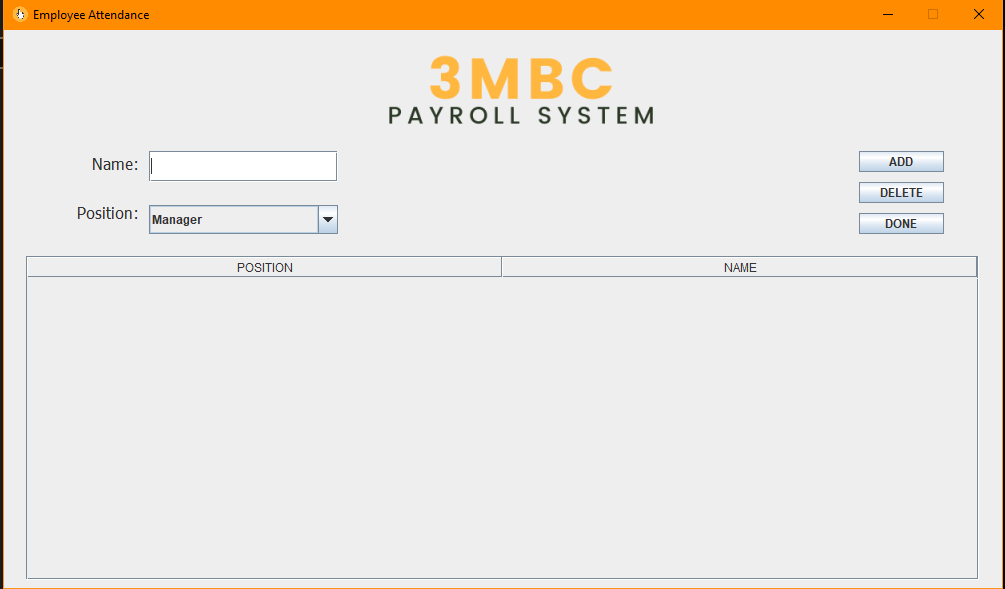
});

btnOK.setBounds(593, 177, 89, 23);

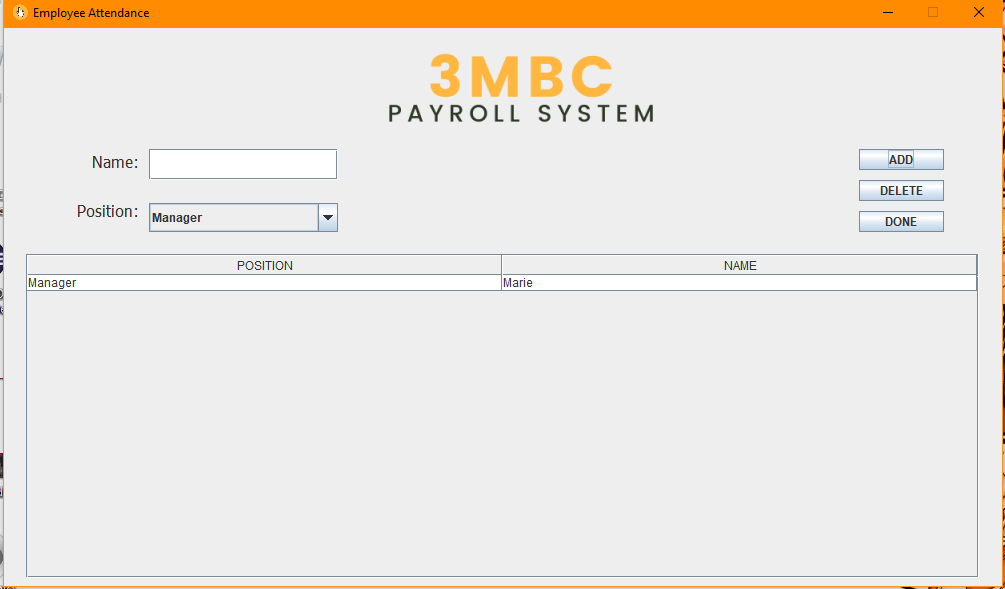
contentPane.add(btnOK);

}

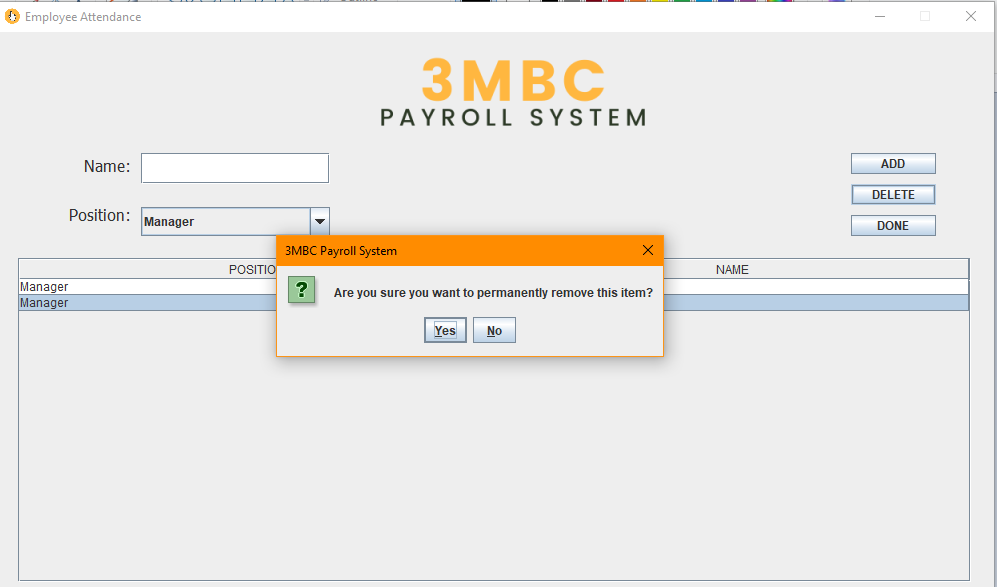
# } **Output**

****

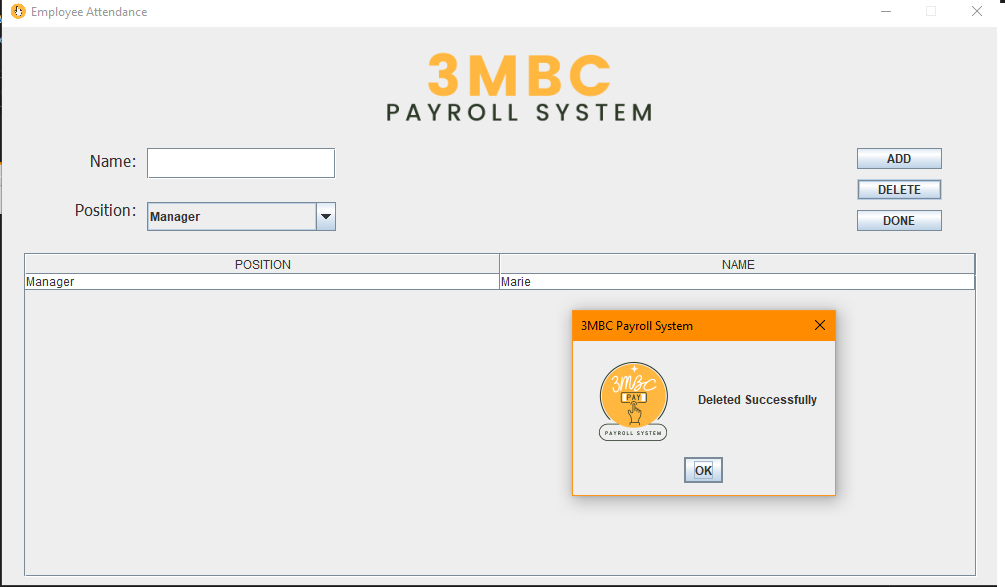
1. Employee Attendance requires for admin to input their names and choose their job position from a *drop-down menu.*

****

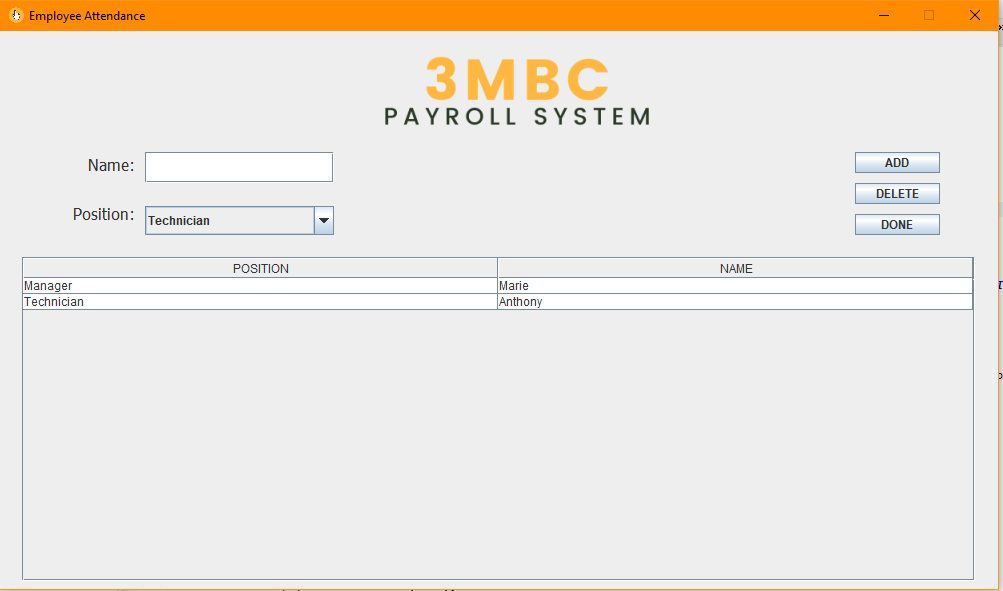
1. After employees have provided their names and work designations, the user may modify and select whether to add additional employees or delete existing entries.

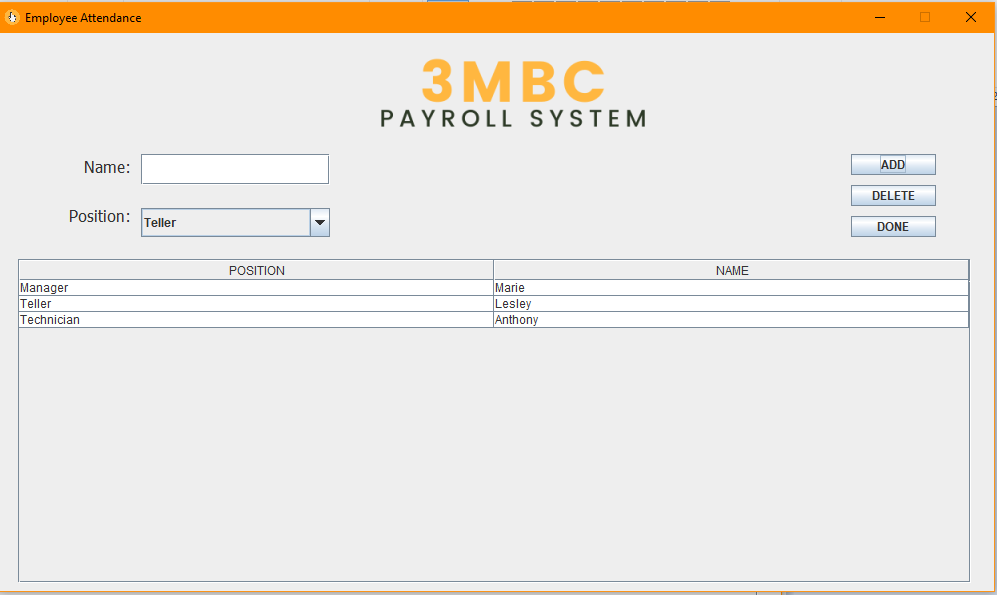


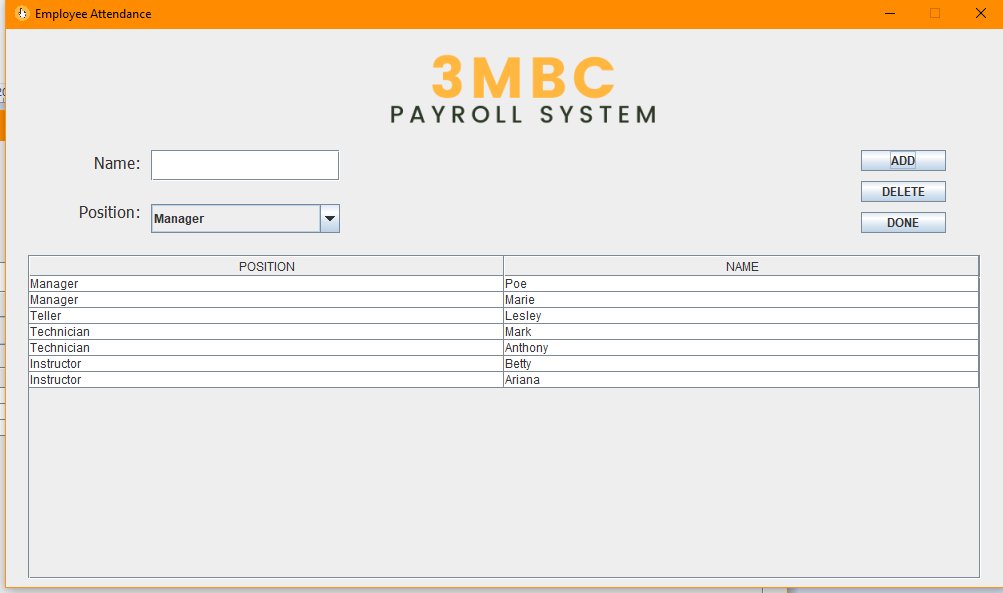
1. The sample output demonstrates how to delete an entry from a table. When a user clicks the delete button, the system will prompt the user to confirm or deny the deletion. It is significant to prevent accidental clicks and to double-check all system activities.

****

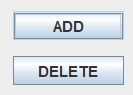
1. When the user confirms the deletion of an entry, the system notifies the user that the deletion has been completed and the entry cannot be recovered.



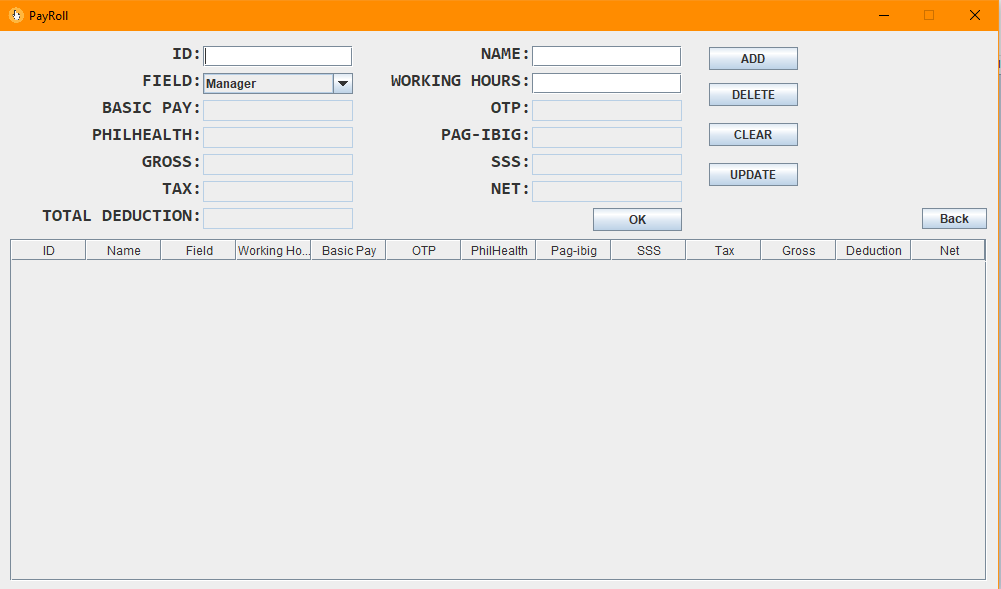


****

1. It is an example of the output when two or more workers are added to the table. Employee attendance simplifies and speeds up the user's life by minimizing the need for multiple transactions. This also shows the hierarchy of employees in the company whereas the manager is the always the first on the list then followed by the other positions.



1. The Payroll\_JFrame class add and delete buttons were utilized to integrate the Queue into the project whilst the LinkedList was implemented in the EmployeeList’s button and table.



1. The “DONE” button will go to the PayRoll\_JFrame class which is the main purpose of the payroll system.

Honor Pledge: “I accept responsibility for my role in ensuring the integrity of the work submitted by the group in which I participated.”