



**Module Code & Module Title CS4001 Programming**

**Assessment weightage & Type 30% Individual Coursework**

**Year and Semester 2024 Spring**

**Student’s Name: Janak Devkota**

**Group: C6**

**College ID: NP04CP4S230019**

**London Met ID: 23048806 Assessment Due Date: March 29, 2024**

**Assessment Submission Date: May 10, 2024**

Table of Contents

[1. Introduction 1](#_Toc164728077)

[2. Class Diagram 2](#_Toc164728078)

[I. Class Diagram Of Teacher Class 2](#_Toc164728079)

[II. Class Diagram Of Lecturer Class 2](#_Toc164728080)

[III. Class Diagram Of Tutor Class 2](#_Toc164728081)

[IV. Class Diagram Of TeacherGui Class 2](#_Toc164728082)

[3. Pseudocode 2](#_Toc164728083)

[4. Method Description 2](#_Toc164728084)

[5. Testing 2](#_Toc164728085)

[I. Test 1 2](#_Toc164728086)

[II. Test 2 2](#_Toc164728087)

[- Add Lecturer 2](#_Toc164728088)

[- Add Tutor 2](#_Toc164728089)

[- Grade Assignment 2](#_Toc164728090)

[- Set Salary 2](#_Toc164728091)

[- Remove Tutor 2](#_Toc164728092)

[- Display 2](#_Toc164728093)

[III. Test 3 2](#_Toc164728094)

[6. Error Detection and Correction 2](#_Toc164728095)

[I. Syntax Error 2](#_Toc164728096)

[II. Runtime Error 2](#_Toc164728097)

[III. Logical Error 2](#_Toc164728098)

[7. Conclusion 2](#_Toc164728099)

[8. Bibliography 2](#_Toc164728100)

[9. Appendix 2](#_Toc164728101)

**Table of Figure**

**List of Tables**

# Introduction

This coursework focuses on making user-friendly interfaces for programs, called graphical user interfaces (GUIs). We've already worked on some classes for this project, like teacher, lecturer, and tutor. Instead of just typing commands into a black screen, GUIs use pictures or symbols to help us communicate with computers. In Java, there are two main ways to make these GUIs: AWT and Swing. We picked Swing because it's newer and has more tools to make fancier interfaces, while still being easy on the computer's resources.

Our main class for this program is TeacherGui. We're storing information in ArrayLists. Last time, we used BlueJ to write our code, but now we're using IntelliJ IDEA.

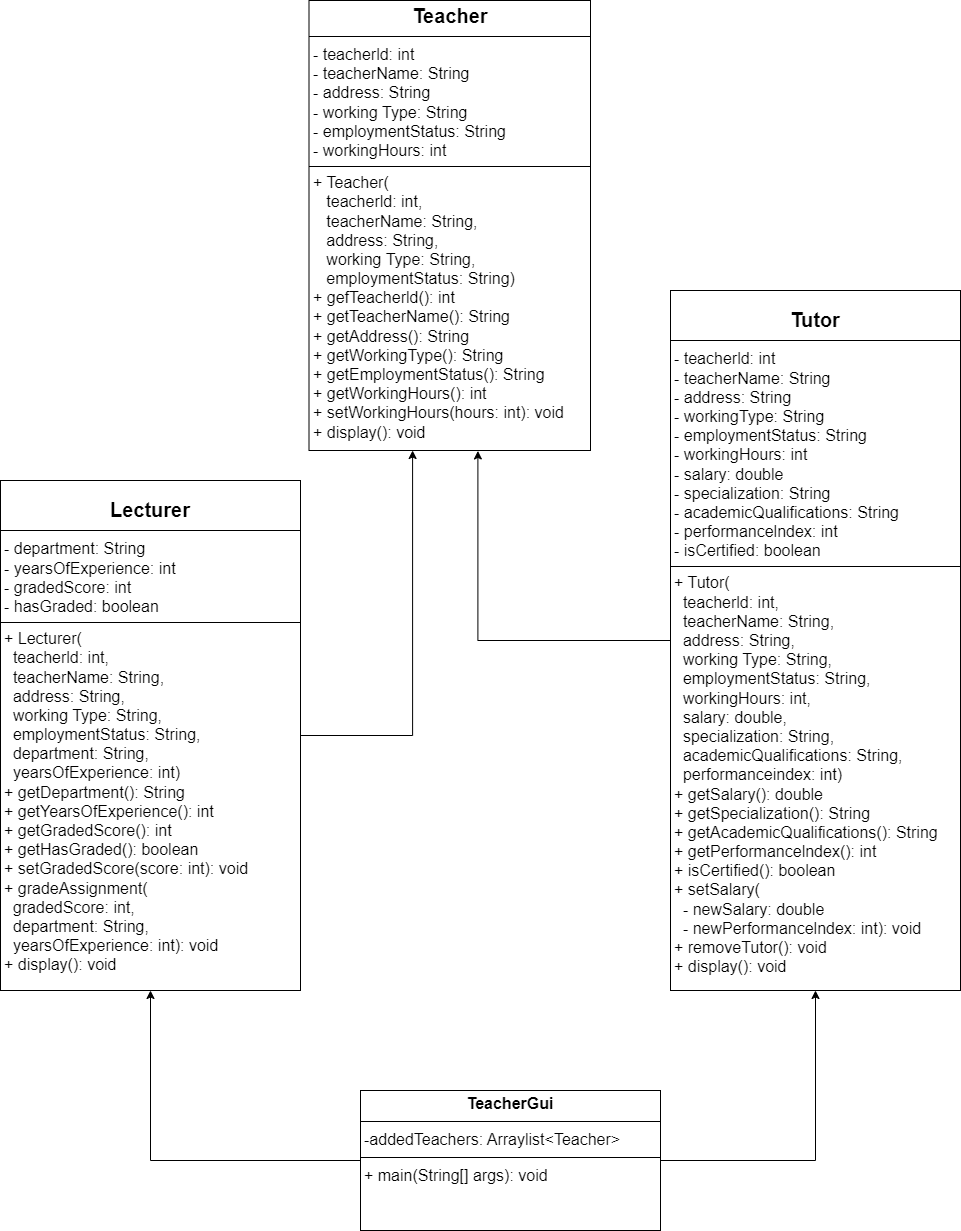
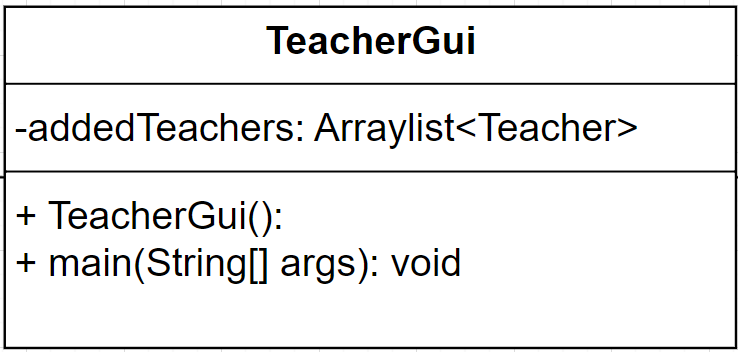
This course covers both writing code and explaining what it does. By practicing, we'll get good at making GUIs, using Java Swing, and working with ArrayLists.

The program itself has a constructor and a main method. Inside the constructor, we set up the frame and sections. Each section has its own job, like creating labels, text fields, and buttons. When we run the program, the main method calls the constructor to show the GUI.

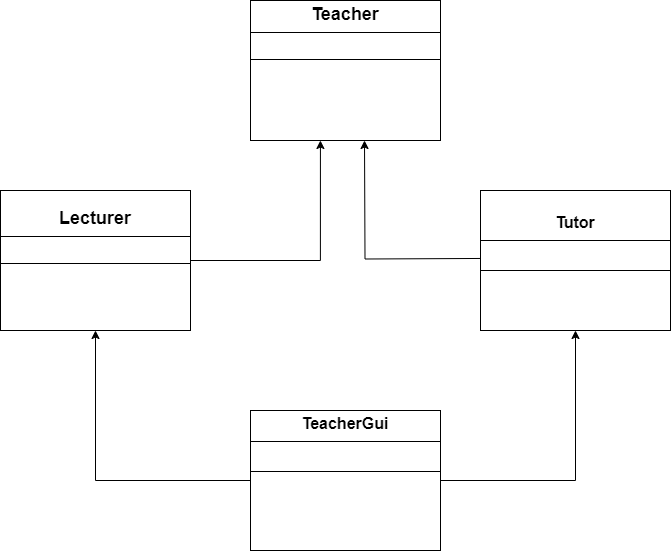
So, in short, we're learning to make programs easier to use with GUIs. We're using Java Swing because it's powerful but not too heavy on the computer. Our TeacherGui class is the heart of the program, and we're storing data in ArrayLists. We're learning both coding and explaining what we've coded, focusing on GUIs, Swing, and ArrayLists. The constructor sets up the interface, and the main method starts everything up.

As I prepare for this coursework, it's crucial to understand not just how to code but also why specific tools and techniques are chosen. By grasping GUI development principles like user experience and the efficiency of Swing over AWT, I become a more versatile programmer. Mastering data structures like ArrayLists enhances my ability to manage information effectively. This coursework not only improves my technical skills but also deepens my understanding of software design principles, laying a solid foundation for future programming challenges.

# Class Diagram

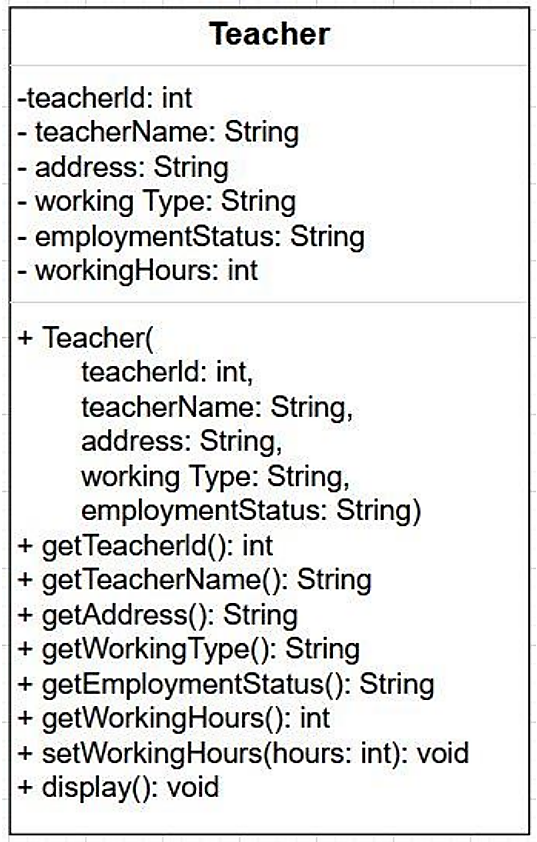


* Empty Class Diagram:



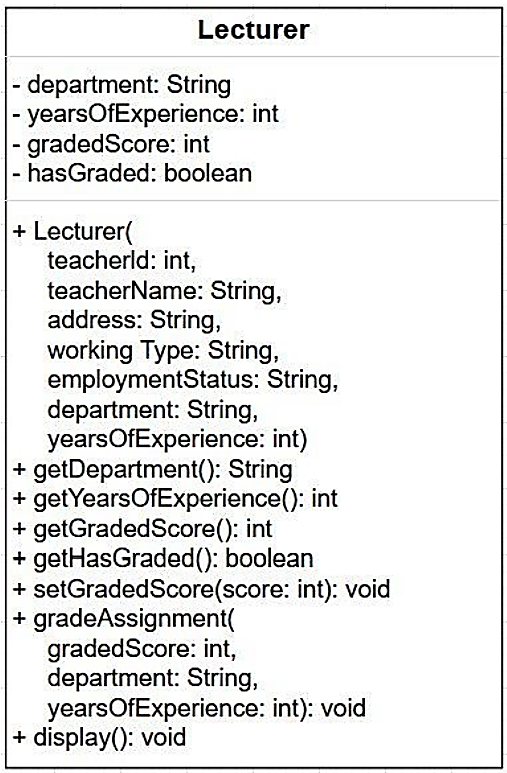
This program features a hierarchical class structure comprising three classes: Teacher, Lecturer, and Tutor. In this design, Teacher acts as the superclass, while Lecturer and Tutor serve as its subclasses. This class diagram encapsulates the hierarchical inheritance structure, showcasing the relationships and attributes of Teacher, Lecturer, and Tutor classes. The strategic use of getter and setter methods underlines the program's commitment to encapsulation, promoting a well-organized and efficient approach to handling private instance variables.

## Class Diagram Of Teacher Class



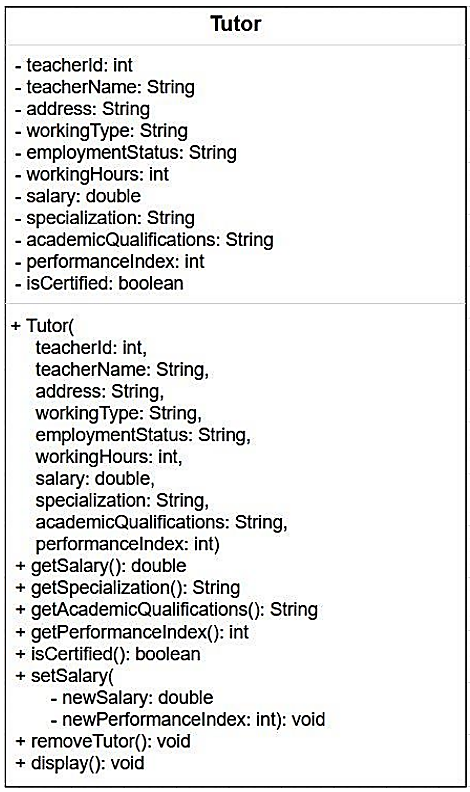
The class diagram for the "Teacher" class defines attributes like teacherId, teacherName, address, workingType, employmentStatus, and workingHours. It includes methods (getters and setters) for accessing and modifying these attributes, facilitating object-oriented management and manipulation of teacher data in a program.

## Class Diagram Of Lecturer Class



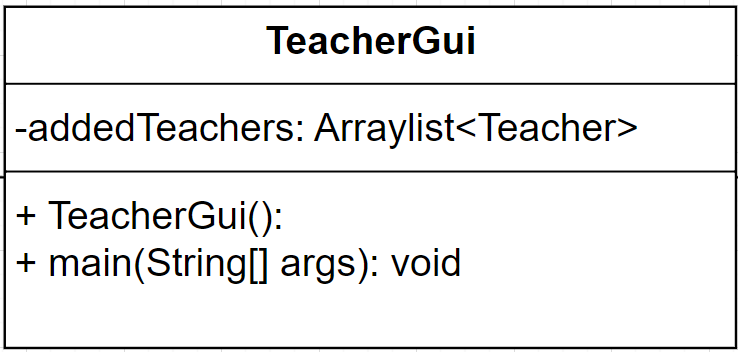
The extended "Lecturer" class introduces additional attributes: department, yearsOfExperience, gradedScore, and hasGraded. It includes methods like getDepartment, getYearsOfExperience, getGradedScore, getHasGraded, setGradedScore, gradeAssignment, and display, tailored for lecturers' unique responsibilities. This extension enhances the object-oriented framework, addressing specific lecturer requirements and functionalities.

## Class Diagram Of Tutor Class



The "Tutor" class expands upon "Teacher" and "Lecturer," featuring attributes such as salary, specialization, academic qualifications, performance index, and isCertified. Accompanying methods include getSalary, getSpecialization, getAcademicQualifications, getPerformanceIndex, isCertified, setSalary, removeTutor, and display. This comprehensive template ensures effective representation and interaction with tutor instances in software systems. Feel free to inquire about specific implementation details or further clarification.

## IV. Class Diagram Of TeacherGui Class



# Pseudocode

|  |
| --- |
| Import necessary packages  class TeacherGui:  ArrayList addedTeachers  Constructor TeacherGui():  setupFrame()  addSections()  setLocationRelativeTo(null)  setVisible(true)  Method addSections():  Add sections to the main panel:  - CreateMainMenuPanel()  - CreateAddLecturerSection()  - CreateAddTutorSection()  - CreateGradeAssignmentsSection()  - CreateSetSalaryOfTutorSection()  - CreateDisplaySection()  Method setupFrame():  Set title to "University Management System"  Set size to 1010x700  Set default close operation to EXIT\_ON\_CLOSE  Set resizable to false  Method createMainMenuPanel():  Create mainMenuPanel as a new JPanel  Set layout of mainMenuPanel to BorderLayout  Set background color of mainMenuPanel to a dark blue color  // Create panel for topic label  Create topicPanel as a new JPanel  Set background color of topicPanel to a dark blue color  Create topicLabel as a new JLabel with text "University Management \_\_\_\_\_\_\_\_\_\_\_System"  Set font of topicLabel to bold Arial with size 20  Set foreground color of topicLabel to orange  Add topicLabel to topicPanel  // Create left panel for buttons  Create buttonPanel as a new JPanel  Set background color of buttonPanel to a brown color  Set layout of buttonPanel to GridLayout with 6 rows, 1 column, and \_\_\_\_\_\_\_\_\_\_\_spacing of 5 pixels  Set border of buttonPanel to empty border with padding (10, 10, 10, 10)  // Create buttons  Create menuAddLecturerButton as a new JButton with text "Add Lecturer"  Add action listener to menuAddLecturerButton  Create menuAddTutorButton as a new JButton with text "Add Tutor"  Add action listener to menuAddTutorButton  Create menuGradeAssignmentButton as a new JButton with text "Grade \_\_\_\_\_\_\_\_\_\_\_Assignment"  Add action listener to menuGradeAssignmentButton  Create menuSetSalaryButton as a new JButton with text "Set Salary"  Add action listener to menuSetSalaryButton  Create nowRemoveTutor as a new JButton with text "Remove Tutor"  Add action listener to nowRemoveTutor  Create nowDisplay as a new JButton with text "Display"  Add action listener to nowDisplay  // Apply styling to buttons  Apply button styles to menuAddLecturerButton, menuAddTutorButton, \_\_\_\_\_\_\_\_\_\_\_menuGradeAssignmentButton,  menuSetSalaryButton, nowRemoveTutor, and nowDisplay  // Add buttons to buttonPanel  Add menuAddLecturerButton to buttonPanel  Add menuAddTutorButton to buttonPanel  Add menuGradeAssignmentButton to buttonPanel  Add menuSetSalaryButton to buttonPanel  Add nowRemoveTutor to buttonPanel  Add nowDisplay to buttonPanel  // Create right panel for description  Create descriptionPanel as a new JPanel with BorderLayout  Set background color of descriptionPanel to a dark blue color  Set border of descriptionPanel to empty border with padding (10, 10, 10, \_\_\_\_\_\_\_\_\_\_\_10)  // Create labels for button descriptions using HTML content  Create descriptionLabel as a new JLabel with HTML content provided  // Add description label to description panel  Add descriptionLabel to descriptionPanel at BorderLayout.CENTER  // Add panels to mainMenuPanel  Add topicPanel to mainMenuPanel at BorderLayout.NORTH  Add buttonPanel to mainMenuPanel at BorderLayout.EAST  Add descriptionPanel to mainMenuPanel at BorderLayout.CENTER  Return mainMenuPanel  Method createAddLecturerSection():  Create lecturerSectionPanel as a new JPanel  Set layout of lecturerSectionPanel to BorderLayout  // Left panel for adding lecturer details  Create lecturerSectionPanelLeft as a new JPanel  Set preferred size of lecturerSectionPanelLeft to (500, 700)  Set border of lecturerSectionPanelLeft to "Add Lecturer Section"  Set layout of lecturerSectionPanelLeft to GridBagLayout  Set background color of lecturerSectionPanelLeft to a light gray color  // Right panel for message box  Create lecturerSectionPanelRight as a new JPanel  Set background color of lecturerSectionPanelRight to a light gray color  Set layout of lecturerSectionPanelRight to BorderLayout  Create GridBagConstraints gbc  Set anchor of gbc to WEST  Set gridx of gbc to 0  Set gridy of gbc to 0  Set insets of gbc to (5, 5, 5, 5) // Set insets for spacing  // Add JLabels and JTextFields for each input field to the left panel  Create and add JLabels and JTextFields for teacher ID, name, address, \_\_\_\_\_\_\_\_\_\_\_working type,  employment status, department, graded score, and years of experience to  lecturerSectionPanelLeft using gbc for positioning  // Add "Add Lecturer" button to the left panel  Create addLecturer as a new JButton with text "Add Lecturer"  Set background color of addLecturer to dark blue  Set foreground color of addLecturer to white  Set font of addLecturer to bold Montserrat with size 15  Add action listener to addLecturer  Add addLecturer to lecturerSectionPanelLeft using gbc for positioning  // Add "Main Menu" button to the left panel  Create mainMenuBtnForLecturer as a new JButton with text "Main Menu"  Set background color of mainMenuBtnForLecturer to orange  Set foreground color of mainMenuBtnForLecturer to white  Set font of mainMenuBtnForLecturer to bold Montserrat with size 15  Add action listener to mainMenuBtnForLecturer  Add mainMenuBtnForLecturer to lecturerSectionPanelLeft using gbc for \_\_\_\_\_\_\_\_\_\_\_positioning  // Add "Clear" button to the left panel  Create clearLecturer as a new JButton with text "Clear"  Set background color of clearLecturer to red  Set foreground color of clearLecturer to white  Set font of clearLecturer to bold Montserrat with size 15  Add action listener to clearLecturer  Add clearLecturer to lecturerSectionPanelLeft using gbc for positioning  // Add message box to the right panel  Create labelTopicLecturer as a new JLabel with text "Lecturer Section: \_\_\_\_\_\_\_\_\_\_\_😊"  Set foreground color of labelTopicLecturer to dark blue  Set font of labelTopicLecturer to bold Montserrat with size 30  Set horizontal alignment of labelTopicLecturer to CENTER  Set border of labelTopicLecturer to empty border with padding (105, 0, 0)  Add labelTopicLecturer to lecturerSectionPanelRight at \_\_\_\_\_\_\_\_\_\_\_BorderLayout.NORTH  Add createMessageBoxForLecturer() to lecturerSectionPanelRight at \_\_\_\_\_\_\_\_\_\_\_BorderLayout.SOUTH  // Add left and right panels to the main lecturerSectionPanel  Add lecturerSectionPanelLeft to lecturerSectionPanel at \_\_\_\_\_\_\_\_\_\_\_BorderLayout.WEST  Add lecturerSectionPanelRight to lecturerSectionPanel at \_\_\_\_\_\_\_\_\_\_\_BorderLayout.EAST  Return lecturerSectionPanel    Method createAddTutorSection():  Create tutorSectionPanel as a new JPanel  Set border of tutorSectionPanel to "Add Tutor Section"  Set layout of tutorSectionPanel to BorderLayout  // Left panel for adding tutor details  Create tutorSectionPanelLeft as a new JPanel  Set preferred size of tutorSectionPanelLeft to (500, 700)  Set layout of tutorSectionPanelLeft to GridBagLayout  Set background color of tutorSectionPanelLeft to a light gray color  // Right panel for message box  Create tutorSectionPanelRight as a new JPanel  Set background color of tutorSectionPanelRight to a light gray color  Set layout of tutorSectionPanelRight to BorderLayout  Create GridBagConstraints gbc  Set anchor of gbc to WEST  Set gridx of gbc to 0  Set gridy of gbc to 0  Set insets of gbc to (5, 5, 5, 5) // Set insets for spacing  // Add JLabels and JTextFields for each input field to the left panel  Create and add JLabels and JTextFields for teacher ID, name, address,  working type, employment status, working hours, salary, specialization,  academic qualifications, and performance index to tutorSectionPanelLeft  using gbc for positioning  // Add "Add Tutor" button to the left panel  Create addTutor as a new JButton with text "Add Tutor"  Set background color of addTutor to dark blue  Set foreground color of addTutor to white  Set font of addTutor to bold Montserrat with size 15  Add action listener to addTutor  Add addTutor to tutorSectionPanelLeft using gbc for positioning  // Add "Main Menu" button to the left panel  Create mainMenuBtnForTutor as a new JButton with text "Main Menu"  Set background color of mainMenuBtnForTutor to orange  Set foreground color of mainMenuBtnForTutor to white  Set font of mainMenuBtnForTutor to bold Montserrat with size 15  Add action listener to mainMenuBtnForTutor  Add mainMenuBtnForTutor to tutorSectionPanelLeft using gbc for \_\_\_\_\_\_\_\_\_\_\_positioning  // Add "Clear" button to the left panel  Create clearTutor as a new JButton with text "Clear"  Set background color of clearTutor to red  Set foreground color of clearTutor to white  Set font of clearTutor to bold Montserrat with size 15  Add action listener to clearTutor  Add clearTutor to tutorSectionPanelLeft using gbc for positioning  // Add message box to the right panel  Create labelTopicTutor as a new JLabel with text "Tutor Section: 😍"  Set foreground color of labelTopicTutor to dark blue  Set font of labelTopicTutor to bold Montserrat with size 30  Set horizontal alignment of labelTopicTutor to CENTER  Set border of labelTopicTutor to empty border with padding (105, 0, 0, 0)  Add labelTopicTutor to tutorSectionPanelRight at BorderLayout.NORTH  Add createMessageBoxForTutor() to tutorSectionPanelRight at \_\_\_\_\_\_\_\_\_\_ BorderLayout.SOUTH  // Add left and right panels to the main tutorSectionPanel  Add tutorSectionPanelLeft to tutorSectionPanel at BorderLayout.WEST  Add tutorSectionPanelRight to tutorSectionPanel at BorderLayout.EAST  Return tutorSectionPanel      Method createGradeAssignmentsSection():  Create gradeAssignmentsSectionPanel as a new JPanel  Set border of gradeAssignmentsSectionPanel to "Grade Assignments \_\_\_\_\_\_\_\_\_\_ Section"  Set layout of gradeAssignmentsSectionPanel to BorderLayout  Set background color of gradeAssignmentsSectionPanel to white  // Left panel for adding assignment details  Create gradeAssignmentsSectionPanelLeft as a new JPanel  Set layout of gradeAssignmentsSectionPanelLeft to GridBagLayout  Set background color of gradeAssignmentsSectionPanelLeft to a light gray \_\_\_\_\_\_\_\_\_\_\_color  // Right panel for message box  Create gradeAssignmentsSectionPanelRight as a new JPanel  Set background color of gradeAssignmentsSectionPanelRight to a light \_\_\_\_\_\_\_\_\_\_\_gray color  Set layout of gradeAssignmentsSectionPanelRight to BorderLayout  Create GridBagConstraints gbc  Set anchor of gbc to WEST  Set gridx of gbc to 0  Set gridy of gbc to 0  Set insets of gbc to (5, 5, 5, 5) // Set insets for spacing  // Add JLabels and JTextFields for each input field to the left panel  Create and add JLabels and JTextFields for teacher ID, graded score, \_\_\_\_\_\_\_\_\_\_\_years of experience, and department to \_\_\_\_\_\_\_\_\_\_\_gradeAssignmentsSectionPanelLeft using gbc for positioning  // Add "Grade Assignments" button to the left panel  Create gradeAssignment as a new JButton with text "Grade Now"  Set background color of gradeAssignment to dark blue  Set foreground color of gradeAssignment to white  Set font of gradeAssignment to bold Montserrat with size 15  Add action listener to gradeAssignment  Add gradeAssignment to gradeAssignmentsSectionPanelLeft using gbc \_\_\_\_\_\_\_\_\_\_\_for positioning  // Add "Main Menu" button to the left panel  Create mainMenuBtnForGrade as a new JButton with text "Main Menu"  Set background color of mainMenuBtnForGrade to orange  Set foreground color of mainMenuBtnForGrade to white  Set font of mainMenuBtnForGrade to bold Montserrat with size 15  Add action listener to mainMenuBtnForGrade  Add mainMenuBtnForGrade to gradeAssignmentsSectionPanelLeft using \_\_\_\_\_\_\_\_\_\_ gbc for positioning  // Add "Clear" button to the left panel  Create clearAssignment as a new JButton with text "Clear"  Set background color of clearAssignment to red  Set foreground color of clearAssignment to white  Set font of clearAssignment to bold Montserrat with size 15  Add action listener to clearAssignment  Add clearAssignment to gradeAssignmentsSectionPanelLeft using gbc for \_\_\_\_\_\_\_\_\_\_\_positioning  // Add message box to the right panel  Create labelTopicAssignment as a new JLabel with text "Assignments \_\_\_\_\_\_\_\_\_\_ Section: 😎"  Set foreground color of labelTopicAssignment to dark blue  Set font of labelTopicAssignment to bold Montserrat with size 30  Set horizontal alignment of labelTopicAssignment to CENTER  Set border of labelTopicAssignment to empty border with padding (105, 0, \_\_\_\_\_\_\_\_\_\_\_0, 0)  Add labelTopicAssignment to gradeAssignmentsSectionPanelRight at \_\_\_\_\_\_\_\_\_\_\_BorderLayout.NORTH  Add createMessageBoxForAssignment() to \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_gradeAssignmentsSectionPanelRight at BorderLayout.SOUTH  // Add left and right panels to the main gradeAssignmentsSectionPanel  Add gradeAssignmentsSectionPanelLeft to \_\_\_\_\_\_\_\_\_\_\_gradeAssignmentsSectionPanel at BorderLayout.CENTER  Add gradeAssignmentsSectionPanelRight to \_\_\_\_\_\_\_\_\_\_\_gradeAssignmentsSectionPanel at BorderLayout.EAST  Return gradeAssignmentsSectionPanel // Return the panel        Method createSetSalaryOfTutorSection():  Create setSalaryOfTutorSectionPanel as a new JPanel  Set layout of setSalaryOfTutorSectionPanel to BorderLayout  Set border of setSalaryOfTutorSectionPanel to "Set Salary Section"  Set background color of setSalaryOfTutorSectionPanel to white  // Left panel for input fields, buttons, and remove tutor section  Create setSalaryOfTutorSectionPanelLeft as a new JPanel  Set preferred size of setSalaryOfTutorSectionPanelLeft to (500, 700)  Set layout of setSalaryOfTutorSectionPanelLeft to GridBagLayout  Set background color of setSalaryOfTutorSectionPanelLeft to a light gray \_\_\_\_\_\_\_\_\_\_\_color  Create GridBagConstraints gbc  // Add topic label for set salary section  Create and add JLabel topicLabelRemove to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning.  // Add JLabels and JTextFields for teacher ID, new performance index, \_\_\_\_\_\_\_\_\_\_\_and new salary  Create and add JLabels and JTextFields for input fields to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning.    Create and add JButton setSalaryOfTutor to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning.  Create and add JButton mainMenuBtnForSalay to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning  Create and add JLabel spaceLabel to setSalaryOfTutorSectionPanelLeft \_\_\_\_\_\_\_\_\_\_\_using gbc for positioning.  Create and add JLabel topicLabelSetSalary to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning.  Create and add JButton clearSetSalaryOfTutor to \_\_\_\_\_\_\_\_\_\_\_setSalaryOfTutorSectionPanelLeft using gbc for positioning.  Create removeTutorPanel as a new JPanel  Set layout of removeTutorPanel to GridBagLayout  Set border of removeTutorPanel to "Remove Tutor".  Set background color of removeTutorPanel to white.    Create and add JLabel removeTutorIdLabel and JTextField . . . . . \_\_\_\_\_\_\_\_\_\_ removeTutorIdTextField to removeTutorPanel using gbc for positioning  Create and add JButton removeTutor to removeTutorPanel using gbc for . / positioning.  Add removeTutorPanel to setSalaryOfTutorSectionPanelLeft using gbc for . positioning.  Create and add JButton displayButton to setSalarySectionPanelLeft using . gbc for positioning  Create setSalaryOfTutorSectionPanelRight as a new JPanel  Set background color of setSalaryOfTutorSectionPanelRight to a light gray . color.  Set layout of setSalaryOfTutorSectionPanelRight to BorderLayout  Create and add JLabel labelTopicSalary to setSalaryPanelRight at \_\_\_\_\_\_\_\_\_\_ BorderLayout.NORTH  Add createMessageBoxForSetSalary() to setSalaryRight at B.SOUTH  Add setSalaryOfTutorSectionPanelLeft to setSalaryOfTutorSectionPanel . . at BorderLayout.WEST  Add setSalaryOfTutorSectionPanelRight to setSalaryOfTutorSectionPanel . at BorderLayout.CENTER  Return setSalaryOfTutorSectionPanel // Return the panel  Method createDisplaySection():  Create mainDisplaySectionPanel as a new JPanel with BorderLayout  Set background color of mainDisplaySectionPanel to orange  Set border of mainDisplaySectionPanel to titled border with title .  Create displaySectionPanel as a new JPanel with GridLayout(2, 1)  Set background color of displaySectionPanel to white  Create displayLecturerPanel as a new JPanel with BorderLayout  Set background color of displayLecturerPanel to white  Set border of displayLecturerPanel to titled border with title "Lecturers"  Set background color of displayLecturerPanel to green  Create lecturerColumnNames as an array of column names  Create lecturerData as a 2D array of data for lecturer table  Create lecturerTable as a new JTable with lecturerData and lecColumn  Call customizeTable(lecturerTable) to customize table appearance  Create lecturerScrollPane as a new JScrollPane with lecturerTable  Add lecturerScrollPane to displayLecturerPanel at BorderLayout.CENTER  Create displayTutorPanel as a new JPanel with BorderLayout  Set background color of displayTutorPanel to white  Set border of displayTutorPanel to titled border with title "Tutors"  Set background color of displayTutorPanel to cyan  Create tutorColumnNames as an array of column names  Create tutorData as a 2D array of data for tutor table  Create tutorTable as a new JTable with tutorData and tutorColumnNames  Call customizeTable(tutorTable) to customize table appearance  Create tutorScrollPane as a new JScrollPane with tutorTable  Add tutorScrollPane to displayTutorPanel at BorderLayout.CENTER  Create displayBtnForDisplaying as a new JButton with text "Display"  Apply button style to displayBtnForDisplaying  Add action listener to displayBtnForDisplaying  Add displayLecturerPanel to displaySectionPanel  Add displayTutorPanel to displaySectionPanel  Add displayBtnForDisplaying to mainDisplaySectionPanel at BLayout.END  Add displaySectionPanel to mainDisplaySectionPanel  Return mainDisplaySectionPanel    Method customizeTable(table: JTable):  Set font of table to Montserrat, plain, size 13  Set font of table header to Montserrat, bold, size 13  Set background color of table header to gray  Set foreground color of table header to white  Set row height of table to 25  Set foreground color of table to black  Set background color of table to white  Set grid color of table to light gray  Set selection background color of table to orange  Set selection foreground color of table to white  Create a centerRenderer  Set horizontal alignment of centerRenderer to CENTER  Set default cell renderer of table to centerRenderer  Method createMessageBoxStyle(textArea: JTextArea):  Set background color of textArea to dark blue  Set foreground color of textArea to light purple  Set text of textArea to "We Show Message Here..... !"  Set editable property of textArea to false  Set font of textArea to Noto Sans, bold, size 17  Set line wrap of textArea to true  Set wrap style word of textArea to true  Return a new JScrollPane with textArea  Method createMessageBoxForLecturer():  Create messageForLecturer as a new JTextArea with size 15 rows and 32 . columns.  Set border of messageForLecturer to empty border with padding (7,1,3,2)  Return createMessageBoxStyle(messageForLecturer)  Method createMessageBoxForTutor():  Create messageForTutor as a new JTextArea with size 15 rows and 31 . . . columns.  Set border of messageForTutor to empty border with padding (7, 10, 3, 2)  Return createMessageBoxStyle(messageForTutor)  Method createMessageBoxForAssignment():  Create messageForAssignment as a new JTextArea with size 15 rows . .. … and 33 columns  Set border of messageForAssignment to empty border with padding (7, . . . 10, 3, 2)  Return createMessageBoxStyle(messageForAssignment)  Method createMessageBoxForSetSalary():  Create messageForSetSalary as a new JTextArea with size 16 rows and . / 37 columns  Set border of messageForSetSalary to empty border with padding ( 7,, . . . 3, 2)  Return createMessageBoxStyle(messageForSetSalary)  Method clearLecturerFields():  Set text of messageForLecturer to "Cleared successfully!"  Clear lecturerIdTextField, lecturerNameTextField, lecturerAddressTextFl,  lecturerWorkingTypeTextField, lecturerEmploymentStatusTextField, . . . . lecturerDepartmentTextField,  lecturerGradedScoreTextField, lecturerYearsOfExperienceTextField  Method clearTutorFields():  Set text of messageForTutor to "Cleared successfully!"  Clear text of tutorIdTextField, tutorNameTextField, tutorAddressTextField,  tutorWorkingTypeTextField, tutorEmploymentStatus, tutorWorkingField,  tutorSalaryTextField,..tutorSpecializationTextField, tutorAcademicField, . . . . . . tutorPerformanceIndexTextField  Method clearAssignmentFields():  Set text of messageForAssignment to "Cleared successfully!"  Clear assignmentTeacherIdTextField, assignmentGradedScoreTextField,  assignmentDepartmentTextField, assignmentYearsOfExperienceTextField  Method clearSetSalaryFields():  Set text of messageForSetSalary to "Cleared successfully!"  Clear text of salaryTutorIdTextField, salaryTutorNewSalaryTextField,  salaryTutorNewPerformanceIndexTextField, removeTutorIdTextField  Method actionPerformed(ActionEvent e):  If e.getSource() is addLecturer:  Call lecturerSectionFunctionality()  Else If e.getSource() is addTutor:  Call addTutor()  Else If e.getSource() is gradeAssignment:  Call gradeAssignment()  Else If e.getSource() is setSalaryOfTutor:  Call setSalaryOfTutor()  Else If e.getSource() is removeTutor:  Call removeTutor()  Else If e.getSource() is clearLecturer:  Call clearLecturerFields()  Else If e.getSource() is clearTutor:  Call clearTutorFields()  Else If e.getSource() is clearAssignment:  Call clearAssignmentFields()  Else If e.getSource() is clearSetSalaryOfTutor:  Call clearSetSalaryFields()  Else If e.getSource() is displayButton:  Call displayInfo()  Else If e.getSource() is preButton:  Call cardLayout.previous(mainPanel)  Else If e.getSource() is menuAddLecturerButton:  Call cardLayout.show(mainPanel, "addLecturer")  Else If e.getSource() is menuAddTutorButton:  Call cardLayout.show(mainPanel, "addTutor")  Else If e.getSource() is menuGradeAssignmentButton:  Call cardLayout.show(mainPanel, "gradeAssignments")  Else If e.getSource() is menuSetSalaryButton:  Call cardLayout.show(mainPanel, "setSalaryOfTutor")  Else If e.getSource() is nowRemoveTutor:  Call cardLayout.show(mainPanel, "setSalaryOfTutor")  Else If e.getSource() is mainMenuBtnForLecturer or \_\_\_\_\_\_\_\_\_\_\_mainMenuBtnForTutor or mainMenuBtnForGrade or \_\_\_\_\_\_\_\_\_\_\_mainMenuBtnForSalay or nextButton:  Call cardLayout.show(mainPanel, "main menu")  Else If e.getSource() is displayBtnForDisplaying:  Call displayInfoTable()  Else If e.getSource() is nowDisplay:  Call cardLayout.show(mainPanel, "displaySection")  method lecturerSectionFunctionality():  call addNewLecturer()  method addTutor():  call addNewTutor()  method gradeAssignment():  call newGradeAssignment()  method setSalaryOfTutor():  call newSetSalaryOfTutor()  method removeTutor():  call newRemoveTutor()  method displayInfo():  call lastlyDisplayInfo()  WHEN addNewLecturer is called  START TRY/CATCH  TRY  ASSIGN String name and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_lecturerNameTextField  ASSIGN String address and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_lecturerAddressTextField  ASSIGN String workingType and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_lecturerWorkingTypeTextField  ASSIGN String employmentStatus and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_lecturerEmploymentStatusTextField  ASSIGN String department and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_lecturerDepartmentTextField  ASSIGN int id, gradedScore, and yearsOfExperience  IF any required field is empty OR any string field contains numeric values  DISPLAY error message  RETURN false  END IF  ASSIGN id and return the integer value of text from lecturerIdTextField  ASSIGN gradedScore and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_lecturerGradedScoreTextField  ASSIGN yearsOfExperience and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_lecturerYearsOfExperienceTextField  FOR each teacher in addedTeachers arraylist  START IF  IF teacher.getTeacherId is equal to id  DISPLAY message "Teacher with ID " + id + " already registered."  RETURN  END IF  END LOOP  CREATE a new Lecturer object with id, name, address, workingType, \_\_\_\_\_\_\_\_\_\_\_employmentStatus, department, and yearsOfExperience  SET gradedScore of newLecturer to gradedScore  ADD newLecturer to addedTeachers arraylist  CREATE messageForLecturers string containing lecturer information  DISPLAY success message "Lecturer added successfully!: " + id  DISPLAY messageForLecturers  CATCH (NumberFormatException ex)  DISPLAY error message "Error: Please enter valid values for ID, Graded Score, \_\_\_\_\_and Years of Experience."  END TRY/CATCH      method validateAddLecturer(name, address, workingType, employmentStatus, \_\_\_\_\_department):  IF name.isEmpty() OR address.isEmpty() OR workingType.isEmpty() OR \_\_\_\_\_.employmentStatus.isEmpty() OR department.isEmpty():  SET messageForLecturer text to "Error: Please fill in all required fields."  RETURN false  END IF  IF containsNumeric(name) OR containsNumeric(address) OR \_\_\_\_\_\_\_\_\_containsNumeric(workingType) OR containsNumeric(employmentStatus) \_\_\_\_\_\_\_\_\_OR containsNumeric(department):  SET messageForLecturer text to "Error: String fields should not contain numeric \_\_\_\_values."  RETURN false  END IF  RETURN true  method containsNumeric(str):  FOR each character c in str:  IF c is a digit:  RETURN true  RETURN false  WHEN addNewTutor is called  START TRY/CATCH  TRY  ASSIGN String name and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_tutorNameTextField  ASSIGN String address and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_tutorAddressTextField  ASSIGN String workingType and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_tutorWorkingTypeTextField  ASSIGN String employmentStatus and return the trimmed string \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_from tutorEmploymentStatusTextField  ASSIGN String specialization and return the trimmed string from \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_tutorSpecializationTextField  ASSIGN String academicQualifications and return the trimmed \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_string from tutorAcademicQualificationsTextField  IF any required field is empty  DISPLAY error message  RETURN  END IF  ASSIGN int id, workingHours, and performanceIndex  ASSIGN double salary  IF !validateAddTutor(name, address, workingType, \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_employmentStatus, workingHours, specialization, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_academicQualifications)  RETURN  END IF  ASSIGN id and return the integer value of text from tutorIdTextField  ASSIGN workingHours and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_ tutorWorkingHoursTextField  ASSIGN salary and return the double value of text from \_\_\_\_\_\_\_\_\_\_\_ tutorSalaryTextField  ASSIGN performanceIndex and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_ tutorPerformanceIndexTextField  FOR each teacher in addedTeachers arraylist  START IF  IF teacher.getTeacherId is equal to id  DISPLAY message "Tutor with ID " + id + " already \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_registered."  RETURN  END IF  END LOOP  CREATE a new Tutor object with id, name, address, workingType, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_employmentStatus, workingHours, salary, specialization, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_academicQualifications, and performanceIndex  ADD newTutor to addedTeachers arraylist  CREATE messageForTutors string containing tutor information  DISPLAY success message "Tutor added successfully!: " + id  DISPLAY messageForTutors  CATCH (NumberFormatException ex)  DISPLAY error message "Error: Please enter valid values for ID, Working \_\_\_\_\_\_\_\_\_\_\_Hours, Salary, and Performance Index."  END TRY/CATCH  method validateAddTutor(name, address, workingType,  employmentStatus, workingHours,  specialization, academicQualifications):  IF name.isEmpty() OR address.isEmpty()  OR workingType.isEmpty() OR employmentStatus.isEmpty()  OR specialization.isEmpty() OR academicQualifications.isEmpty():  SET messageForTutor text to "Error: Please fill in all required \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_fields."  RETURN false  END IF  IF containsNumeric(name) OR containsNumeric(address)  OR containsNumeric(workingType) OR \_\_\_\_\_\_\_\_\_\_\_containsNumeric(employmentStatus)  OR containsNumeric(specialization) OR \_\_\_\_\_\_\_\_\_\_\_containsNumeric(academicQualifications):  SET messageForTutor text to "Error: String fields should not \_\_\_\_\_\_\_\_\_\_\_ contain numeric values."  RETURN false  END IF  IF workingHours < 20:  DISPLAY warning message "Your Working Hour is less than 20:\n \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Appraisal will not be applied later."  END IF  RETURN true  WHEN newGradeAssignment is called  START TRY/CATCH  TRY  ASSIGN int teacherId and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_assignmentTeacherIdTextField  ASSIGN int gradedScore and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_assignmentGradedScoreTextField  ASSIGN String department and return the string from \_\_\_\_\_\_\_\_\_\_\_assignmentDepartmentTextField  ASSIGN int yearsOfExperience and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_assignmentYearsOfExperienceTextField  CREATE message string containing assignment information  DISPLAY message  DISPLAY "Your Information is Processing"  ASSIGN boolean lecturerFound to false  FOR each teacher in addedTeachers arraylist  START IF  IF teacher is instance of Lecturer AND teacher.getTeacherId() is \_\_\_\_\_\_\_\_\_\_\_ equal to teacherId  ASSIGN Lecturer lecturer as teacher  START IF  IF yearsOfExperience is greater than or equal to 5 AND \_\_\_\_\_\_\_\_\_\_\_ lecturer.getDepartment() is equal to department AND \_\_\_\_\_\_\_\_\_\_\_ gradedScore is between 0 and 100  CALL lecturer.gradeAssignment(gradedScore, department, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_yearsOfExperience)  CALL lecturer.setGradedScore(gradedScore)  DISPLAY "Graded Successfully Teacher: " + teacherId  DISPLAY message  DISPLAY "Teacher with ID: " + teacherId + " has Graded"  ELSE  CALL lecturer.gradeAssignment(gradedScore, department, \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_yearsOfExperience)  DISPLAY "Grading failed."  DISPLAY "Check:"  DISPLAY "1. 5+ years exp. required."  DISPLAY "2. Dept. must match lecturer's."  DISPLAY "3. Graded score: 0-100."  END IF  ASSIGN lecturerFound to true  BREAK  END IF  END LOOP  START IF  IF ! lecturerFound  DISPLAY error message "No lecturer found with the provided ID: " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_+ teacherId  END IF  CATCH (NumberFormatException)  DISPLAY error message "Grading failed! Check: 1. 5+ years exp. required. 2. \_\_\_\_\_ Dept. must match lecturer's. 3. Graded score: 0-100."  END TRY/CATCH  WHEN newSetSalaryOfTutor is called  START TRY/CATCH  TRY  ASSIGN int tutorId and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_salaryTutorIdTextField  ASSIGN double newSalary and return the double value of text from \_\_\_\_\_\_\_\_\_\_\_salaryTutorNewSalaryTextField  ASSIGN int newPerformanceIndex and return the integer value of text \_\_\_\_\_\_\_\_\_\_\_from salaryTutorNewPerformanceIndexTextField  CREATE message string containing tutor information  DISPLAY message  DISPLAY "Your Information is Processing"  ASSIGN boolean tutorFound to false  FOR each teacher in addedTeachers arraylist  START IF  IF teacher is instance of Tutor AND teacher.getTeacherId() is equal \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_to tutorId  ASSIGN Tutor tutor as teacher  START IF  IF newPerformanceIndex is greater than or equal to 3 AND \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_tutor.getWorkingHours() is greater than 20  CALL tutor.setSalary(newSalary, newPerformanceIndex)  DISPLAY "Salary Set Successfully Teacher: " + tutorId  DISPLAY message  DISPLAY "Teacher with ID: " + tutorId + " has Set Salary"  RETURN  ELSE IF tutor.getWorkingHours() is less than or equal to 20  CALL tutor.setSalary(newSalary, newPerformanceIndex)  DISPLAY "Teacher: " + tutorId + " has Working Hour less \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_than 20 hrs."  END IF  ASSIGN tutorFound to true  BREAK  END IF  END LOOP  START IF  IF ! tutorFound  DISPLAY error message "No tutor found with the provided ID: " + \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ tutorId  END IF  CATCH (NumberFormatException)  DISPLAY error message "Salary Set failed. Check: 1. 3+ PerformanceIndex \_\_\_\_\_ required. 2. WorkingHours of Tutor must be > 20. 3. Check Numeric values"  END TRY/CATCH        WHEN newRemoveTutor is called  START TRY/CATCH  TRY  ASSIGN int tutorId and return the integer value of text from \_\_\_\_\_\_\_\_\_\_\_removeTutorIdTextField  FOR each teacher in addedTeachers arraylist  START IF  IF teacher is instance of Tutor AND teacher.getTeacherId() is equal \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_to tutorId  ASSIGN Tutor tutor as teacher  CALL tutor.removeTutor()  CALL addedTeachers.remove(teacher)  DISPLAY "Tutor with ID " + tutorId + " removed successfully."  RETURN  END LOOP  DISPLAY error message "No tutor found with the provided ID."  CATCH (NumberFormatException)  DISPLAY error message "Please enter a valid numeric value for Tutor ID."  END TRY/CATCH  WHEN displayInfoTable is called  ASSIGN lecturerData as a new Object array with dimensions \_\_\_\_\_\_\_\_\_\_\_ [addedTeachers.size()][8]  ASSIGN tutorData as a new Object array with dimensions \_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_ [addedTeachers.size()][10]  ASSIGN tutorColumnNames as an array containing column names for \_\_\_\_\_\_\_\_\_\_\_tutors  ASSIGN lecturerColumnNames as an array containing column names for \_\_\_\_\_\_\_\_\_\_\_lecturers  ASSIGN lecturerIndex to 0  ASSIGN tutorIndex to 0  FOR each teacher in addedTeachers  START IF  IF teacher is instance of Lecturer  ASSIGN lecturer as teacher  ASSIGN lecturerData[lecturerIndex] as a new Object \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_array containing lecturer information  INCREMENT lecturerIndex by 1  ELSE IF teacher is instance of Tutor  ASSIGN tutor as teacher  ASSIGN tutorData[tutorIndex] as a new Object array \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_containing tutor information  INCREMENT tutorIndex by 1  END IF  END LOOP  CREATE lecturerTableModel as a new DefaultTableModel with \_\_\_\_\_\_\_\_\_\_\_lecturerData and lecturerColumnNames  CREATE tutorTableModel as a new DefaultTableModel with tutorData and \_\_\_\_\_\_\_\_\_\_\_tutorColumnNames  SET lecturerTable model to lecturerTableModel  SET tutorTable model to tutorTableModel  WHEN lastlyDisplayInfo is called  CREATE a new StringBuilder info  FOR each teacher in addedTeachers  START IF  IF teacher is instance of Lecturer  ASSIGN lecturer as teacher  CALL teacher.display()  APPEND "Lecturer Information:\n" to info  APPEND "Teacher ID: " + lecturer.getTeacherId() + "\n" to info  APPEND "Teacher Name: " + lecturer.getTeacherName() + "\n" to info  APPEND "Address: " + lecturer.getAddress() + "\n" to info  APPEND "Working Type: " + lecturer.getWorkingType() + "\n" to info  APPEND "Employment Status: " + lecturer.getEmploymentStatus() + "\n" to \_\_\_\_\_\_\_\_info  APPEND "Department: " + lecturer.getDepartment() + "\n" to info  APPEND "Years of Experience: " + lecturer.getYearsOfExperience() + "\n" to \_\_\_\_\_\_\_\_info  APPEND "Graded Score: " + lecturer.getGradedScore() + "\n" to info  APPEND "\n" to info  ELSE IF teacher is instance of Tutor  ASSIGN tutor as teacher  CALL teacher.display()  APPEND "Tutor Information:\n" to info  APPEND "Teacher ID: " + tutor.getTeacherId() + "\n" to info  APPEND "Teacher Name: " + tutor.getTeacherName() + "\n" to info  APPEND "Address: " + tutor.getAddress() + "\n" to info  APPEND "Working Type: " + tutor.getWorkingType() + "\n" to info  APPEND "Employment Status: " + tutor.getEmploymentStatus() + "\n" to info  APPEND "Working Hours: " + tutor.getWorkingHours() + "\n" to info  APPEND "Salary: " + tutor.getSalary() + "\n" to info  APPEND "Specialization: " + tutor.getSpecialization() + "\n" to info  APPEND "Academic Qualifications: " + tutor.getAcademicQualifications() + \_\_\_\_\_\_\_\_"\n" to info  APPEND "Performance Index: " + tutor.getPerformanceIndex() + "\n" to info  APPEND "\n" to info  END IF  END LOOP  SET messageForSetSalary text to info.toString()  DISPLAY message dialog with info.toString() |

# 4. Method Description

**I. .Constructor TeacherGui()**

**II. Main method()**

# 5. Testing

## I. Test 1

## II. Test 2

### - Add Lecturer

### - Add Tutor

### - Grade Assignment

### - Set Salary

### - Remove Tutor

### - Display

## III. Test 3

# 6. Error Detection and Correction

## I. Syntax Error

## II. Runtime Error

## III. Logical Error

# 7. Conclusion

# 8. Bibliography

# 9. Appendix

}

//Main method that call constructor public static void main(String[] args) {

new StudentGui();

}

}