**TECHNOLOGICAL INSTITUTE OF THE PHILIPPINES**

**QUEZON CITY**

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

**CS 201 - Data Structures and Algorithms**

|  |  |
| --- | --- |
| **Name: Aristotle Buenaventura** | **Date: 10/04/2021** |
| **Program/Section: IT21S1** | **Instructor: Ms. Rosmina Joy M. Cabauatan** |
| **Assessment Task: Assignment 5. | Final Project Development** | |

# **BinarySearch** **class (Tree)**

package Assignment5;

import javax.swing.JOptionPane;

class node

{

node left;

node right;

String name;

String ID;

String field;

public node(String name, String ID, String field) {

left = null;

right = null;

this.name = name;

this.ID = ID;

this.field = field;

}

}

class BinaryTree {

public static node root;

String name;

String ID;

String field;

BinaryTree() {

root = null;

}

public void create(Object namePayroll, Object IDPayroll, Object fieldPayroll)

{

int flag = 0;

node ptr, any;

name = (String) namePayroll;

ID = (String) IDPayroll;

any = root;

while (any != null) {

if (any.ID.compareTo(ID) == 0) {

flag = 1;

break;

}

if ((any.ID).compareTo(ID) > 0) {

any = any.left;

} else {

any = any.right;

}

}

field = (String) fieldPayroll;

node temp = new node(name, ID, field);

if (root == null) {

root = temp;

}

else {

ptr = root;

while (flag == 0) {

if ((temp.name).compareTo(ptr.name) > 0) {

if (ptr.right != null) {

ptr = ptr.right;

} else {

ptr.right = temp;

flag = 1;

}

}

if ((temp.name).compareTo(ptr.name) < 0) {

if (ptr.left != null) {

ptr = ptr.left;

} else {

ptr.left = temp;

flag = 1;

}

}

}

}

}

public void update(String updateName, String updateID, String updateField) {

int flag = 0;

node ptr;

String key = updateName;

ptr = root;

while (ptr != null) {

if ((ptr.name).compareTo(key) == 0) {

flag = 1;

break;

}

if ((ptr.name).compareTo(key) > 0) {

ptr = ptr.left;

} else {

ptr = ptr.right;

}

}

if (flag == 1) {

String newID;

newID = updateID;

ptr.ID = newID;

ptr.field = updateField;

}

}

public void search(String searchPayroll) {

int flag = 0;

node ptr;

String key = searchPayroll;

ptr = root;

while (ptr != null)

{

if (ptr.name.toLowerCase().compareTo(key) == 0) {

flag = 1;

SearchWindow search = new SearchWindow();

search.setVisible(true);

search.SearchWindow(ptr.name,ptr.ID,ptr.field);

break;

}

if ((ptr.name).toLowerCase().compareTo(key) > 0) {

ptr = ptr.left;

} else {

ptr = ptr.right;

}

}

if (flag == 0) {

JOptionPane.showMessageDialog( null, "-----RECORD NOT FOUND-----","3MBC Payroll System", JOptionPane.ERROR\_MESSAGE);

}

}

public void delete(String deleteName) {

int flag = 0;

node ptr, parent = null;

String key = deleteName;

ptr = root;

while (ptr != null) {

if ((ptr.name).compareTo(key) == 0)

{

flag = 1;

break;

}

if ((ptr.name).compareTo(key) > 0)

{

parent = ptr;

ptr = ptr.left;

}

else

{

parent = ptr;

ptr = ptr.right;

}

}

System.out.println(ptr.name);

if (flag == 1) {

if (ptr.left != null && ptr.right == null) {

if (parent.left == ptr) {

parent.left = ptr.left;

} else {

parent.right = ptr.left;

}

}

if (ptr.left == null && ptr.right != null) {

if (parent.left == ptr) {

parent.left = ptr.right;

} else {

parent.right = ptr.right;

}

}

if (ptr.left != null && ptr.right != null) {

node p;

p = ptr.left;

System.out.println("ptr= " + ptr.name);

while (p.right != null) {

parent = p;

p = p.right;

}

ptr.name = p.name;

ptr.ID = p.ID;

if (p.left != null) {

parent.right = p.left;

} else {

ptr.left = null;

}

}

if (ptr.left == null && ptr.right == null) {

if (parent.left == ptr) {

parent.left = null;

} else {

parent.right = null;

}

}

}

}

}

# **Graph class (Graph)**

package Assignment5;

import java.util.ArrayList;

import java.util.LinkedList;

import java.util.List;

class Employees {

int index;

String name; //Employee Name

String position; //Employee Position

LinkedList<Employees> connectedEmployees = new LinkedList<>();

public Employees(int i, String name, String position) {

index = i;

this.name = name;

this.position = position;

}

}

public class Graph {

int size =0;

List<Employees> company = new ArrayList<>();

void addEmployee(String code, String name){

int i = size++;

Employees employee = new Employees(i,name, code);

company.add(employee);

}

Employees companyLookup(String code){

for(Employees c:company){

if(c.position.equalsIgnoreCase(code)){

return c;

}

}

return null;

}

void connectEmp(String code1, String code2){

Employees c1 = companyLookup(code1);

Employees c2 = companyLookup(code2);

if( c1 == null || c2 == null){

System.out.println("Cannot connect, Invalid Company position.");

return;

}

c1.connectedEmployees.add(c2);

}

public void findBFSpath(String position) {

boolean[] visited = new boolean[size];

Employees employee = companyLookup(position);

System.out.println();

System.out.println(" Meet the people who made this project possible: ");

System.out.println();

LinkedList<Employees> bfs = new LinkedList<>();

bfs.add(employee);

visited[employee.index] = true;

while(!bfs.isEmpty()){

Employees currentElement = bfs.poll();

System.out.print(" >> "+ currentElement.name);

for(Employees curr : currentElement.connectedEmployees){

if(!visited[curr.index]){

visited[curr.index] = true;

bfs.add(curr);

}

}

}

}

public void findDFSpath(String position) {

boolean[] visited = new boolean[size];

Employees employee = companyLookup(position);

System.out.println();

System.out.println(" Meet the people who made this project possible");

System.out.println();

DFS(employee, visited);

}

public void DFS(Employees company, boolean[] visited) {

visited[company.index] = true;

for(Employees c :company.connectedEmployees){

if(!visited[c.index]){

DFS(c,visited);

}

}

System.out.print("\n " + company.name);

}

}

# **Queue class (Queue)**

package Assignment5;

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++;

}

}

# **SearchWindow class (JFrame)**

package Assignment5;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.JLabel;

import javax.swing.JTextField;

import javax.swing.JButton;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

import java.awt.Toolkit;

public class SearchWindow extends JFrame {

public SearchWindow() {

setTitle("Search");

setIconImage(Toolkit.getDefaultToolkit().getImage(SearchWindow.class.getResource("/res/3.png")));

}

private JPanel contentPane;

private JTextField textFieldName;

private JTextField textFieldID;

private JTextField textFieldField;

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

SearchWindow frame = new SearchWindow();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

public void SearchWindow(String name, String ID, String field) {

setTitle("RESULT");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 251, 197);

contentPane = new JPanel();

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

setContentPane(contentPane);

contentPane.setLayout(null);

JLabel lblNewLabel = new JLabel("--RECORD FOUND--");

lblNewLabel.setBounds(70, 11, 128, 14);

contentPane.add(lblNewLabel);

textFieldName = new JTextField();

textFieldName.setEditable(false);

textFieldName.setBounds(79, 67, 86, 20);

contentPane.add(textFieldName);

textFieldName.setColumns(10);

textFieldName.setText(name);

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

lblName.setBounds(23, 70, 46, 14);

contentPane.add(lblName);

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

lblID.setBounds(23, 39, 46, 14);

contentPane.add(lblID);

textFieldID = new JTextField();

textFieldID.setEditable(false);

textFieldID.setColumns(10);

textFieldID.setBounds(79, 36, 86, 20);

contentPane.add(textFieldID);

textFieldID.setText(ID);

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

lblField.setBounds(23, 104, 46, 14);

contentPane.add(lblField);

textFieldField = new JTextField();

textFieldField.setEditable(false);

textFieldField.setColumns(10);

textFieldField.setBounds(79, 101, 86, 20);

contentPane.add(textFieldField);

textFieldField.setText(field);

JButton btnNewButton = new JButton("OK");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

SearchWindow.this.dispose();

}

});

btnNewButton.setBounds(173, 124, 52, 23);

contentPane.add(btnNewButton);

}

}

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

package Assignment5;

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;

private JTextField textFieldSearch;

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() {

BinaryTree b = new BinaryTree();

Graph g = new Graph();

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);

// String name = model.getColumnName(1);

// b.delete(name);

}

}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);

b.update(textFieldName.getText(),textFieldID.getText(),comboBox\_1.getSelectedItem().toString());

}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);

b.create(row[1],row[0],row[2]);

}

}

});

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 Dashboard

AboutUs About= new AboutUs();

About.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);

JLabel lblSearch = new JLabel("SEARCH:");

lblSearch.setHorizontalAlignment(SwingConstants.RIGHT);

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

lblSearch.setBounds(824, 15, 73, 21);

contentPane.add(lblSearch);

textFieldSearch = new JTextField();

textFieldSearch.setColumns(10);

textFieldSearch.setBounds(905, 14, 93, 21);

contentPane.add(textFieldSearch);

JButton btnSearch = new JButton("SEARCH");

btnSearch.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

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

JFrame errorFrame = new JFrame();

Toolkit.getDefaultToolkit().beep();

JOptionPane.showMessageDialog(errorFrame, "Please Enter the Correct Name","3MBC Payroll System", JOptionPane.ERROR\_MESSAGE);

}else {

b.search(textFieldSearch.getText().toString().toLowerCase());

}

}

});

btnSearch.setBounds(905, 40, 89, 23);

contentPane.add(btnSearch);

}

}

# **AboutUs class (JFrame)**

package Assignment5;

import java.util.\*;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.border.EmptyBorder;

import javax.swing.text.View;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextArea;

import java.awt.Font;

import java.awt.Image;

import java.awt.Color;

import javax.swing.JTextPane;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import java.awt.Toolkit;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

public class AboutUs extends JFrame {

/\*\*

\*

\*/

private static final long serialVersionUID = 2L;

private JPanel contentPane;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

AboutUs frame = new AboutUs();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public AboutUs() {

setIconImage(Toolkit.getDefaultToolkit().getImage(AboutUs.class.getResource("/res/3.png")));

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(450, 190, 1014, 597);

contentPane = new JPanel();

contentPane.setBackground(new Color(255, 153, 0));

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

setTitle("About Us");

setContentPane(contentPane);

setResizable(false);

setLocationRelativeTo(null);

contentPane.setLayout(null);

JTextArea txtPayroll = new JTextArea();

txtPayroll.setBounds(225, 99, 571, 176);

txtPayroll.setWrapStyleWord(true);

txtPayroll.setFont(new Font("Dialog", Font.PLAIN, 15));

txtPayroll.setLineWrap(true);

txtPayroll.setForeground(new Color(255, 255, 255));

txtPayroll.setBackground(new Color(255, 153, 0));

txtPayroll.setEditable(false);

txtPayroll.setText("\tThe 3MBC Payroll System is designed to digitize the current manual systems used by companies nowadays. It was developed to help the manager/administrator to arrange the employees' salaries in a more efficient, secure, and timely manner. It encrypts sensitive data and information and enables storage over an extended length of time while allowing for easy access and modification. This system can keep and view digital records without obtaining redundant entries. The project comprises how to manage employees' data effectively and provide enhanced services to employees.");

contentPane.add(txtPayroll);

JLabel lblBehind = new JLabel("Behind the Name");

lblBehind.setBounds(370, 286, 289, 53);

lblBehind.setForeground(new Color(255, 255, 51));

lblBehind.setBackground(new Color(255, 255, 255));

lblBehind.setFont(new Font("Consolas", Font.PLAIN, 35));

contentPane.add(lblBehind);

JLabel lblAbout = new JLabel("About Us");

lblAbout.setBounds(427, 55, 188, 41);

contentPane.add(lblAbout);

lblAbout.setForeground(new Color(255, 255, 51));

lblAbout.setFont(new Font("Consolas", Font.PLAIN, 35));

JTextPane txtpnForMoreDe = new JTextPane();

txtpnForMoreDe.setBounds(225, 326, 571, 71);

txtpnForMoreDe.setFont(new Font("Dialog", Font.PLAIN, 15));

txtpnForMoreDe.setForeground(new Color(255, 255, 255));

txtpnForMoreDe.setBackground(new Color(255, 153, 0));

txtpnForMoreDe.setText("\tThe developers wish to come up with a very intriguing name. Something eye-catching and short. They decided to use their surnames for this project namely 'Mendoza, Moraga and Muñez' for the 3M then B and C for 'Buenaventura and Cruz'." );

contentPane.add(txtpnForMoreDe);

JLabel lblContact1 = new JLabel("Problem with the program?");

lblContact1.setBounds(225, 461, 192, 14);

lblContact1.setFont(new Font("Tahoma", Font.BOLD, 13));

lblContact1.setForeground(new Color(255, 255, 255));

contentPane.add(lblContact1);

JLabel lblContact2 = new JLabel("Reach us at 0932498990 or email us at 3MBC@gmail.com");

lblContact2.setBounds(225, 480, 356, 14);

lblContact2.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblContact2.setForeground(new Color(255, 255, 255));

contentPane.add(lblContact2);

JButton btnNewButton = new JButton("Home");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

}

});

btnNewButton.setForeground(Color.BLACK);

btnNewButton.setFont(new Font("Lucida Grande", Font.PLAIN, 14));

btnNewButton.addMouseListener(new MouseAdapter() {

@Override

public void mouseClicked(MouseEvent e) {

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

Frame.setVisible(true);

AboutUs.this.dispose();

}

});

btnNewButton.setBounds(462, 506, 89, 29);

contentPane.add(btnNewButton);

JLabel Logo = new JLabel("");

Logo.setBounds(850, 393, 158, 176);

contentPane.add(Logo);

Logo.setIcon(new ImageIcon(new javax.swing.ImageIcon(getClass().getResource("/res/3MBC Payroll System LOGO.png")).getImage().getScaledInstance(150, 150, Image.SCALE\_SMOOTH)));

JButton meet= new JButton("Click Me!");

meet.setFont(new Font("Tahoma", Font.BOLD, 11));

meet.setBounds(462, 408, 89, 29);

contentPane.add(meet);

meet.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

Scanner sc= new Scanner (System.in);

Graph graph = new Graph();

graph.addEmployee("Manager","Gracia");

graph.addEmployee("Main Programmer 1","Aristotle");

graph.addEmployee("Main Programmer 2","Yunice");

graph.addEmployee("Sub Programmer","Jackilou");

graph.addEmployee("System Analyst","Kathlynne");

graph.connectEmp("Manager","Main Programmer 1");

graph.connectEmp("Manager","Main Programmer 2");

graph.connectEmp("Manager","Sub Programmer");

graph.connectEmp("Manager","System Analyst");

int choice;

do {

System.out.println();

System.out.println();

System.out.print("Choose what type of transversal (1 or 2):");

System.out.print("\n 1) BFS \t 2) DFS \t 3)Exit");

System.out.println();

System.out.println();

System.out.print("Answer: ");

choice = sc.nextInt();

switch(choice) {

case 1:

graph.findBFSpath("Manager");break;

case 2:

graph.findDFSpath("Manager");break;

case 3:

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

payroll.setVisible(true);

payroll.setExtendedState(JFrame.NORMAL);

payroll.setAlwaysOnTop(true);

payroll.requestFocus();

payroll.setAlwaysOnTop(false);

AboutUs.this.dispose();

;break;

default:

return;

}

} while(choice<3);

}

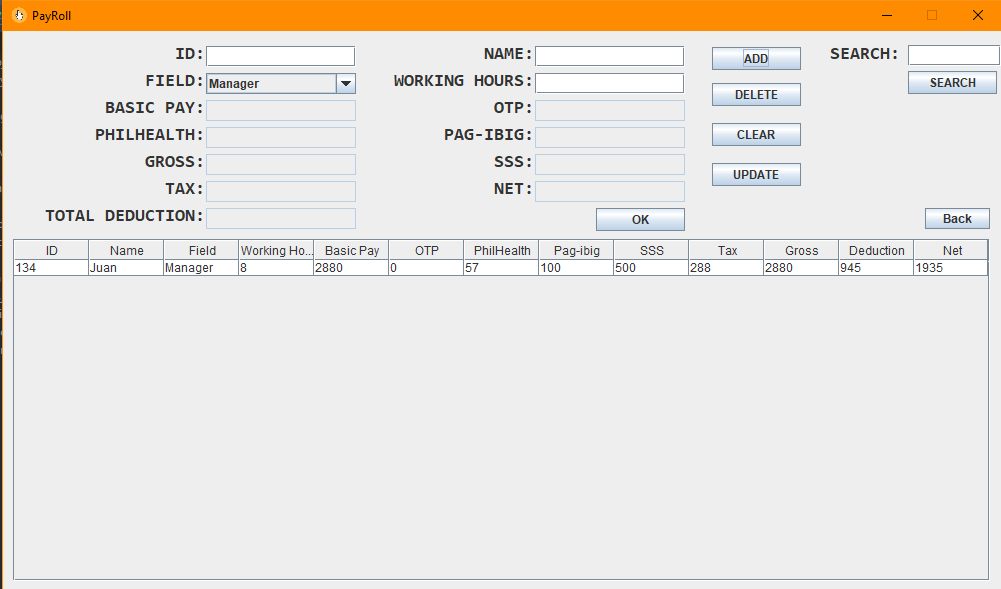
});

}

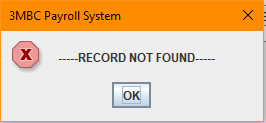
}

# **Output**

1) The developers added a search text field and a search button to the payroll area to make it easier to locate an employee's data. This button helps the admin avoid wasting time and makes their job more efficient.

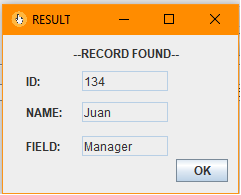


2) Once users input an employee's name that does not appear on the list of employees, a warning will appear. It will indicate that no record was located.

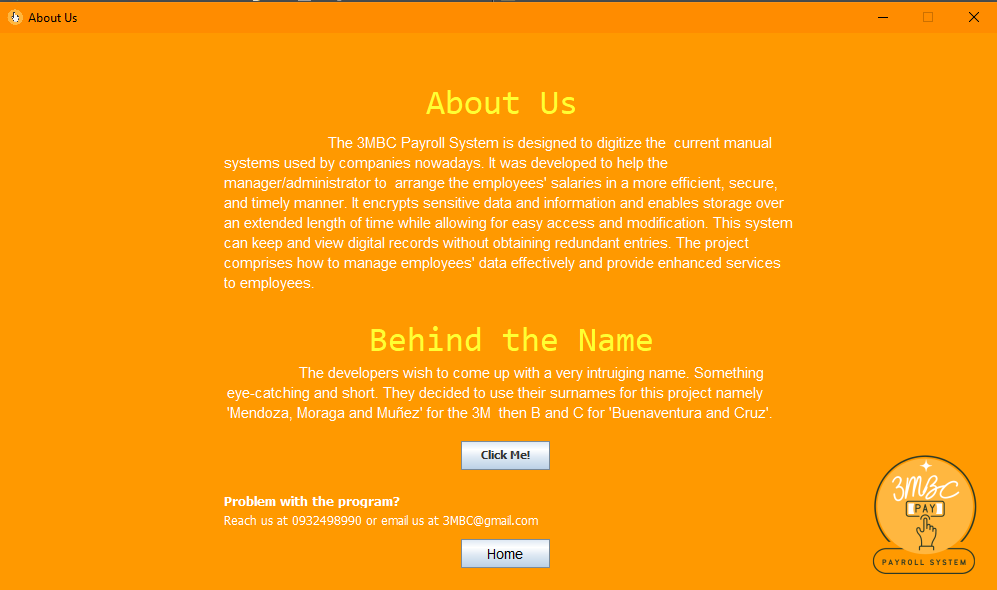


.

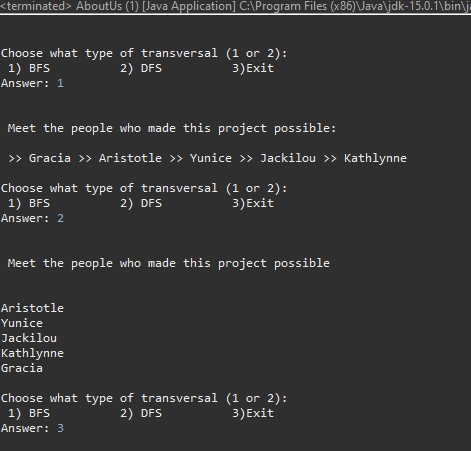
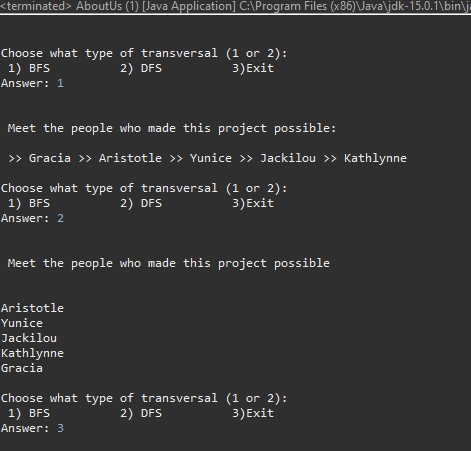
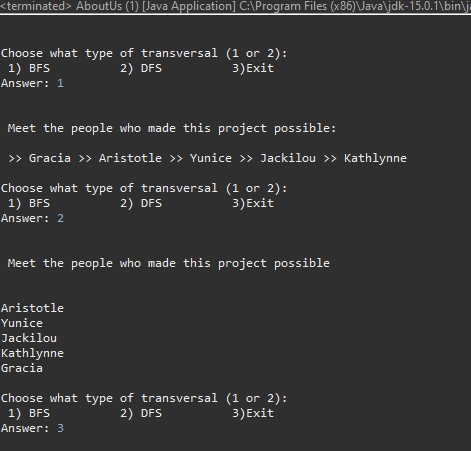
3) The employee's name should be provided when using the search button. When the name appears on the list of employees, the system will prompt for the employee's details. These include the employee's identification number, name, and expertise.



4) The developers may be found in the *About Us* section. To view the individuals behind the system, users can select between two graphs: breadth-first search and depth-first search. This is to be done with the *Click Me* button. The *Home* button will lead to the Payroll section of the program



5) The breadth first and depth first searches are described in detail in the about us section. These options will display in the system's console. Users will be able to select the type of graph they wish to view.



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