



**Module Code & Module Title CC4001NP Programming**

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

**Year and Semester 2023 Spring**

**Student’s Name: Bibek Bot Group: C6**

**College ID: NP04CP4S230019**

**London Met ID: 22085890 Assessment Due Date: 11th August 2023**

**Assessment Submission Date: 11th August 2023**

Table of Contents

1. [Introduction 1](#_bookmark0)
2. [Class Diagram 2](#_bookmark1)
   1. [Class Diagram Of Student Class 3](#_bookmark4)
   2. [Class Diagram Of Regular Class 3](#_bookmark5)
   3. [Class Diagram Of Dropout Class 4](#_bookmark6)
   4. [Class Diagram Of StudentGui Class 4](#_bookmark7)
3. [Pseudocode 5](#_bookmark8)
4. [Method Description 40](#_bookmark9)
   1. [Constructor StudentGui() 40](#_bookmark10)
   2. [Main method() 40](#_bookmark11)
5. [Testing 41](#_bookmark12)
   1. [Test 1 41](#_bookmark13)
   2. [Test 2 42](#_bookmark17)

[Add a Regular Student 42](#_bookmark18)

[Add a Dropout Student 44](#_bookmark21)

[Calculate Present Percentage of regular 45](#_bookmark25)

[Grant Certificate of Regular Student 47](#_bookmark27)

[Pays the bills of Dropout 49](#_bookmark30)

[Remove Dropout Student 51](#_bookmark33)

* 1. [Test 3 52](#_bookmark37)

1. [Error Detection and Correction 54](#_bookmark41)
   1. [Syntax Error 54](#_bookmark42)
   2. [Runtime Error 55](#_bookmark45)
   3. [Logical Error 56](#_bookmark49)
2. [Conclusion 59](#_bookmark54)

[Bibliography 60](#_bookmark55)

1. [Appendix 61](#_bookmark56)

**Table of Figure**

[**Figure 1: Class diagram of student, regular, dropout and studentGui Class 2**](#_bookmark2)

[**Figure 2: Empty class Diagram 2**](#_bookmark3)

[**Figure 3: Class diagram of student class 3**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656064)

[**Figure 4: Class diagram of regular class 3**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656065)

[**Figure 5: Class diagram of dropout class 4**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656066)

[**Figure 6: Class diagram of studentGui 4**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656067)

[**Figure 7: Writing command in cmd prompt 41**](#_bookmark15)

[**Figure 8: GUI run from cmd prompt 42**](#_bookmark16)

[**Figure 9: Fill the appropriate values 43**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656070)

[**Figure 10: After Clicking add button 43**](#_bookmark20)

[**Figure 11: Displaying the added infromation 44**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656072)

[**Figure 12: Fill the appropriate values 44**](#_bookmark22)

[**Figure 13: After clicking add to dropout button 45**](#_bookmark23)

[**Figure 14: Display add dropout student 45**](#_bookmark24)

[**Figure 15: Fill the enrollment Id and days present 46**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656076)

[**Figure 16: Outcomes of present percentage 46**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656077)

[**Figure 17: Filling the enrollment id , course name and date of enrollment 47**](#_bookmark29)

[**Figure 18: The outcomes of grant certificate 48**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656079)

[**Figure 19: Before checking bill pays 49**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656080)

[**Figure 20: Fill appropriate Enrollment id 49**](file://localhost/D:/bibek/Desktop/Java%20Documentation.docx%23_Toc142656081)

[**Figure 21: After checking bill pays 50**](#_bookmark32)

[**Figure 22: Fill the appropriate enrollmentId 51**](#_bookmark35)

[**Figure 23: After clicking button 51**](#_bookmark36)

[**Figure 24: Before Adding in arraylist 52**](#_bookmark39)

[**Figure 25: After trying to add in arraylist 53**](#_bookmark40)

[**Figure 26: Syntax error detection 54**](#_bookmark43)

[**Figure 27: Error Correction 54**](#_bookmark44)

[**Figure 28: Runtime Error 55**](#_bookmark46)

[**Figure 29: Runtime Error Detection 55**](#_bookmark47)

[**Figure 30: Runtime error correction 56**](#_bookmark48)

[**Figure 31: Logical error 56**](#_bookmark50)

[**Figure 32: Logical error detection 57**](#_bookmark51)

[**Figure 33: Logical error Correction 57**](#_bookmark52)

[**Figure 34: After Logical error correction 58**](#_bookmark53)

**List of Tables**

### [Table 1: Test 1 table 41](#_bookmark14)

### [Table 2: Add regular student table 42](#_bookmark19)

### [Table 3: Calculate present percentage case 45](#_bookmark26)

### [Table 4: Grant Certifiacte Case 47](#_bookmark28)

### [Table 5: Pay bill case 49](#_bookmark31)

### [Table 6: Remove case 51](#_bookmark34)

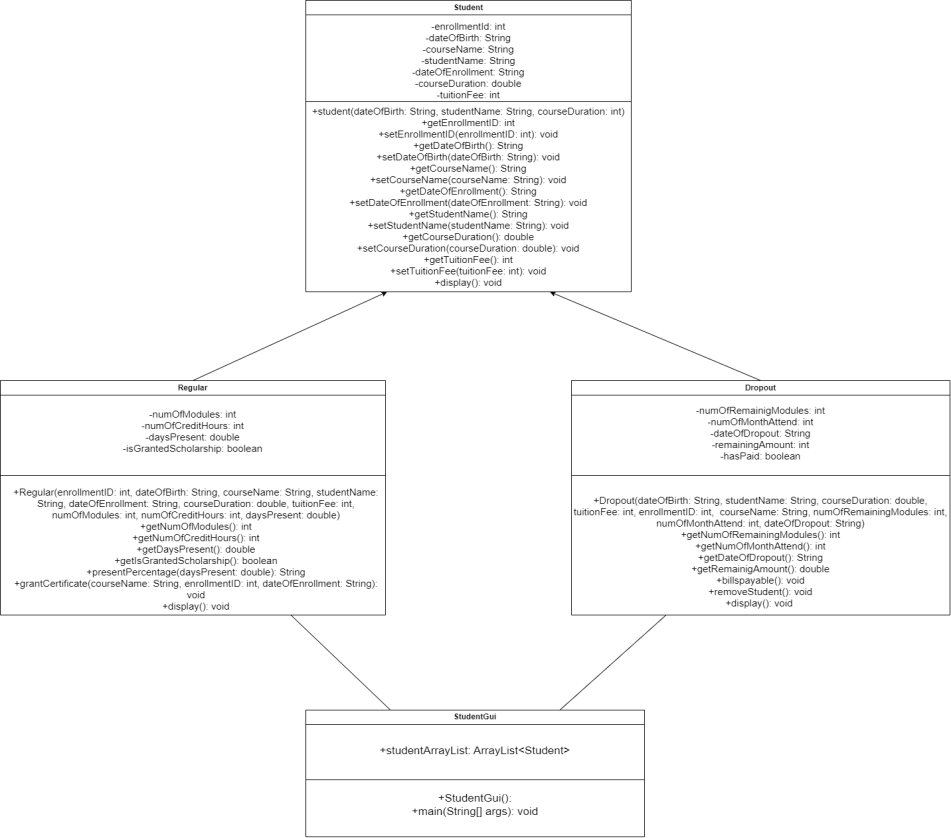
### [Table 7: Test 3 case 52](#_bookmark38)

# Introduction

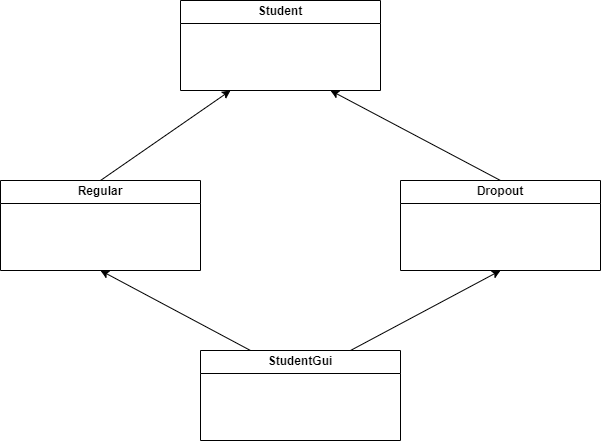
This coursework focuses on the development of graphical user interfaces (GUIs) and includes classes from the first part of a project that we have already completed, which includes student, regular, and dropout classes. A graphical user interface (GUI) is a user interface that uses icons or other visual indicators to interact with electronic devices rather than just text via a command line. For graphics programming, there are two sets of Java APIs: AWT (Abstract Windowing Toolkit) and Swing. We use Java Swing rather than AWT in the program because Swing is the most recent GUI Toolkit and provides a richer set of interface components than AWT while also being more lightweight. This program's main class is the StudentGui. The program saves Student's information in an ArrayLists. In the previous course, we used the BlueJ tool to create program, but in this course, we used the IntelliJ Idea tool.

Both programming and documentation are covered in this course. Through practice, the course's programming component enables students to apply Graphical User Interface (GUI), Java Swing, and ArrayList concepts. This program contents 1 constructer and 1 main method inside the constructer there are many buttons, textfield, label and each buttons and textfield has its own functionality. All the works like creating frame, label panel, textfield, buttons are done inside the constructor and the constructor is call from main method to display the GUI of this program.

# Class Diagram

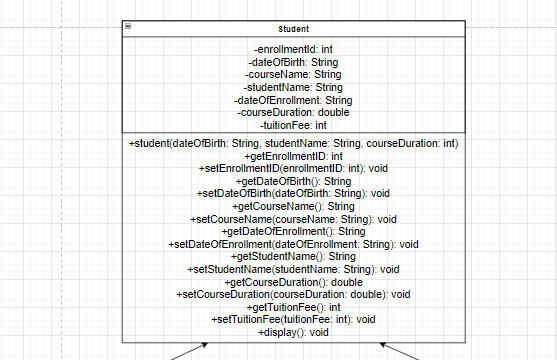


*Figure 1: Class diagram of student, regular, dropout and studentGui Class*



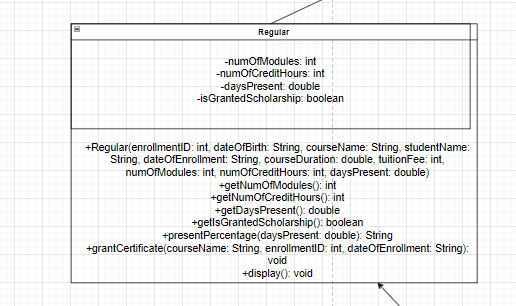
*Figure 2: Empty class Diagram*

## Class Diagram Of Student Class



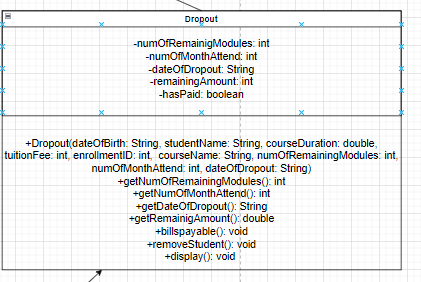
*Figure 3: Class diagram of student class*

## Class Diagram Of Regular Class



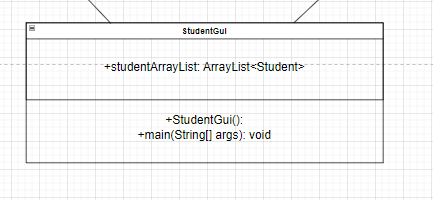
*Figure 4: Class diagram of regular class*

## Class Diagram Of Dropout Class



*Figure 5: Class diagram of dropout class*

## Class Diagram Of StudentGui Class



*Figure 6: Class diagram of studentGui*

# Pseudocode

CREATE class StudentGui

CREATE public ArrayList student with student type CREATE a constructor

CREATE Frame frame INITIALIZE GridBagLayout layout

INITIALIZE GridBagConstraints gbc CREATE Panel

CREATE Label studentName as "Student Name:"

SET Font "Serif" make BOLD and set size 15 for studentName CREATE TextField studentNameTf with columns 20

CREATE Label dateOfBirth as "Date Of Birth: "

SET Font "Serif" make BOLD and set size 15 for dateOfBirth CONSTRUCT array birthYears assign datatype String CONSTRUCT array birthMonths assign datatype String CONSTRUCT array birthDays assign datatype String

INITIALIZE birthYearsCombobox and add birthYears Array INITIALIZE birthMonthsCombobox and add birthMonths Array INITIALIZE birthDaysCombobox and add birthDays Array

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Label courseName as "Course Name:"

SET Font "Serif" make BOLD and set size 15 for courseName CREATE TextField courseNameTf with columns 20

CREATE Label courseDuration as "Course Duration:"

SET Font "Serif" make BOLD and set size 15 for courseDuration CREATE TextField courseDurationTf with columns 20

CREATE Label dateOfEnrollment as "Date Of Enrollment: "

SET Font "Serif" make BOLD and set size 15 for dateOfEnrollment Cons array enrollmentYears of datatype String

Construct array enrollmentMonths of datatype String Construct array enrollmentDays of datatype String

ITIALIZE enrollmentYearsCombobox and add enrollmentYears Array INTIALIZE enrollmentMonthsCombobox and add enrollmentMonths Array INITIALIZE enrollmentDaysCombobox and add enrollmentDays Array

CREATE Label tuitionFee as "Tuition Fee:"

SET Font "Serif" make BOLD and set size 15 for tuitionFee CREATE TextField tuitionFeeTf with columns 20

CREATE Label numberOfModules as "Number Of modules:"

SET Font "Serif" make BOLD and set size 15 for numberOfModules CREATE TextField numberOfModulesTf with columns 20

CREATE Label numberOfCreditHours "Number Of Credit Hours:"

SET Font "Serif" make BOLD and set size 15 for numberOfCreditHours Create TextField numberOfCreditHoursTf with columns 20

CREATE Label numberOfDaysPresent as "Number Of Days Present:" SET Font "Serif" make BOLD and set size 15 for numberOfDaysPresent CREATE TextField numberOfDaysPresentTf with columns 20

CREATE Label numberOfRemainingmodules as "Number Of Remaining Modules:"

SET Font "Serif" make BOLD and set size 15 for numberOfRemainingModules

CREATE TextField numberOfRemainingModulesTf with columns 20

CREATE Label numberOfMonthsAttend as "Number Of Months Attend:" SET Font "Serif" make BOLD and set size 15 for numberOfMonthsAttend CREATE TextField numberOfMonthsAttendTf with columns 20

CREATE JLabel dateOfDropout as "Date Of Dropout: "

SET Font "Serif" make BOLD and set size 15 for dateOfDropout CONSTRUCT array dropoutYears of datatype String CONSTRUCT array dropoutMonths of datatype String CONSTRUCT array dropoutDays of datatype String

INITIALIZE dropoutYearsCombobox and add dropoutYears Array INITIALIZE dropoutMonthsCombobox and add dropoutMonths Array INITIALIZE dropoutDaysCombobox and add dropoutDays Array

CREATE Label remainingAmount as "Remaining Amount:"

SET Font "Serif" make BOLD and set size 15 for remainingAmount CREATE TextField remainingAmountTf with columns 20

CREATE JButton addRegularStudentButton CREATE JButton addDropoutStudentButton CREATE JButton calculaePresentPercentButton Create JButton grantCertificateButton

Create JButton paybillsButton

Create JButton removeStudentButton Create JButton displayButton

CREATE JButtton clearButton SET Layout for panel

GIVE y coordinate 0 to studentNameLabel GIVE x coordinate 0 to studentNameLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for studentnameLabel ADD studentNameLabel in panel

GIVE x coordinate 1 to studentNameTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for studentnameTf ADD studentNameTf in panel

GIVE y coordinate 1 to dateOfBirthLabel GIVE x coordinate 0 to dateOfBirthLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for dateOfBirthLabel ADD dateOfBirthLabel in panel

GIVE x coordinate 1 to birthYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for birthYearsComboBox

ADD birthYearsComboBox in panel

GIVE x coordinate 2 to birthMonthsComboBox ADD birthMonthsComboBox in panel

GIVE x coordinate 3 to birthDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for birthDaysComboBox ADD birthDaysComboBox in panel

GIVE y coordinate 2 to enrollmentIdLabel

GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel ADD enrollmentIdLabel in panel

GIVE x coordinate 1 to enrollmentIdTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD enrollmentIdTf in panel

GIVE y coordinate 3 to courseNameLabel GIVE x coordinate 0 to courseNameLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for courseNameLabel ADD courseNameLabel in panel

GIVE x coordinate 1 to courseNameTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for courseNameTf ADD courseNameTf in panel

GIVE y coordinate 4 to courseDurationLabel GIVE x coordinate 0 to coursedurationLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for courseDurationLabel ADD courseDurationLabel in panel

GIVE gridx coordinate 1 to courseDurationTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD courseDurationTf in panel

GIVE y coordinate 5 to DateOfEnrollmentLabel GIVE x coordinate 0 to DateOfEnrollmentLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for DateOfEnrollmentLabel

ADD DateOfEnrollmentLabel in panel

GIVE x coordinate 1 to enrollmentYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for enrollmentYearsComboBox

ADD enrollmentYearsComboBox in panel

GIVE x coordinate 2 to enrollmentMonthsComboBox ADD enrollmentMonthsComboBox in panel

GIVE x coordinate 3 to enrollmentDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentDaysComboBox

ADD enrollmentDaysComboBox in panel

GIVE y coordinate 6 to tuitionFeeLabel GIVE x coordinate 0 to tuitionFeeLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for tuitionFeeLabel ADD tuittionFeeLabel in panel

GIVE x coordinate 1 to tuitionFeeTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for tuitionFeeTf ADD tuitionFeeTf in panel

GIVE gridy coordinate 0 to numberOfModulesLabel GIVE gridx coordinate 8 to numberOfmodulesLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfmodulesLabel

ADD numberOfModulesLabel in panel

GIVE x coordinate 9 to numberOfModulesTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfModulesTf

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ADD numberOfModulesTf in panel |  | | | |
| GIVE y coordinate 1 to numberOfCreditHoursLabel GIVE x coordinate 8 to numberOfCreditHoursLabel |
| SET insets 10 at top, 0 at left, 0 at bottom,  numberOfCreditHoursLabel | 0 | at | right | for |
| ADD numberOfCreditHoursLabel in panel |  |  |  |  |
| GIVE x coordinate 9 to numberOfCreditHoursTf |  |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| numberOfCreditHoursTf  ADD numberOfCreditHoursTf in panel |  |  |  |  |
| GIVE y coordinate 2 to numberOfDaysPresentLabel | |  |  |  |
| GIVE x coordinate 8 to numberOfDaysPresentHoursLabel  SET insets 10 at top, 0 at left, 0 at bottom, 0 | | at | right | for |
| numberOfDaysPresentHoursLabel | |  |  |  |
| ADD numberOfDaysPresentLabel in panel | |  |  |  |
| GIVE x coordinate 9 to numberOfDaysPresentTf | |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom, 0  numberOfDaysPresentTf | | at | right | for |
| ADD numberOfDaysPresentTf in panel | |  |  |  |
| GIVE y coordinate 3 to numberOfRemainingModulesLabel | |  |  |  |
| GIVE x coordinate 8 to numberOfRemainingModulesLabel | |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom, 0  numberOfRemainingModulesLabel | | at | right | for |
| ADD numberOfRemainingModules in panel | |  |  |  |
| GIVE x coordinate 9 to numberOfRemainingModulesTf | |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SET insets 10 at top, 0 at left, 0 at bottom, numberOfRemainingModulesTf  ADD numberOfRemaingModulesTf in panel | 0 | at | right | for |
| GIVE y coordinate 4 to numberOfMonthAttendLabel GIVE x coordinate 8 to numberOfMonthAttendLabel |  |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom,  numberOfMonthAttendLabel | 0 | at | right | for |
| ADD numberOfMonthAttendLabel in panel |  |  |  |  |
| GIVE x coordinate 9 to numberOfMonthAttendedTf  SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| numberOfMonthAttendedTf  ADDnumberOfMonthAttendedTf in panel |  |  |  |  |

GIVE y coordinate 5 to DateOfDropoutLabel GIVE x coordinate 8 to DateOfDropoutLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for DateOfDropoutLabel ADD DateOfDropoutLabel in panel

GIVE x coordinate 9 to dropoutYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for dropoutYearsComboBox

ADD dropoutYearsComboBox in panel

GIVE x coordinate 10 to dropoutMonthsComboBox ADD dropoutMonthsComboBox in panel

GIVE x coordinate 11 to dropoutDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for dropoutDaysComboBox

ADD dropoutDaysComboBox in panel

GIVE y coordinate 6 to remainingAmountLabel GIVE x coordinate 8 to remainingAmountLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for remainingAmountLabel

ADD remainingAmountLabel in panel

GIVE x coordinate 9 to remainingAmountTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for remainingAmountTf ADD remainingAmountTf in panel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GIVE y coordinate 7 addRegularStudentButton  GIVE x coordinate 1 to addRegularStudentButton |  |  |  |  |
| Fill addRegularStudentButton to Horizontal  SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| addRegularStudentButton |  |  |  |  |
| ADD addRegularStudentButton in panel |  |  |  |  |
| GIVE y coordinate 8 addDropoutStudentButton |  |  |  |  |
| GIVE x coordinate 1 to addDropoutStudentButton  SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| addDropoutStudentButton  ADD addDropoutStudentButton in panel |  |  |  |  |
| GIVE y coordinate 9 calculatePresentPercentButton |  |  |  |  |
| GIVE x coordinate 1 to calculatePresentPercentButton  SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| calculatePresentPercentButton  ADD calculatePresentPercentButton in panel |  |  |  |  |

GIVE y coordinate 10 displayButton GIVE x coordinate 1 to displayButton

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for displayButton ADD displayButton in panel

GIVE y coordinate 7 grantCertificateButton GIVE x coordinate 9 to grantCertificateButton

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for grantCertificateButton

ADD grantCertificateButton in panel

GIVE y coordinate 8 payBillsButton GIVE x coordinate 9 to payBillsButton

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for payBillsButton ADD payBillsButton in panel

GIVE y coordinate 9 removeStudentButton GIVE x coordinate 9 to removeStudentButton

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for removeStudentButton

ADD removeStudentButton in panel

GIVE y coordinate 10 clearButton GIVE x coordinate 9 to clearButton

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for clearButton ADD clearButton in panel

WHEN addRegularStudentButton is clicked CREATE Frame frame1

INITIALIZE GridBagLayout layout INITIALIZE GridBagConstraints gbc

CREATE Panel

CREATE Label studentName as "Student Name:"

SET Font "Serif" make BOLD and set size 15 for studentName CREATE TextField studentNameTf with columns 20

CREATE Label dateOfBirth as "Date Of Birth: "

SET Font "Serif" make BOLD and set size 15 for dateOfBirth CONSTRUCT array birthYears assign datatype String CONSTRUCT array birthMonths assign datatype String CONSTRUCT array birthDays assign datatype String

INITIALIZE birthYearsCombobox and add birthYears Array INITIALIZE birthMonthsCombobox and add birthMonths Array INITIALIZE birthDaysCombobox and add birthDays Array

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Label courseName as "Course Name:"

SET Font "Serif" make BOLD and set size 15 for courseName CREATE TextField courseNameTf with columns 20

CREATE Label courseDuration as "Course Duration:"

SET Font "Serif" make BOLD and set size 15 for courseDuration CREATE TextField courseDurationTf with columns 20

CREATE Label tuitionFee as "Tuition Fee:"

SET Font "Serif" make BOLD and set size 15 for tuitionFee CREATE TextField tuitionFeeTf with columns 20

CREATE Label numberOfModules as "Number Of modules:"

SET Font "Serif" make BOLD and set size 15 for numberOfModules CREATE TextField numberOfModulesTf with columns 20

CREATE Label numberOfCreditHours "Number Of Credit Hours:" SET Font "Serif" make BOLD and set size 15 for numberOfCreditHours

CREATE TextField numberOfCreditHoursTf with columns 20

CREATE Label numberOfDaysPresent as "Number Of Days Present:"

SET Font "Serif" make BOLD and set size 15 for numberOfDaysPresent

CREATE TextField numberOfDaysPresentTf with columns 20

CREATE Button addButton CREATE Button cancelButton

SET Layout for panel

GIVE y coordinate 0 to studentNameLabel GIVE x coordinate 0 to studentNameLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for studentnameLabel

ADD studentNameLabel in panel

GIVE x coordinate 1 to studentNameTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for studentnameTf

ADD studentNameTf in panel

GIVE y coordinate 1 to dateOfBirthLabel GIVE x coordinate 0 to dateOfBirthLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for dateOfBirthLabel

ADD dateOfBirthLabel in panel

GIVE x coordinate 1 to birthYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for birthYearsComboBox

ADD birthYearsComboBox in panel

GIVE x coordinate 2 to birthMonthsComboBox ADD birthMonthsComboBox in panel

GIVE x coordinate 3 to birthDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for birthDaysComboBox

ADD birthDaysComboBox in panel

GIVE y coordinate 2 to enrollmentIdLabel GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel

ADD enrollmentIdLabel in panel

GIVE x coordinate 1 to enrollmentIdTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD enrollmentIdTf in panel

GIVE y coordinate 3 to courseNameLabel

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GIVE x coordinate 0 to courseNameLabel  SET insets 10 at top, 0 at left, 0 at bottom, | 0 | at | right | for |
| courseNameLabel  ADD courseNameLabel in panel |  |  |  |  |
| GIVE x coordinate 1 to courseNameTf |  |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom,  courseNameTf | 0 | at | right | for |
| ADD courseNameTf in panel |  |  |  |  |
| GIVE y coordinate 4 to courseDurationLabel GIVE x coordinate 0 to coursedurationLabel |  |  |  |  |
| SET insets 10 at top, 0 at left, 0 at bottom,  courseDurationLabel | 0 | at | right | for |
| ADD courseDurationLabel in panel |  |  |  |  |

GIVE x coordinate 1 to courseDurationTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD courseDurationTf in panel

GIVE y coordinate 5 to DateOfEnrollmentLabel GIVE x coordinate 0 to DateOfEnrollmentLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for DateOfEnrollmentLabel

ADD DateOfEnrollmentLabel in panel

GIVE x coordinate 1 to enrollmentYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for enrollmentYearsComboBox

ADD enrollmentYearsComboBox in panel

GIVE x coordinate 2 to enrollmentMonthsComboBox ADD enrollmentMonthsComboBox in panel

GIVE x coordinate 3 to enrollmentDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentDaysComboBox

ADD enrollmentDaysComboBox in panel

GIVE y coordinate 6 to tuitionFeeLabel GIVE x coordinate 0 to tuitionFeeLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for tuitionFeeLabel

ADD tuittionFeeLabel in panel

GIVE x coordinate 1 to tuitionFeeTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for tuitionFeeTf ADD tuitionFeeTf in panel

GIVE y coordinate 7 to numberOfModulesLabel GIVE x coordinate 0 to numberOfmodulesLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfmodulesLabel

ADD numberOfModulesLabel in panel

GIVE x coordinate 1 to numberOfModulesTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfModulesTf

ADD numberOfModulesTf in panel

GIVE y coordinate 9 to numberOfCreditHoursLabel GIVE x coordinate 0 to numberOfCreditHoursLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfCreditHoursLabel

ADD numberOfCreditHoursLabel in panel

GIVE x coordinate 1 to numberOfCreditHoursTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfCreditHoursTf

ADD numberOfCreditHoursTf in panel

GIVE y coordinate 10 to numberOfDaysPresentLabel GIVE x coordinate 0 to numberOfDaysPresentHoursLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfDaysPresentHoursLabel

ADD numberOfDaysPresentLabel in panel

GIVE x coordinate 1 to numberOfDaysPresentTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfDaysPresentTf

ADD numberOfDaysPresentTf in panel

GIVE y coordinate 11 addButton GIVE x coordinate 1 to addButton FILL addButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for addButton ADD addButton in panel

GIVE y coordinate 13 cancelButton GIVE x coordinate 1 to cancelButton FILL cancelButton to Horizontal

SETinsets 10 at top, 0 at left, 0 at bottom, 0 at right for cancelButton ADD cancelButton in panel

WHEN addButton is clicked START IF

IF studentNameTf, birthYearsComboBox, birthMonthsComboBox, birthDaysComboBox , enrollmentIdTf, courseNameTf, courseDurationTf, enrollmentYearsComboBo, enrollmentMonthsComboBox, enrollmentDaysComboBox, tuitionFeeTf, numberOfModulesTf, numberOfCreditHours, numberOfDaysPresentTf is empty

DISPLAY popup message "Please fill the textfield"

ELSE

START TRY/CATCH TRY

ASSIGN String studentName and return the value of text from studentNameTf

ASSIGN dateOfBirth variable with String datatype store the selected date

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

ASSIGN String courseName and return the value of text from courseNameTf

ASSIGN courseduration variable with double datatype and return the double value of text from courseDurationTf

ASSIGN dateOfEnrollment variable with String datatype store the selected date

ASSIGN tuitionFee variable with int datatype and return the int value of text from tuitionFeeTf

ASSIGN numberOfModules variable with int datatype and return the in value of text from numberOfModulesTf

ASSIGN daysPresent variable with double datatype and return the double value of text from daysPresentTf

CREATE object of Regular Class ADD object in studentArrayLisT

DISPLAY popup message "Sucessfully Added" CATCH (Exception e1)

DISPLAY popup message "Please enter valid information"

END TRY/CATCH END IF

WHEN cancelButton is clicked SET frame1 visiblity to false

SET frame1 visiblity to f ADD panel in frame1 SET pack in frame1

SET dispose\_on\_close as the DefaultCloseOperation

WHEN dropoutStudentButton is clicked START IF

IF studentName, birthYearsComboBox, birthMonthsComboBox, birthDaysComboBox, enrollmentId, courseDuration, courseName, enrollmentYearsComboBox, enrollmentMonthsComboBox, enrollmentDaysComboBox, tuitionFee, numberOfRemainingModules, numberOfMonthAttended, dropoutYearsComboBox, dropoutMonthsComboBox, dropoutDaysComboBox is empty

DISPLAY popup message "Please fill the form

ELSE

START TRY/CATCH

TRY

ASSIGN String studentName and return the value of text from studentNameTf

ASSIGN dateOfBirth variable with String datatype store the selected date

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

ASSIGN String courseName and return the value of text from courseNameTf

ASSIGN courseduration variable with double datatype and return the double value of text from courseDurationTf ASSIGN dateOfEnrollment variable with String datatype store the selected date

ASSIGN tuitionFee variable with int datatype and return the int value of text from tuitionFeeTf

ASSIGN int numberOfRemainingModules and return the integer value of text from numberOfRemainingModulesTf ASSIGN numberOfMonthAttends variable with int datatype and return the int value of text from numberOfMonthAttendedTf

ASSIGN dateOfDropout variable with String datatype store the selected date

CREATE object of Dropout Class ADD object in studentArrayList

DISPLAY popup message "Sucessfully Added" CATCH (Exception e1)

DISPLAY popup message "Please enter valid information" END TRY/CATCH

END IF

WHEN calculatePresentPercentButton is clicked CREATE Frame frame2

INITIALIZE GridBagLayout layout INITIALIZE GridBagConstraints gbc CREATE Panel

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Label numberOfDaysPresent as "Number Of Days Present:" SET Font "Serif" make BOLD and set size 15 for numberOfDaysPresent CREATE TextField numberOfDaysPresentTf with columns 20

CREATE Button calculateButton CREATE Button cancelButton

SET layout for panel

GIVE y coordinate 0 to enrollmentIdLabel GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel ADD enrollmentIdLabel in panel

GIVE x coordinate 1 to enrollmentIdTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD enrollmentIdTf in panel

GIVE y coordinate 1 to numberOfDaysPresentLabel

GIVE x coordinate 0 to numberOfDaysPresentHoursLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfDaysPresentHoursLabel

ADD numberOfDaysPresentLabel in panel

GIVE x coordinate 1 to numberOfDaysPresentTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for numberOfDaysPresentTf

ADD numberOfDaysPresentTf in panel

GIVE y coordinate 2 calculateButton GIVE x coordinate 1 to calculateButton FILL calculateButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for calculateButton ADD calculateButton in panel

GIVE y coordinate 3 cancelButton GIVE x coordinate 1 to cancelButton FILL cancelButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for cancelButton ADD cancelButton in panel

WHEN calculateButton is clicked START IF

IF enrollmentIdTf and numberOfdaysPresentTf is empty DISPLAY popup message "Please fill the detail"

ELSE

START TRY/CATCH

TRY

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

ASSIGN daysPresent variable with double datatype and return the double value of text from daysPresentTf

DECLAER regular variable with Boolean datatype and set Boolean value to False

START FOR LOOP

FOR each student in studentArrayList arraylist START IF

IF student is instanceof Regular DOWNCAST student as Regular DECLEAR grade with string datatype

START NESTED IF

IF !grade is empty

DISPLAY popup message "Present percentage" +grade

ELSE

DISPLAY popup message "Invalid Input"

END NESTED IF

ASSIGN regular to true BREAK

END IF

START IF

IF ! regular

DISPLAY popup message "Enrollment Id doesn't exists"

CATCH(Exception e1)

DISPLAY popup message "Please enter valid number"

WHEN cancelButton is clicked SET frame2 visibilty to false

SET frame2 visibilty to true ADD panel to frame2

SET pack in frame2

SET dispose\_on\_close as the DefaultCloseOperation

WHEN grantCertificateButton is clicked CREATE Frame frame3 INITIALIZE GridBagLayout layout INITIALIZE GridBagConstraints gbc CREATE Panel

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Label courseName as "Course Name:"

SET Font "Serif" make BOLD and set size 15 for courseName CREATE TextField courseNameTf with columns 20

CREATE Label dateOfEnrollment as "Date Of Enrollment: "

SET Font "Serif" make BOLD and set size 15 for dateOfEnrollment CONSTRUCT array enrollmentYears of datatype String CONSTRUCT array enrollmentMonths of datatype String CONSTRUCT array enrollmentDays of datatype String

INITIALIZE enrollmentYearsCombobox and add enrollmentYears Array INITIALIZE enrollmentMonthsCombobox and add enrollmentMonths Array

INITILIZE enrollmentDaysCombobox and add enrollmentDays Array CREATE Button grantButton

CREATE Button cancelButton SET layout for panel

GIVE y coordinate 0 to enrollmentIdLabel GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel ADD enrollmentIdLabel in panel

GIVE x coordinate 1 to enrollmentIdTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdTf ADD enrollmentIdTf in panel

GIVE y coordinate 1 to courseNameLabel GIVE x coordinate 0 to courseNameLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for courseNameLabel ADD courseNameLabel in panel

GIVE x coordinate 1 to courseNameTf

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for courseNameTf ADD courseNameTf in panel

GIVE y coordinate 2 to DateOfEnrollmentLabel GIVE x coordinate 0 to DateOfEnrollmentLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for DateOfEnrollmentLabel

ADD DateOfEnrollmentLabel in panel

GIVE x coordinate 1 to enrollmentYearsComboBox

SET insets 10 at top, -130 at left, 0 at bottom, 0 at right for enrollmentYearsComboBox

ADD enrollmentYearsComboBox in panel

GIVE x coordinate 2 to enrollmentMonthsComboBox ADD enrollmentMonthsComboBox in panel

GIVE x coordinate 3 to enrollmentDaysComboBox

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentDaysComboBox

ADD enrollmentDaysComboBox in panel

GIVE y coordinate 3 grantButton GIVE x coordinate 1 to grantButton FILL grantButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for grantButton ADD grantButton in panel

GIVE y coordinate 4 cancelButton GIVE x coordinate 1 to cancelButton FILL cancelButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for cancelButton ADD cancelButton in panel

WHEN grantButton is clicked START IF

IF enrollmentIdTf, courseNameTf, enrollmentYearsComboBox, enrollmentMonthsComboBox, enrollmentDaysComboBox is empty

DISPLAY popup message "Invalid"

ELSE

START TRY/CATCH TRY

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

ASSIGN String courseName and return the value of text from courseNameTf

ASSIGN dateOfEnrollment variable with String datatype store the selected date

DECLAER scholarship variable with Boolean datatype and set Boolean value to False

START FOR LOOP

FOR each student in studentArrayList arraylist START IF

IF student is instanceof Regular DOWNCAST student as Regular

STORE courseName, enrollmentId, dateOfEnrollment in Regular class grantCertificate method

START NESTED IF

IF regulars isGrantedScholarship is true DISPLAY popup message "Scholarship has been granted"

ELSE

DISPLAY popup message "Scholarship is not granted"

END NESTED IF

SET scholarship Boolean value to true BREAK

END IF

END LOOP

START IF

IF ! scholarship

DISPLAY popup message "Enrollment ID not found"

END IF

CATCH(Exception e1)

DISPLAY popup message "Invalid"

WHEN cancelButton is clicked SET frame3 visibility to false

SET frame3 visibility to true ADD panel in frame3

SET pack in frame3

SET dispose\_on\_close as the DefaultCloseOperation

WHEN paysbillButton is clicked CREATE Frame frame4 INITIALIZE GridBagLayout layout

INITIALIZE GridBagConstraints gbc CREATE Panel

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Button paysButton CREATE Button cancelButton

SET layout for panel

GIVE y coordinate 0 to enrollmentIdLabel GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel ADD enrollmentIdLabel in panel

GIVE y coordinate 1 paysButton GIVE x coordinate 1 to paysButton FILL paysButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for paysButton ADD paysButton in panel

GIVE y coordinate 2 cancelButton GIVE x coordinate 1 to cancelButton FILL cancelButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for cancelButton ADD cancelButton in panel

WHEN paysButton is clicked START IF

IF enrollmentIdTf is empty

DISPLAY popup message "Please enter id to check"

ELSE

START TRY/CATCH TRY

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

DECLAER billpay variable with Boolean datatype and set Boolean value to False

START FOR LOOP

FOR each student in dropoutArrayList arraylist START IF

IF student is instanceof Dropout Class DOWNCAST student as Dropout

START NESTED IF

IF dropout class hasPaid is true DISPLAY popup message "Bill has been paid"

ELSE

DISPLAY popup message "Bill has to paid"

END NESTED IF

SET billpay Boolean to true BREAK

END IF END LOOP START IF

IF ! billpay

DISPLAY popup message " Given enrollment id not found"

END IF

CATCH (Exception e1)

DISPLAY popup message "Invalid"

END TRY/CATCH

WHEN cancelButton is clicked SET frame4 visibility to false

SET frame4 visibility to true ADD panel in frame4

SET pack in frame4

SET dispose\_on\_close as the DefaultCloseOperation

WHEN removeStudentButton is clicked CREATE Frame frame5 INITIALIZE GridBagLayout layout INITIALIZE GridBagConstraints gbc CREATE Panel

CREATE Label enrollmentId as "Enrollment Id:"

SET Font "Serif" make BOLD and set size 15 for enrollmentId CREATE TextField enrollmentIdTf with columns 20

CREATE Button removeButton CREATE Button cancelButton

SET layout for panel

GIVE y coordinate 0 to enrollmentIdLabel GIVE x coordinate 0 to enrollmentIdLabel

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for enrollmentIdLabel ADD enrollmentIdLabel in panel

GIVE y coordinate 1 removeButton

GIVE x coordinate 1 to removeButton FILL removeButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for removeButton ADD removeButton in panel

GIVE y coordinate 2 cancelButton GIVE x coordinate 1 to cancelButton FILL cancelButton to Horizontal

SET insets 10 at top, 0 at left, 0 at bottom, 0 at right for cancelButton ADD cancelButton in panel

WHEN removeButton is clicked START TRY/CATCH TRY

ASSIGN int enrollmentId and return the integer value of text from enrollmentIdTf

DECLAER remove variable with Boolean datatype and set Boolean value to False

START FOR LOOP

FOR each student in studentArrayList arraylist START IF

IF student is instanceof Dropout DOWNCAST student as dropout CALL removeStudent() of dropout START NESTED IF

IF dropout hasPaid is false

DISPLAY popup message "Dropout student with given enrollment id remove successfully"

ELSE

DISPLAY popup message "No need to remove"

END NESTED IF

SET remove Boolean value to true BREAK

END IF END LOOP START IF

IF ! remove

DISPLAY popup message "Dropout student with given enrollment id is not found"

END IF

CATCH (Exception e1)

DISPLAY popup message "Invalid input. Please enter valid information"

CREATE an addActionListener for the cancelButton SET visibility of frame5 to false

SET visibility of frame5 to true ADD panel to frame5

SET pack to frame5

SET dispose\_on\_close as the DefaultCloseOperation

WHEN displayButton is clicked START FOR LOOP

FOR each student in studentArrayList arryalist IF student is instanceof Regular

CREATE Frame frame8

DECLEAR regularHeader as String datatype Array

CONSTRUCT defaultTable model and add regularHeader and row value

DOWNCAST Regular as rStudent

DECLEAR enrollmentId with int datatype and return the value of getEnrollmentId method

DECLEAR courseName with String datatype and return the value of getCourseName method DECLEAR studentName with String datatype and return the value of getStudentName method DECLEAR dateOfBirth with String datatype and return the value of getDateOfBirth method

DECLEAR dateOfEnrollment with String datatype and return the value of getDateEnrollment method DECLEAR courseDuration with double datatype and return the value of getCourseDuration method DECLEAR tuitionFee with int datatype and return the value of getTuitionFee method

DECLEAR numOfModules with int datatype and return the value of getNumOfModules method

DECLEAR numOfCreditHours with int datatype and return the value of getNumOfCreditHours method DECLEAR daysPresent with double datatype and return the value of getDaysPresent method

ADD row in model of table and store object on row

CREATE table and add model in table CREATE scrollpane and add table in scrollpane

ADD scrollpane in frame8

SET height and width of frame8

SET dispose\_on\_close as the DefaultCloseOperation SET visibility of frame8 to true

IF student is instanceof Dropout CREATE Frame frame9

DECLEAR dropoutHeader as String datatype Array

CONSTRUCT defaultTable model and add dropoutHeader and row value

DOWNCAST Dropout as dStudent

DECLEAR enrollmentId with int datatype and return the value of getEnrollmentId method

DECLEAR courseName with String datatype and return the value of getCourseName method DECLEAR studentName with String datatype and return the value of getStudentName method DECLEAR dateOfBirth with String datatype and return the value of getDateOfBirth method

DECLEAR dateOfEnrollment with String datatype and return the value of getDateEnrollment method DECLEAR courseDuration with double datatype and return the value of getCourseDuration method DECLEAR tuitionFee with int datatype and return the value of getTuitionFee method

DECLEAR numOfRemainingModules with int datatype and return the value of getNumOfRemainingModules method

DECLEAR numOfMonthsAttended with int datatype and return the value of getNumMonthsAttended method

DECLEAR dateOfDropout with String datatype and return the value of getDateOfDropout

DECLEAR remainigAmount with int datatype and return the value of getRe9mainingAmount

ADD row in model of table and store object on row

CREATE table and add model in table CREATE scrollpane and add table in scrollpane

ADD scrollpane in frame9

SET height and width of frame9

SET dispose\_on\_close as the DefaultCloseOperation SET visibility of frame9 to true

END LOOP

WHEN clearButton is clicked SET all textfield to empty

SET all comboBox to index 0

ADD panel in frame

SET size 1500 width 800 height of frame SET visibility of frame to true

SET exit\_on\_close as the DefaultCloseOperation

CREATE METHOD

CALL the constructor

# Method Description

## Constructor StudentGui()

This method initialize panel, frame, textfield, label, button and combox. Inside this method the GUI is design using the all components that is initialize inside this method. This method contents one panel which holds all the components according to own designs. This method has 7 button actions. When the addRegularButton is clicked new frame of regular student is open and the provided information is store in arraylist. When addDropoutStudentButton is clicked the provided information of dropout student is store in arraylist. When calculatePresentPercentButton is clicked the new frame will open and the compare the enrollment id with its store enrollment id if the condition is fulfill the present percentage is calculate and other button are also add with seprate function. When display button is clicked the details which are added inside arraylist are shown in table format in new frame.

## Main method()

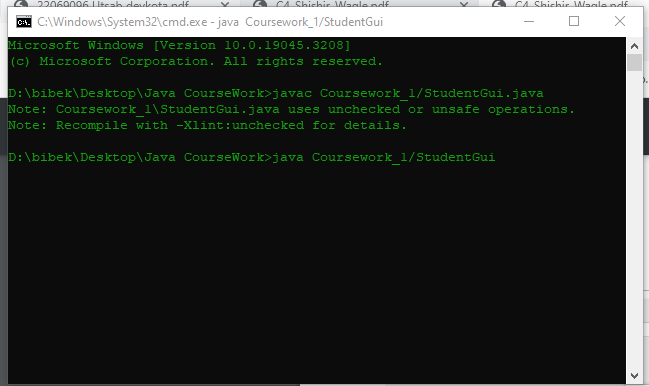
Call the Constructor StudentGui()

# Testing

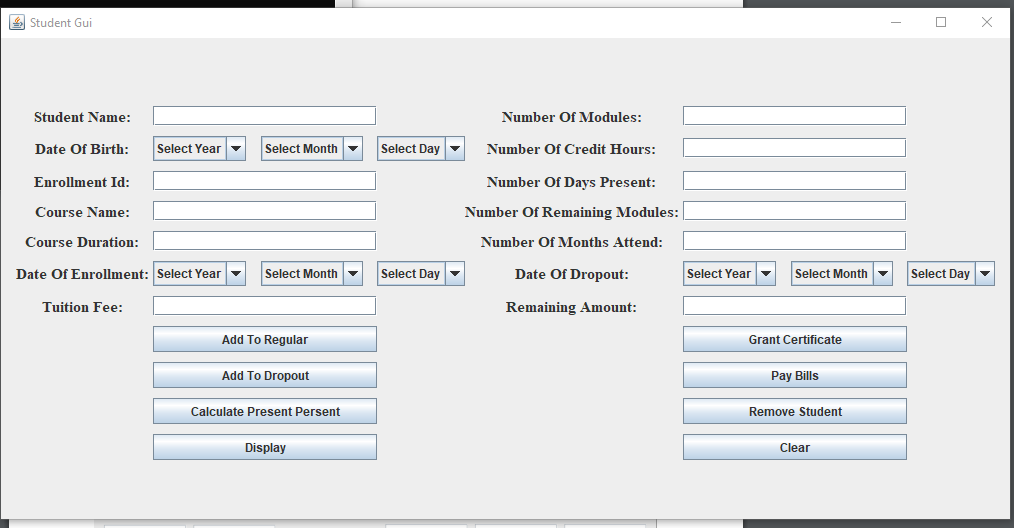
## Test 1

*Table 1: Test 1 table*

|  |  |
| --- | --- |
| Objectives | Run and compile the program using command prompt |
| Action | 1. First write the command in command prompt to compile with suitable path with program file name 2. Second write the command to run with file path and file name |
| Expected Outcomes | The frame with gui component should run with no error |
| Actual Outcomes | The frame with gui run properly |
| Conclusion | Test Sucessful |



*Figure 7: Writing command in cmd prompt*



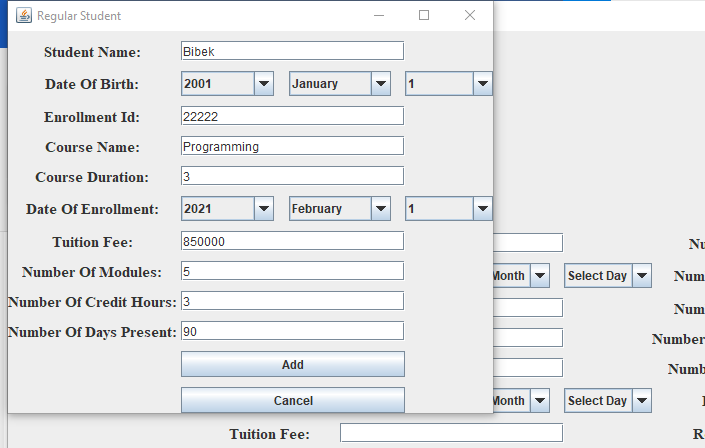
*Figure 8: GUI run from cmd prompt*

## Test 2

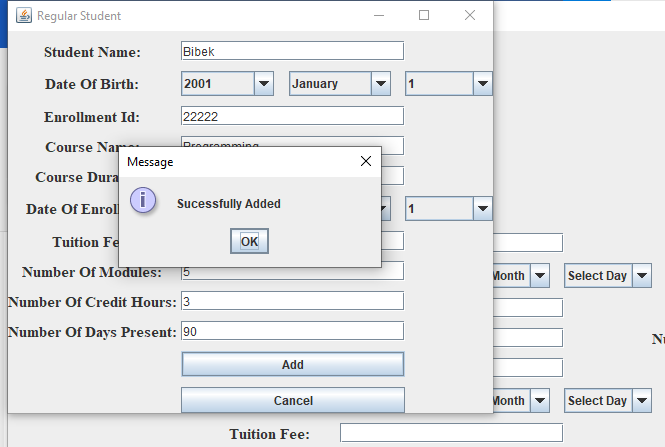
### Add a Regular Student

*Table 2: Add regular student table*

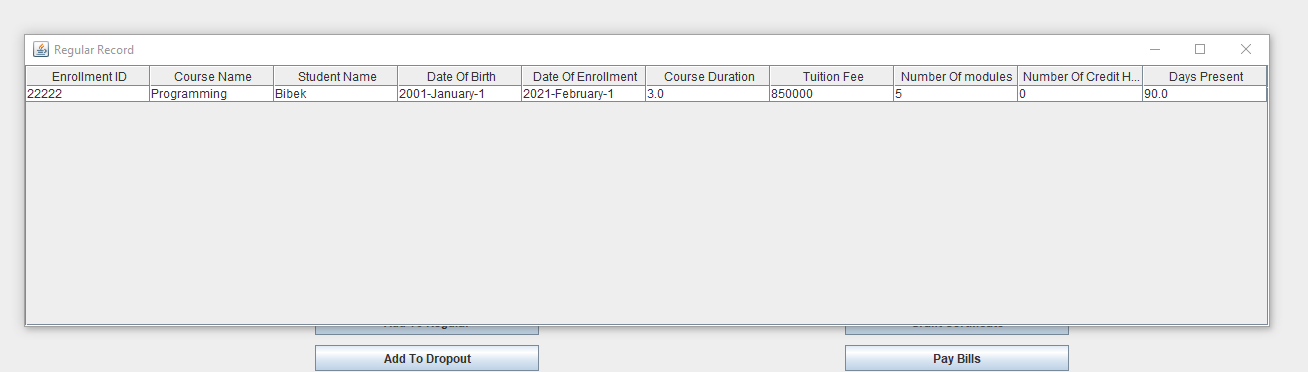
|  |  |
| --- | --- |
| Objectives | Check the add regular button work properly or not |
| Action | Fill the appropriate values Click add button |
| Expected Outcomes | The enter values should added in regular |
| Actual Outcomes | The enter value added in arraylist |
| Conclusion | Test Sucessful |



*Figure 9: Fill the appropriate values*



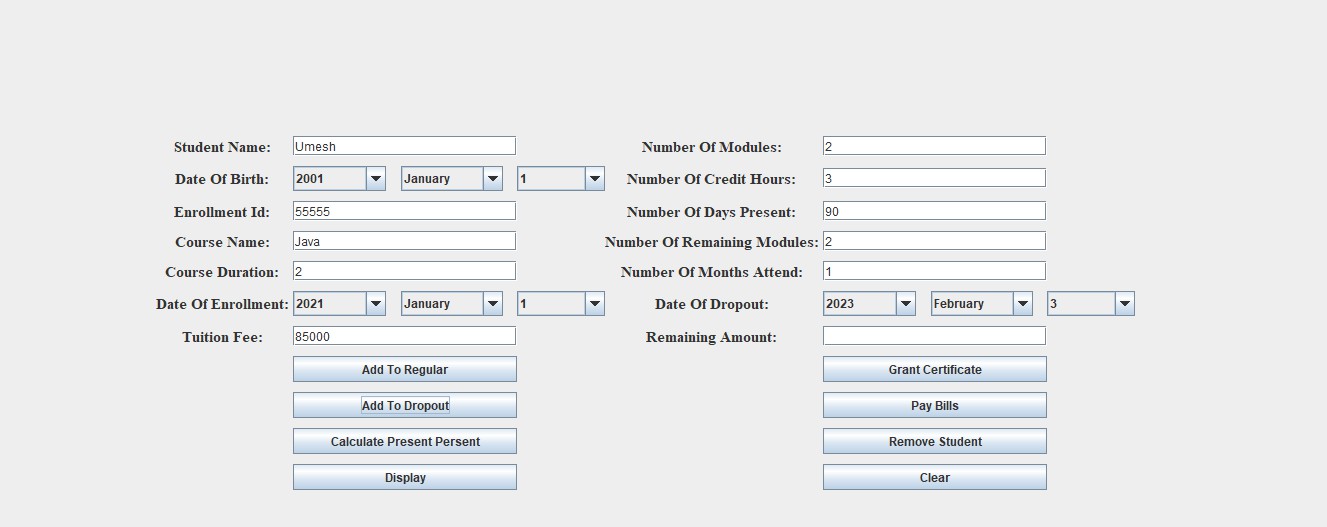
*Figure 10: After Clicking add button*



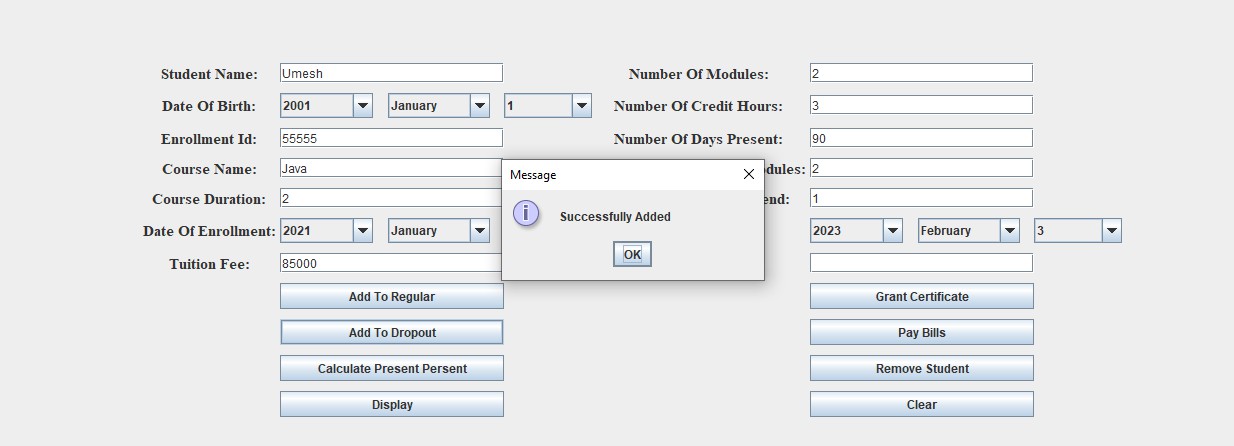
*Figure 11: Displaying the added infromation*

### Add a Dropout Student

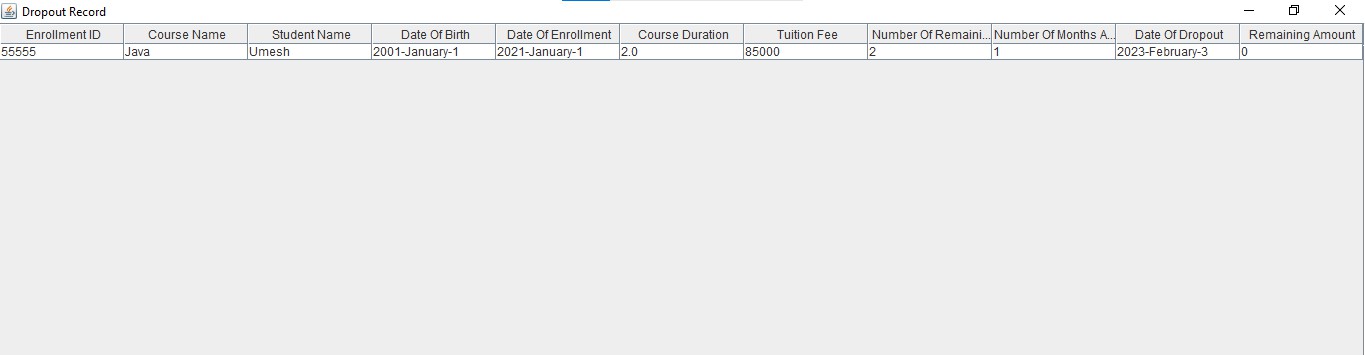
|  |  |
| --- | --- |
| Objectives | Check the dropout button work properly or not |
| Action | Fill the appropriate values Click add button |
| Expected Outcomes | The enter values should added in dropout |
| Actual Outcomes | The enter value added in arraylist |
| Conclusion | Test Sucessful |



*Figure 12: Fill the appropriate values*



*Figure 13: After clicking add to dropout button*

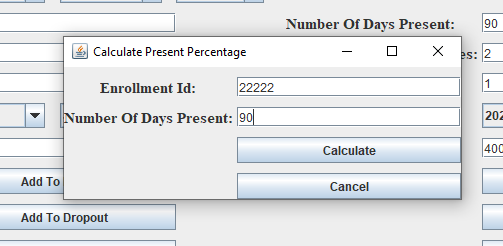


*Figure 14: Display add dropout student*

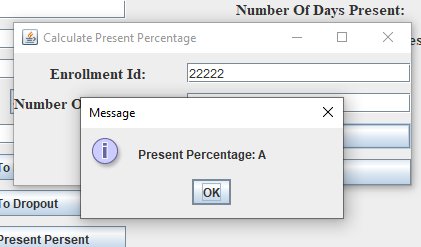
### Calculate Present Percentage of regular

*Table 3: Calculate present percentage case*

|  |  |
| --- | --- |
| Objectives | Calculate the present percentage of regular student |
| Action | Fill the appropriate enrollment id and days present |
| Expected Outcomes | The grade should be A |
| Actual Outcomes | The calculate grade is A |
| Conclusion | Test Sucessful |



*Figure 15: Fill the enrollment Id and days present*



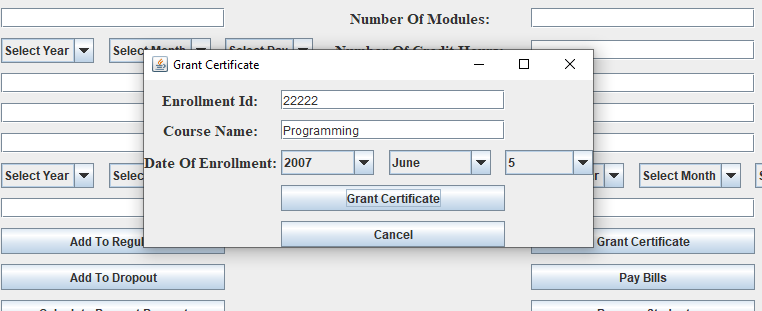
*Figure 16: Outcomes of present percentage*

The outcomes grade is A because the day present values is equals to the store days present values of regular student.

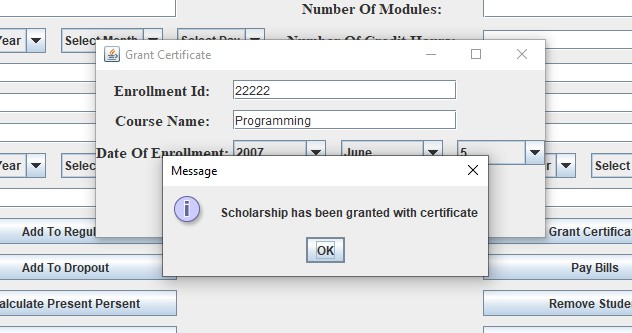
### Grant Certificate of Regular Student

*Table 4: Grant Certifiacte Case*

|  |  |
| --- | --- |
| Objectives | Check weather the scholarship and certificate is granted or not |
| Action | Fill the appropriate enrollment id, course name and date of enrollment |
| Expected Outcomes | The scholarship an certificate should grant |
| Actual Outcomes | The calculate grade is A |
| Conclusion | Test Sucessful |



*Figure 17: Filling the enrollment id , course name and date of enrollment*



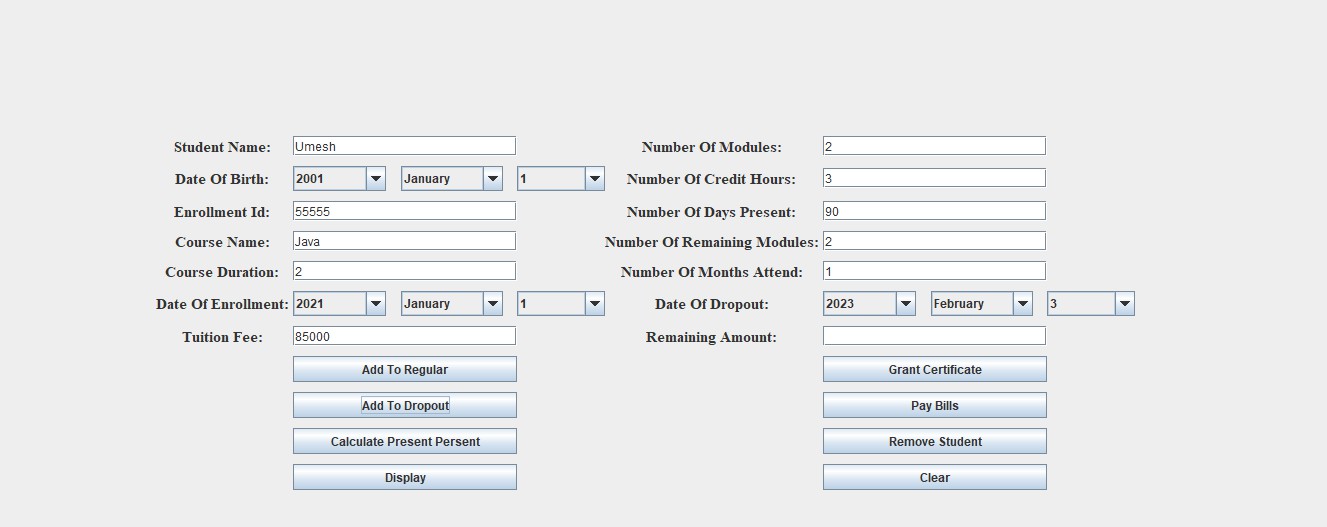
*Figure 18: The outcomes of grant certificate*

The scholarship and certificate is granted because the present percentage grade is A and in the program if the the days present is greater than 80 grant scholarship condition is written in regular class and in this test the condition of program is fulfill and the scholarship and certificate is granted

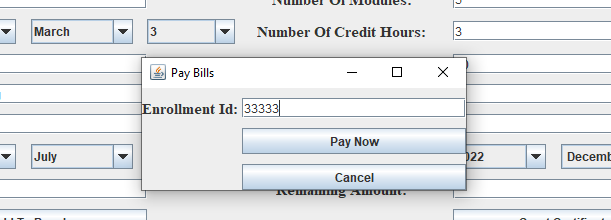
### Pays the bills of Dropout

*Table 5: Pay bill case*

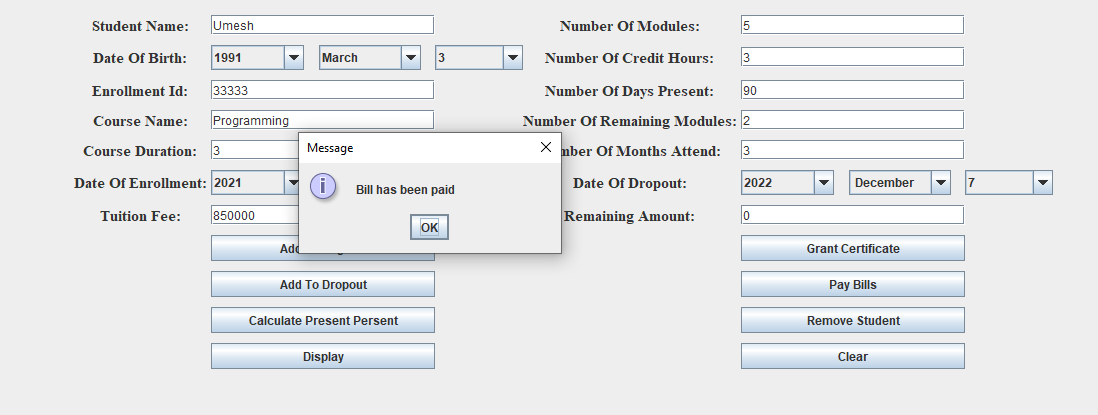
|  |  |
| --- | --- |
| Objectives | Check weather the bill is paid or not |
| Action | Fill the appropriate enrollment id |
| Expected Outcomes | The outcome of bill should be clear |
| Actual Outcomes | The bill has been paid |
| Conclusion | Test Sucessful |



*Figure 19: Before checking bill pays*



*Figure 20: Fill appropriate Enrollment id*



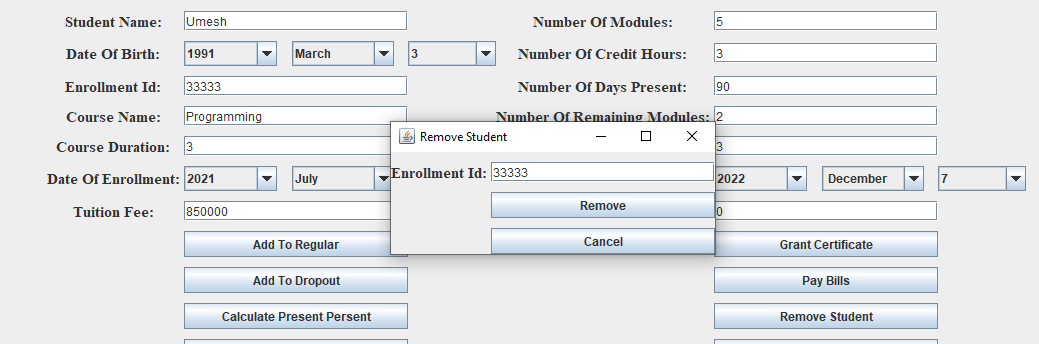
*Figure 21: After checking bill pays*

The outcome of this test is bill has been paid because the course duration and months attended equals and the remaining amount appear 0 in remaining amount textfield

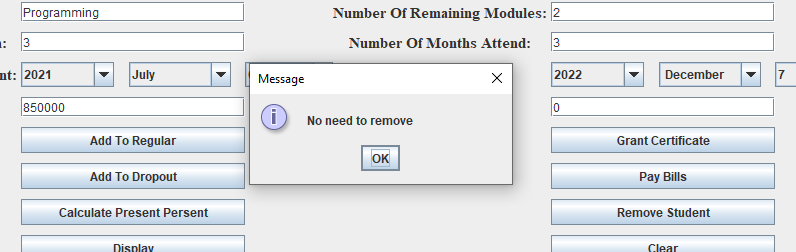
### Remove Dropout Student

*Table 6: Remove case*

|  |  |
| --- | --- |
| Objectives | Check weather the student need to remove or not |
| Action | Fill the appropriate enrollment id |
| Expected Outcomes | The outcome should no need to remove student |
| Actual Outcomes | No need to remove |
| Conclusion | Test Successful |



*Figure 22: Fill the appropriate enrollmentId*



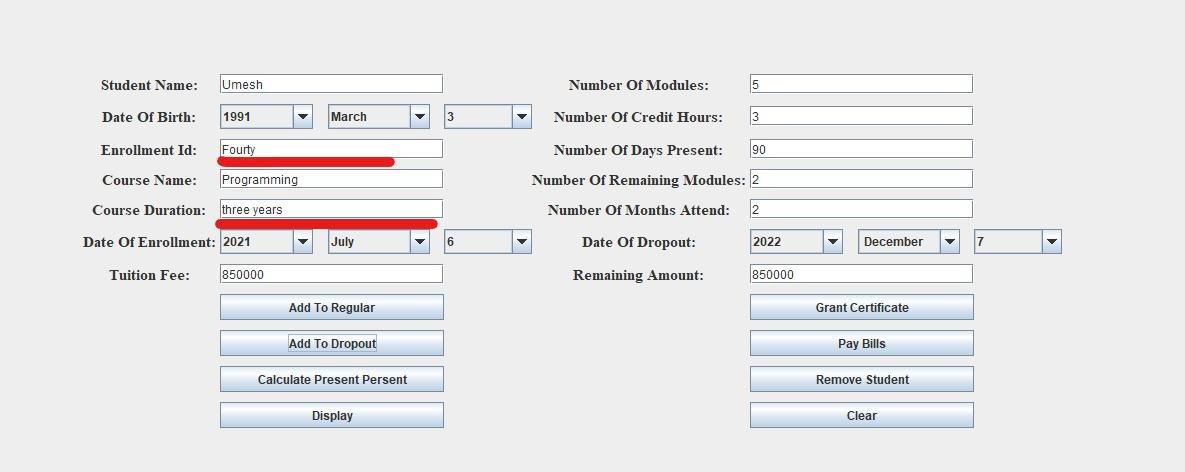
*Figure 23: After clicking button*

The outcome is no need to remove because the remaining amount of the student 0 and fullfill the condition of program.

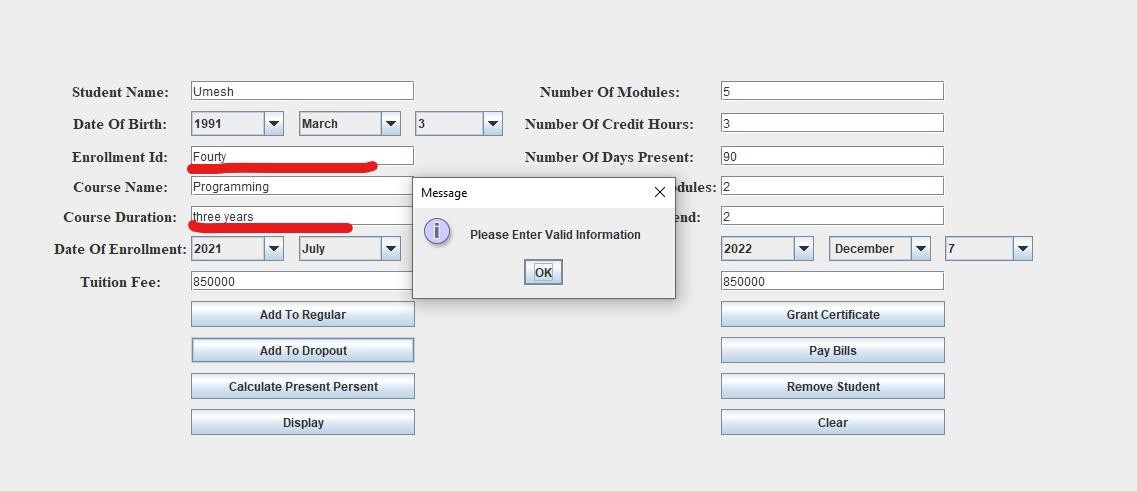
## Test 3

*Table 7: Test 3 case*

|  |  |
| --- | --- |
| Objectives | Check weather the user input text or Integer in textfield |
| Action | Fill the text in enrollment id and course duration |
| Expected Outcomes | The outcome should exception |
| Actual Outcomes | Show exception when trying to store in arraylist |
| Conclusion | Test Successful |



*Figure 24: Before Adding in arraylist*



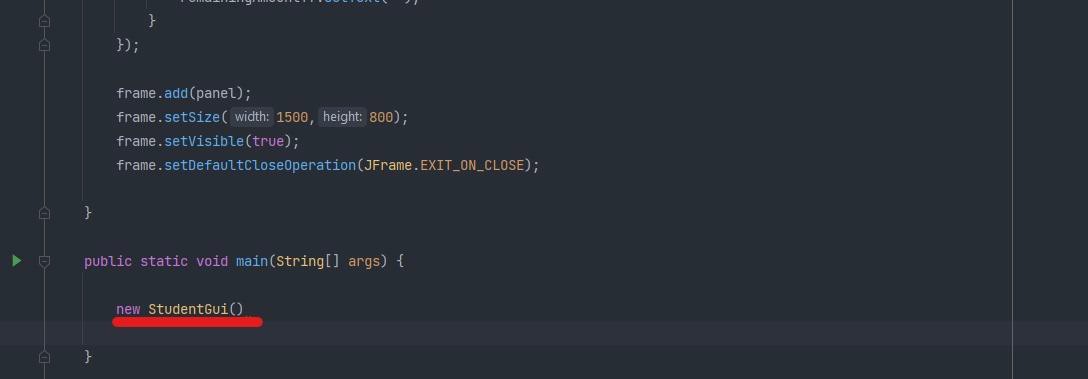
*Figure 25: After trying to add in arraylist*

The outcome show popup because in the field integer value I try to store String value and the try catch display the popup message.

# Error Detection and Correction

## Syntax Error

When a programme encounters syntactical issues as a result of improper syntax usage, syntax errors take place (Mueller, 2016). For instance, if a variable is used that hasn't been defined, if a sentence is finished without a semicolon, or if the punctuation is incorrect,



*Figure 26: Syntax error detection*

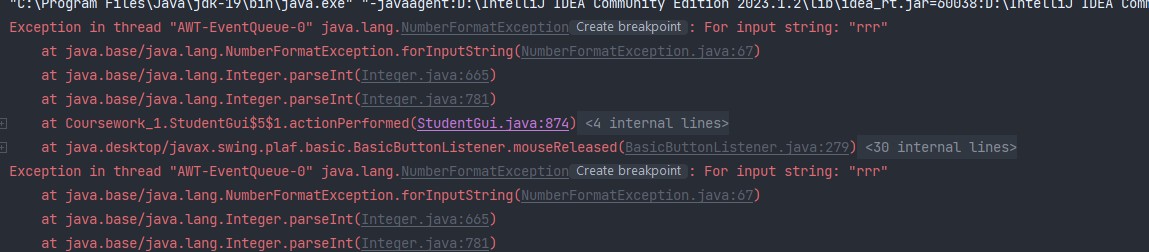


*Figure 27: Error Correction*

I forgot to insert a semicolon here, which resulted in a syntax error.

## Runtime Error

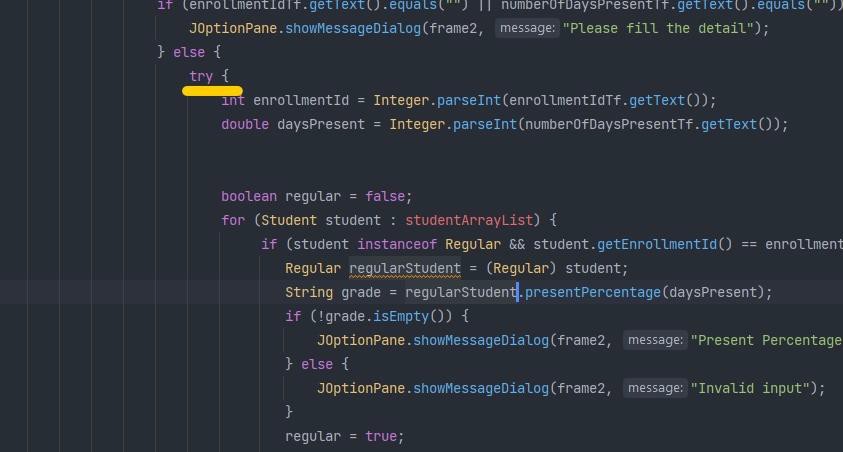
An error that manifests during a program's execution after being successfully compiled is known as a runtime error. (Bharti, 2020). There are numerous causes for this, including attempting to access an out-of-bounds array. The compiler fails to detect these errors.



*Figure 28: Runtime Error*



*Figure 29: Runtime Error Detection*

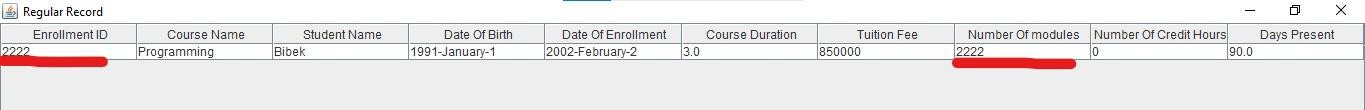


*Figure 30: Runtime error correction*

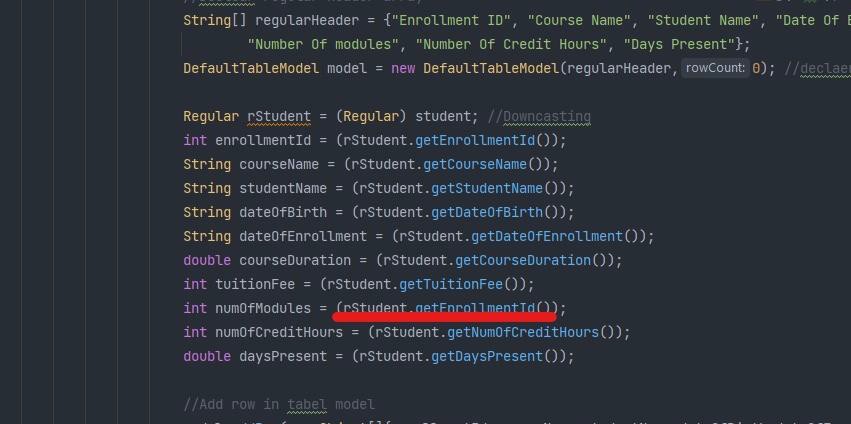
I forgot to implement code inside the Try/Catch which resulted in a runtime error.

## Logical Error

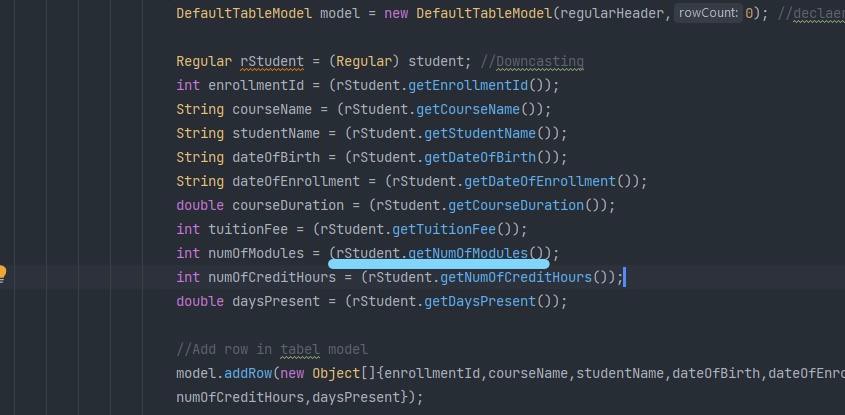
A logic error is a bug in a program's source code that causes unexpected and erroneous behaviour (Rouse, 2023). It does not cause the program to crash or generate any syntax errors. These are difficult to come by because the program will run but will not produce the desired results.



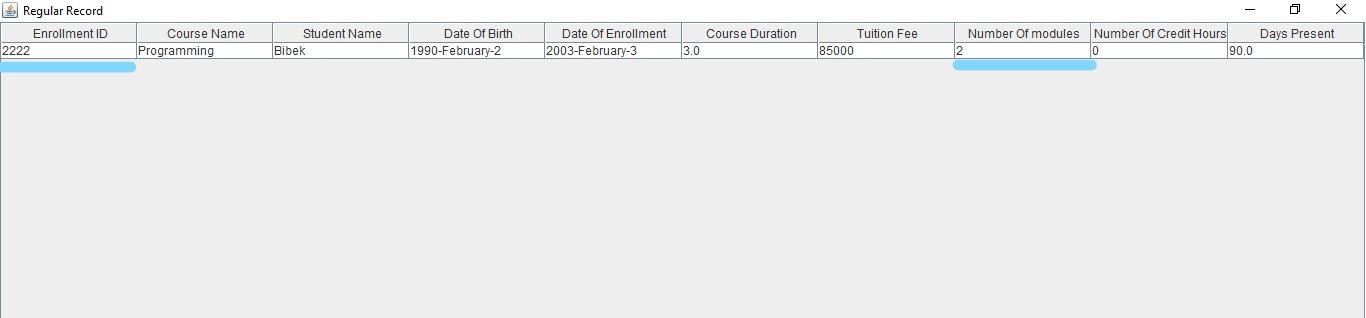
*Figure 31: Logical error*



*Figure 32: Logical error detection*



*Figure 33: Logical error Correction*



*Figure 34: After Logical error correction*

In above figure i mistakenly pass the enrollment id method instead of passing number of modules method which cause the logical error .

# Conclusion

The coursework that was given to us for the creation of a graphical user interface served as the foundation for our first coursework after the assigned task was completed. This coursework's main goal was to gain a better understanding of creating a graphical user interface using the Java platform.

I faced many difficulties as I completed my coursework. I had a lot of problems when I was learning to code a program. We must create our own design first. I had a different design in mind when I started the project, and as I was designing, I found myself having to make numerous changes. I spent a lot of time making those changes and had to remove a lot of lengthy codes in the process. The button actions were more difficult to design than the GUI itself. It required a lot of reasoning, patience, and persistence. It made working on the project difficult for me. Even a very minor issue prevented me from performing button actions for a considerable amount of time. It was challenging to arrange the various text fields, labels, and buttons in the panel created during coding, so the grid bag layout concept was used. However, creating the generated button—for which we used the ActionListener concept—was also a challenging task.

Additionally, the program was created in Intellij Idea, which greatly facilitated the coding process compared to the previous compiler BlueJ. The primary motivation behind developing the Graphical user interface for the classes Student, regular, and dropout to carry out various job tasks was without a doubt that the generated Graphical user interface improved my vision and made it easier to get data from the Arraylist.

# Bibliography

Bharti, N., 2020. *GeeksforGeeks.* [Online] Available at: https://[www.geeksforgeeks.org/runtime-errors/](http://www.geeksforgeeks.org/runtime-errors/) [Accessed 5 August 2023].

Mueller, J. P., 2016. *Dummies.* [Online] Available at: https:/[/www.d](http://www.dummies.com/article/technology/programming-web-)u[mmies.com/article/technology/programming-web-](http://www.dummies.com/article/technology/programming-web-) design/java/syntactical-errors-in-java-153704/

[Accessed 5 August 2023].

Rouse, M., 2023. *Techopedia.* [Online] Available at: https://[www.techopedia.com/definition/8122/logic-](http://www.techopedia.com/definition/8122/logic-) error#:~:text=A%20logic%20error%20is%20an,program%20to%20crash%20when%20r unning.

[Accessed 5 August 2023].

# Appendix

package Coursework\_1; import javax.swing.\*;

import javax.swing.table.DefaultTableModel; import java.awt.event.\*;

import java.awt.\*; import java.util.\*;

//create class named studentgui public class StudentGui {

//Arraylist of Student Class

public ArrayList<Student> studentArrayList = new ArrayList<>();

//Constructor of studentGui public StudentGui(){

//Create frame of studentgui

JFrame frame = new JFrame("Student Gui");

//Declear gridbaglayout

GridBagLayout layout = new GridBagLayout(); GridBagConstraints gbc = new GridBagConstraints();

//Create panel

JPanel panel = new JPanel();

//construct label, textfield and set font

JLabel studentName = new JLabel("Student Name: "); studentName.setFont(new Font("Serif", Font.BOLD,15)); JTextField studentNameTf = new JTextField(20);

//create label of dateofbirth

JLabel dateOfBirth = new JLabel("Date Of Birth: "); dateOfBirth.setFont(new Font("Serif", Font.BOLD,15));

//create array of birth years, months and days

String birthYears[] = {"Select Year","1990","1991","1992","1993","1994","1995","1996","1997","1998",

"1999","2000","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021","2022","2023"};

String birthMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String birthDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10",

"11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24", "25", "26",

"27", "28",

"29", "30", "31"};

//create combo box and passing birth years, months and days array JComboBox birthYearsComboBox = new JComboBox<>(birthYears);

JComboBox birthMonthsComboBox = new JComboBox(birthMonths); JComboBox birthDaysComboBox = new JComboBox(birthDays);

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15)); JTextField enrollmentIdTf = new JTextField(20);

//construct label, textfield and set font

JLabel courseName = new JLabel("Course Name: "); courseName.setFont(new Font("Serif", Font.BOLD,15)); JTextField courseNameTf = new JTextField(20);

//construct label, textfield and set font

JLabel courseDuration = new JLabel("Course Duration: "); courseDuration.setFont(new Font("Serif", Font.BOLD,15)); JTextField courseDurationTf = new JTextField(20);

JLabel dateOfEnrollment = new JLabel("Date Of Enrollment: "); dateOfEnrollment.setFont(new Font("Serif", Font.BOLD,15));

String enrollmentYears[] = {"Select Year","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021","2022","2023"};

String enrollmentMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String enrollmentDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7", "8", "9",

"10", "11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24", "25",

"26", "27", "28",

"29", "30", "31"};

JComboBox enrollmentYearsComboBox = new JComboBox<>(enrollmentYears);

JComboBox enrollmentMonthsComboBox = new JComboBox(enrollmentMonths);

JComboBox enrollmentDaysComboBox = new JComboBox(enrollmentDays);

//construct label, textfield and set font

JLabel tuitionFeeLabel = new JLabel("Tuition Fee: "); tuitionFeeLabel.setFont(new Font("Serif", Font.BOLD,15)); JTextField tuitionFeeTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfModulesLabel = new JLabel("Number Of Modules: "); numberOfModulesLabel.setFont(new Font("Serif", Font.BOLD, 15)); JTextField numberOfModulesTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfCreditHoursLabel = new JLabel("Number Of Credit Hours:

");

numberOfCreditHoursLabel.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfCreditHoursTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfDaysPresentLabel = new JLabel("Number Of Days Present:

");

numberOfDaysPresentLabel.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfDaysPresentTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfRemainingModulesLabel = new JLabel("Number Of Remaining Modules: ");

numberOfRemainingModulesLabel.setFont(new Font("Serif", Font.BOLD,15));

JTextField numberOfRemainingModulesTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfMonthsAttenedLabel = new JLabel("Number Of Months Attend: ");

numberOfMonthsAttenedLabel.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfMonthsAttendedTf = new JTextField(20);

JLabel dateOfDropout = new JLabel("Date Of Dropout: "); dateOfDropout.setFont(new Font("Serif", Font.BOLD,15));

String dropoutYears[] = {"Select Year","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021","2022","2023"};

String dropoutMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String dropoutDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7", "8", "9",

"10", "11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24", "25",

"26", "27", "28",

"29", "30", "31"};

JComboBox dropoutYearsComboBox = new JComboBox(dropoutYears); JComboBox dropoutMonthsComboBox = new JComboBox(dropoutMonths); JComboBox dropoutDaysComboBox = new JComboBox(dropoutDays);

//construct label, textfield and set font

JLabel remainingAmount = new JLabel("Remaining Amount: "); remainingAmount.setFont(new Font("Serif", Font.BOLD,15)); JTextField remainingAmountTf = new JTextField(20);

//construct buttons

JButton addRegularStudentButton = new JButton("Add To Regular");

JButton addDropoutStudentButton = new JButton("Add To Dropout"); JButton calculatePresentPercentButton = new JButton("Calculate Present

Persent");

JButton grantCertificateButton = new JButton("Grant Certificate"); JButton paybillsButton = new JButton("Pay Bills");

JButton removeStudentButton = new JButton("Remove Student"); JButton displayButton = new JButton("Display");

JButton clearButton = new JButton("Clear");

//set layout in panel panel.setLayout(layout);

//Set grid x and y and insets to arrage textfield, label and button gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(studentName,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(studentNameTf, gbc);

gbc.gridy = 1;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0);

panel.add(dateOfBirth,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,-130,0,0); panel.add(birthYearsComboBox,gbc);

gbc.gridx = 2; panel.add(birthMonthsComboBox,gbc);

gbc.gridx = 3;

gbc.insets = new Insets(10,0,0,0); panel.add(birthDaysComboBox,gbc);

gbc.gridy = 2;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 3;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(courseName, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(courseNameTf, gbc);

gbc.gridy = 4;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(courseDuration, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(courseDurationTf, gbc);

gbc.gridy = 5;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(dateOfEnrollment, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,-130,0,0); panel.add(enrollmentYearsComboBox, gbc);

gbc.gridx = 2; panel.add(enrollmentMonthsComboBox, gbc);

gbc.gridx = 3;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentDaysComboBox, gbc);

gbc.gridy = 6;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(tuitionFeeLabel, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(tuitionFeeTf, gbc);

gbc.gridy = 0;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfModulesLabel, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfModulesTf, gbc);

gbc.gridy = 1;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfCreditHoursLabel, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfCreditHoursTf, gbc);

gbc.gridy = 2;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresentLabel, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresentTf, gbc);

gbc.gridy = 3;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfRemainingModulesLabel, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfRemainingModulesTf, gbc);

gbc.gridy = 4;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfMonthsAttenedLabel, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfMonthsAttendedTf, gbc);

gbc.gridy = 5;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(dateOfDropout, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,-130,0,0); panel.add(dropoutYearsComboBox, gbc);

gbc.gridx = 10; panel.add(dropoutMonthsComboBox, gbc);

gbc.gridx = 11;

gbc.insets = new Insets(10,0,0,0); panel.add(dropoutDaysComboBox, gbc);

gbc.gridy = 6;

gbc.gridx = 8;

gbc.insets = new Insets(10,0,0,0); panel.add(remainingAmount, gbc);

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(remainingAmountTf, gbc);

gbc.gridy = 7;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL;

gbc.insets = new Insets(10,0,0,0); panel.add(addRegularStudentButton, gbc);

gbc.gridy = 8;

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(addDropoutStudentButton, gbc);

gbc.gridy = 9;

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(calculatePresentPercentButton, gbc);

gbc.gridy = 10;

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(displayButton, gbc);

gbc.gridy = 7;

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(grantCertificateButton, gbc);

gbc.gridy = 8;

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(paybillsButton, gbc);

gbc.gridy = 9;

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(removeStudentButton, gbc);

gbc.gridy = 10;

gbc.gridx = 9;

gbc.insets = new Insets(10,0,0,0); panel.add(clearButton, gbc);

//addregular button functionality addRegularStudentButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//create frame

JFrame frame1 = new JFrame("Regular Student");

//Declear grid layout

GridBagLayout layout = new GridBagLayout();

GridBagConstraints gbc = new GridBagConstraints();

//construct new panel

JPanel panel = new JPanel();

//construct label, textfield and set font

JLabel studentName = new JLabel("Student Name: "); studentName.setFont(new Font("Serif", Font.BOLD,15)); JTextField studentNameTf = new JTextField(20);

JLabel dateOfBirth = new JLabel("Date Of Birth: "); dateOfBirth.setFont(new Font("Serif", Font.BOLD,15));

String birthYears[] = {"Select Year","1990","1991","1992","1993","1994","1995","1996","1997","1998",

"1999","2000","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021","2022","2023"};

String birthMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String birthDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7", "8", "9",

"10", "11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24", "25",

"26", "27", "28",

"29", "30", "31"};

JComboBox birthYearsComboBox = new JComboBox<>(birthYears); JComboBox birthMonthsComboBox = new JComboBox(birthMonths); JComboBox birthDaysComboBox = new JComboBox(birthDays);

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15)); JTextField enrollmentIdTf = new JTextField(20);

//construct label, textfield and set font

JLabel courseName = new JLabel("Course Name: "); courseName.setFont(new Font("Serif", Font.BOLD,15)); JTextField courseNameTf = new JTextField(20);

//construct label, textfield and set font

JLabel courseDuration = new JLabel("Course Duration: "); courseDuration.setFont(new Font("Serif", Font.BOLD,15)); JTextField courseDurationTf = new JTextField(20);

//Create label and set font desgin

JLabel dateOfEnrollment = new JLabel("Date Of Enrollment: "); dateOfEnrollment.setFont(new Font("Serif", Font.BOLD,15));

//Create array of enrollment years, months and days

String enrollmentYears[] = {"Select Year","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021","2022","2023"};

String enrollmentMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String enrollmentDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7",

"8", "9", "10", "11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",

"25", "26", "27", "28",

"29", "30", "31"};

//Create object of combo box and passing array of enrollment years, month and days

JComboBox enrollmentYearsComboBox = new JComboBox<>(enrollmentYears);

JComboBox enrollmentMonthsComboBox = new JComboBox(enrollmentMonths);

JComboBox enrollmentDaysComboBox = new JComboBox(enrollmentDays);

//construct label, textfield and set font

JLabel tuitionFee = new JLabel("Tuition Fee: "); tuitionFee.setFont(new Font("Serif", Font.BOLD,15)); JTextField tuitionFeeTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfModules = new JLabel("Number Of Modules: "); numberOfModules.setFont(new Font("Serif", Font.BOLD, 15)); JTextField numberOfModulesTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfCreditHours = new JLabel("Number Of Credit Hours:

");

numberOfCreditHours.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfCreditHoursTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfDaysPresent = new JLabel("Number Of Days Present:

");

numberOfDaysPresent.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfDaysPresentTf = new JTextField(20);

//Create add button

JButton addButton = new JButton("Add");

//Create cancel button

JButton cancelButton = new JButton("Cancel");

//set layout in panel panel.setLayout(layout);

gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(studentName,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(studentNameTf, gbc);

gbc.gridy = 1;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(dateOfBirth,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,-130,0,0); panel.add(birthYearsComboBox,gbc);

gbc.gridx = 2; panel.add(birthMonthsComboBox,gbc);

gbc.gridx = 3;

gbc.insets = new Insets(10,0,0,0); panel.add(birthDaysComboBox,gbc);

gbc.gridy = 2;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 3;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(courseName, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(courseNameTf, gbc);

gbc.gridy = 4;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(courseDuration, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(courseDurationTf, gbc);

gbc.gridy = 5;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(dateOfEnrollment, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,-130,0,0); panel.add(enrollmentYearsComboBox, gbc);

gbc.gridx = 2; panel.add(enrollmentMonthsComboBox, gbc);

gbc.gridx = 3;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentDaysComboBox, gbc);

gbc.gridy = 6;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0);

panel.add(tuitionFee, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(tuitionFeeTf, gbc);

gbc.gridy = 7;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfModules, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfModulesTf, gbc);

gbc.gridy = 8;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfCreditHours, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfCreditHoursTf, gbc);

gbc.gridy = 9;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresent, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresentTf, gbc);

gbc.gridy = 11;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(addButton, gbc);

gbc.gridy = 13;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(cancelButton, gbc);

//add button functionality

addButton.addActionListener(new ActionListener() { @Override

public void actionPerformed(ActionEvent e) {

//Using if else to check condition

if (studentNameTf.getText().equals("") || birthYearsComboBox.getSelectedIndex()==0 ||

birthMonthsComboBox.getSelectedIndex()==0 || birthDaysComboBox.getSelectedIndex()==0 ||

enrollmentIdTf.getText().equals("") || courseNameTf.getText().equals("") || courseDurationTf.getText().equals("") || enrollmentYearsComboBox.getSelectedIndex()==0 ||

enrollmentMonthsComboBox.getSelectedIndex()==0 || enrollmentDaysComboBox.getSelectedIndex()==0 || tuitionFeeTf.getText().equals("") || numberOfModulesTf.getText().equals("") ||

numberOfCreditHoursTf.getText().equals("") || numberOfDaysPresentTf.getText().equals("")){

JOptionPane.showMessageDialog(frame1, "Invalid");

}else {

//using try catch to show popup if user input text in field of integer

and double

try {

String studentName = studentNameTf.getText();

String dateOfBirth = birthYearsComboBox.getSelectedItem()

+ "-" + birthMonthsComboBox.getSelectedItem() + "-" + birthDaysComboBox.getSelectedItem();

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText());

String courseName = courseNameTf.getText();

double courseDuration = Double.parseDouble(courseDurationTf.getText());

|  |  |  |
| --- | --- | --- |
| String dateOfEnrollment  enrollmentYearsComboBox.getSelectedItem() + | "-" | =  + |
| enrollmentMonthsComboBox.getSelectedItem() +  enrollmentDaysComboBox.getSelectedItem(); | "-" | + |

int tuitionFee = Integer.parseInt(tuitionFeeTf.getText());

int numberOfModules = Integer.parseInt(numberOfModulesTf.getText());

double daysPresent = Double.parseDouble(numberOfDaysPresentTf.getText());

//create object of regular class

Regular student = new Regular(enrollmentId, dateOfBirth, courseName, studentName, dateOfEnrollment, courseDuration, tuitionFee, numberOfModules, daysPresent);

//Add in arraylist studentArrayList.add(student);

//popup message

JOptionPane.showMessageDialog(frame1, "Sucessfully

Added");

} catch (Exception e1) {

//popup message

JOptionPane.showMessageDialog(frame1, "Please Enter

Valid Information");

}

}

}

});

//Cancel Button

cancelButton.addActionListener(new ActionListener() { @Override

public void actionPerformed(ActionEvent e) { frame1.setVisible(false); //Setting frame1 invisible false

}

});

frame1.setVisible(true); //set frame1 visisble frame1.add(panel); //add panel in frame1 frame1.pack(); //using pack in frame1

frame1.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

//Default operation Dispose on close

}

});

//Dropout button

addDropoutStudentButton.addActionListener(new ActionListener() { @Override

public void actionPerformed(ActionEvent e) {

//Using if else to check condition

if (studentNameTf.getText().equals("") ||

|  |  |
| --- | --- |
| birthYearsComboBox.getSelectedIndex()==0 birthMonthsComboBox.getSelectedIndex()==0  birthDaysComboBox.getSelectedIndex()==0 || | ||  || |
| enrollmentIdTf.getText().equals("") | || |
| courseDurationTf.getText().equals("") || courseNameTf.getText().equals("") | || |
| enrollmentYearsComboBox.getSelectedIndex()==0 || |  |
| enrollmentMonthsComboBox.getSelectedIndex()==0 | || |
| enrollmentDaysComboBox.getSelectedIndex()==0 | || |
| tuitionFeeTf.getText().equals("")  numberOfRemainingModulesTf.getText().equals("") || | || |
| numberOfMonthsAttendedTf.getText().equals("") | || |
| dropoutYearsComboBox.getSelectedIndex()==0 | || |
| dropoutMonthsComboBox.getSelectedIndex()==0  dropoutDaysComboBox.getSelectedIndex()==0){ | || |

JOptionPane.showMessageDialog(frame, "Please fill the form");

}else {

//using try catch to show popup if user input text in field of integer and

double

try{

//type conversion and provide appropriate data type String studentName = studentNameTf.getText();

String dateOfBirth = birthYearsComboBox.getSelectedItem()+ "-" + birthMonthsComboBox.getSelectedItem()+ "-" + birthDaysComboBox.getSelectedItem();

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText()); String courseName = courseNameTf.getText();

double courseDuration = Double.parseDouble(courseDurationTf.getText());

String dateOfEnrollment =

|  |  |  |
| --- | --- | --- |
| enrollmentYearsComboBox.getSelectedItem()+ | "-" | + |
| enrollmentMonthsComboBox.getSelectedItem()+  +enrollmentDaysComboBox.getSelectedItem(); |  | "-" |

int tuitionFee = Integer.parseInt(tuitionFeeTf.getText());

int numberOfRemainingModules = Integer.parseInt(numberOfRemainingModulesTf.getText());

int numberOfMonthAttends = Integer.parseInt(numberOfMonthsAttendedTf.getText());

String dateOfDropout =

|  |  |  |
| --- | --- | --- |
| dropoutYearsComboBox.getSelectedItem()+ | "-" | + |
| dropoutMonthsComboBox.getSelectedItem()+  dropoutDaysComboBox.getSelectedItem(); | "-" | + |

//create object of dropout class

Dropout dropout = new Dropout(enrollmentId, courseName, dateOfEnrollment, dateOfBirth, studentName, courseDuration, tuitionFee, numberOfRemainingModules, numberOfMonthAttends, dateOfDropout);

//Add in arraylist studentArrayList.add(dropout);

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

}catch (Exception e1){

//popup message

JOptionPane.showMessageDialog(frame, "Please Enter Valid

Information");

}

}

}

});

//Calculate present precentage button calculatePresentPercentButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFrame frame2 = new JFrame("Calculate Present Percentage"); JPanel panel = new JPanel();

GridBagLayout layout = new GridBagLayout(); GridBagConstraints gbc = new GridBagConstraints();

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15));

JTextField enrollmentIdTf = new JTextField(20);

//construct label, textfield and set font

JLabel numberOfDaysPresent = new JLabel("Number Of Days Present:

");

numberOfDaysPresent.setFont(new Font("Serif", Font.BOLD,15)); JTextField numberOfDaysPresentTf = new JTextField(20);

//creat calculate button

JButton calculateButton = new JButton("Calculate");

//create cancel button

JButton cancelButton = new JButton("Cancel");

panel.setLayout(layout);

gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 1;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresent,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(numberOfDaysPresentTf,gbc);

gbc.gridy = 2;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(calculateButton, gbc);

gbc.gridy = 3;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(cancelButton, gbc);

//calculate button functionality calculateButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//Usinf else to check condition

if (enrollmentIdTf.getText().equals("") || numberOfDaysPresentTf.getText().equals("")) {

JOptionPane.showMessageDialog(frame2, "Please fill the

detail");

} else {

//using try catch to show popup if user input text in field of integer

and double

try {

//Type conversion and provide appropriate data type

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText()); double daysPresent =

Integer.parseInt(numberOfDaysPresentTf.getText());

boolean regular = false;

//using for each loop

for (Student student : studentArrayList) {

if (student instanceof Regular && student.getEnrollmentId() == enrollmentId) {

Regular regularStudent = (Regular) student;

//Downcasting regular class as regularstudent

String grade = regularStudent.presentPercentage(daysPresent); //call the regular class present percentage method

//Nested if

if (!grade.isEmpty()) {

JOptionPane.showMessageDialog(frame2, "Present

Percentage: " + grade);

} else {

JOptionPane.showMessageDialog(frame2, "Invalid

input");

}

regular = true; break;

}

}

if (!regular) {

JOptionPane.showMessageDialog(frame2, "Enrollment Id

Doesn't Exists");

}

} catch (Exception e1) {

JOptionPane.showMessageDialog(frame2, "Please enter

valid numbers.");

}

}

}

});

//cancel button functionality cancelButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) { frame2.setVisible(false);

}

});

frame2.setVisible(true); frame2.add(panel); frame2.pack();

frame2.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

}

});

//grantCertificateButtton functionality grantCertificateButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

JFrame frame3 = new JFrame("Grant Certificate");

JPanel panel = new JPanel();

GridBagLayout layout = new GridBagLayout(); GridBagConstraints gbc = new GridBagConstraints();

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15)); JTextField enrollmentIdTf = new JTextField(20);

//construct label, textfield and set font

JLabel courseName = new JLabel("Course Name: "); courseName.setFont(new Font("Serif", Font.BOLD,15)); JTextField courseNameTf = new JTextField(20);

//create label and set font of date of enrollment

JLabel dateOfEnrollment = new JLabel("Date Of Enrollment: "); dateOfEnrollment.setFont(new Font("Serif", Font.BOLD,15));

//create array of enrollment years, month and days

String enrollmentYears[] = {"Select Year","2001","2002","2003","2004","2005","2006","2007",

"2008","2009","2010",

"2011","2012","2013","2014","2015","2016","2017",

"2018","2019","2020","2021"," 2022","2023"};

String enrollmentMonths[] = {"Select Month","January", "February", "March", "April", "May", "June", "July", "August",

"September", "October", "November", " December"};

String enrollmentDays[] = {"Select Day", "1", "2", "3", "4", "5", "6", "7",

"8", "9", "10", "11", "12",

"13", "14", "15", "16", "17", "18", "19", "20", "21", "22", "23", "24",

"25", "26", "27", "28",

"29", "30", "31"};

//create conmbo box and passing enrollment years, months JComboBox enrollmentYearsComboBox = new

JComboBox<>(enrollmentYears);

JComboBox enrollmentMonthsComboBox = new JComboBox(enrollmentMonths);

JComboBox enrollmentDaysComboBox = new JComboBox(enrollmentDays);

//Create grant button

JButton grantButton = new JButton("Grant Certificate");

// cancel button

JButton cancelButton = new JButton("Cancel");

//set layout in panel panel.setLayout(layout);

gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 1;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(courseName,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(courseNameTf,gbc);

gbc.gridy = 2;

gbc.gridx = 0;

gbc.insets = new Insets(10,0,0,0); panel.add(dateOfEnrollment, gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,-130,0,0); panel.add(enrollmentYearsComboBox, gbc);

gbc.gridx = 2;

panel.add(enrollmentMonthsComboBox, gbc);

gbc.gridx = 3;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentDaysComboBox, gbc);

gbc.gridy = 3;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(grantButton, gbc);

gbc.gridy = 4;

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0);

gbc.fill = GridBagConstraints.HORIZONTAL; panel.add(cancelButton, gbc);

//grantButtonn function grantButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

if (enrollmentIdTf.getText().equals("") || courseNameTf.getText().equals("") ||

enrollmentYearsComboBox.getSelectedIndex() == 0 ||

enrollmentMonthsComboBox.getSelectedIndex() == 0 || enrollmentDaysComboBox.getSelectedIndex() == 0) {

JOptionPane.showMessageDialog(frame3, "Invalid");

} else {

//using try catch to show popup if user input text in field of integer

and double

try {

//Type conversion and provide appropriate datatype

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText());

// try conversion

String courseName = courseNameTf.getText();

String dateOfEnrollment =

|  |  |  |  |
| --- | --- | --- | --- |
| enrollmentYearsComboBox.getSelectedItem() | + | "-" | + |
| enrollmentMonthsComboBox.getSelectedItem()  enrollmentDaysComboBox.getSelectedItem(); | + | "-" | + |

boolean scholarship = false;

//Using for each loop

for (Student student : studentArrayList) {

if (student instanceof Regular && student.getEnrollmentId()

== enrollmentId) {

Regular regularStudent = (Regular) student;

//Downcasting regular class as regularStudent

regularStudent.grantCertificate(courseName, enrollmentId, dateOfEnrollment); //Call the grantcertificate method of regular class

//nested if

if (regularStudent.getIsGrantedScholarship() == true) { JOptionPane.showMessageDialog(frame3,

"Scholarship has been granted with certificate");

} else {

JOptionPane.showMessageDialog(frame3, "Scholarship is not granted but certificate is granted");

}

scholarship = true; break;

}

}

if (!scholarship) {

JOptionPane.showMessageDialog(frame3, "Enrollment ID

not found.");

}

} catch (Exception e1) { JOptionPane.showMessageDialog(frame3, "Invalid");

}

}

}

});

// cancel button functionality cancelButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//set frame3 visible false frame3.setVisible(false);

}

});

frame3.setVisible(true); frame3.add(panel); frame3.pack();

frame3.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

}

});

paybillsButton.addActionListener(new ActionListener() { @Override

public void actionPerformed(ActionEvent e) { JFrame frame4 = new JFrame("Pay Bills"); JPanel panel = new JPanel();

GridBagLayout layout = new GridBagLayout(); GridBagConstraints gbc = new GridBagConstraints();

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15)); JTextField enrollmentIdTf = new JTextField(20);

JButton paysButton = new JButton("Pay Now"); JButton cancelButton = new JButton("Cancel");

panel.setLayout(layout);

gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 1;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(paysButton, gbc);

gbc.gridy = 3;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(cancelButton, gbc);

//pays button functionality paysButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//using if else to check condition

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

JOptionPane.showMessageDialog(frame4, "Please enter id to

check");

}else {

//using try catch to show popup if user input text in field of integer

and double

try {

//Type conversion and provide appropriate datatype

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText());

boolean billpay = false;

for (Student dropoutStudent : studentArrayList) {

if (dropoutStudent instanceof Dropout && dropoutStudent.getEnrollmentId()==enrollmentId) {

Dropout student = (Dropout) dropoutStudent;

//Downcasting dropout class as dropout student

((Dropout) dropoutStudent).billsPayable(); //call dropout class billspayable method

remainingAmountTf.setText(((Dropout) dropoutStudent).getRemainingAmount()+"");

//Nested if

if(((Dropout) dropoutStudent).getHasPaid()==true){

JOptionPane.showMessageDialog(frame,"Bill has

been paid");

}

else {

JOptionPane.showMessageDialog(frame,"Bill has to

paid");

}

billpay = true; break;

}

}

if (!billpay) {

JOptionPane.showMessageDialog(frame, "Given Enrollment ID not found.");

}

}catch (Exception e1){ JOptionPane.showMessageDialog(frame4, "Invalid");

}

}

}

});

//cancel button functionality cancelButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

frame4.setVisible(false);

}

});

frame4.setVisible(true); frame4.add(panel); frame4.pack();

frame4.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

}

});

//remove button functionality removeStudentButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//Create frame

JFrame frame5 = new JFrame("Remove Student");

//Create panel objcet

JPanel panel = new JPanel();

//create gridbaglayout object

GridBagLayout layout = new GridBagLayout();

//create gridbagconstraints object GridBagConstraints gbc = new GridBagConstraints();

//construct label, textfield and set font

JLabel enrollmentId = new JLabel("Enrollment Id: "); enrollmentId.setFont(new Font("Serif", Font.BOLD,15)); JTextField enrollmentIdTf = new JTextField(20);

//construct rmove button

JButton removeButton = new JButton("Remove");

//construct cancel button

JButton cancelButton = new JButton("Cancel");

//add layout in panel panel.setLayout(layout);

//set grid x and y for arrangement and design gbc.gridy = 0;

gbc.gridx =0;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentId,gbc);

gbc.gridx = 1;

gbc.insets = new Insets(10,0,0,0); panel.add(enrollmentIdTf, gbc);

gbc.gridy = 1;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(removeButton, gbc);

gbc.gridy = 3;

gbc.gridx = 1;

gbc.fill = GridBagConstraints.HORIZONTAL; gbc.insets = new Insets(10,0,0,0); panel.add(cancelButton, gbc);

//remove button functionality removeButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) { try{

//Type conversion and providing appropriate datatype

int enrollmentId = Integer.parseInt(enrollmentIdTf.getText());

boolean remove = false;

//for each loop

for (Student removeStudent : studentArrayList) {

if (removeStudent instanceof Dropout && removeStudent.getEnrollmentId() == enrollmentId) {

Dropout dropout=(Dropout) removeStudent; //downcasting

dropout as dropout

dropout.removeStudent(); //call the method of remove student of dropout class

//nested if if(dropout.getHasPaid()==false) {

//Popup message JOptionPane.showMessageDialog(frame, "Dropout

student with given Enrollment ID removed successfully!");

}

else{

//Popup message

JOptionPane.showMessageDialog(frame,"No need to

remove");

}

remove = true; break;

}

}

if (!remove) {

//Popup message

JOptionPane.showMessageDialog(frame, "Dropout student with the given Enrollment ID not found.");

}

} catch (NumberFormatException ex) {

//Popup message

JOptionPane.showMessageDialog(frame, "Invalid input. Please enter valid numbers.");

}

}

});

//cancel button functionality cancelButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//set frame5 visibity to false frame5.setVisible(false);

}

});

//add panel in frame5 frame5.add(panel);

//set visibilty to true frame5.setVisible(true);

//set default close opertion to dispose on close frame5.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

//set pack in frame which will provide require frame size frame5.pack();

}

});

//Displya button functionality displayButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//Using for each loop

for (Student student : studentArrayList) { if (student instanceof Regular) {

JFrame frame8 = new JFrame("Regular Record"); //frame of regular student record

//Declear regular header array

String[] regularHeader = {"Enrollment ID", "Course Name", "Student Name", "Date Of Birth", "Date Of Enrollment", "Course Duration", "Tuition Fee",

"Number Of modules", "Number Of Credit Hours", "Days

Present"};

//Create table model

DefaultTableModel model = new DefaultTableModel(regularHeader,0); //declaering table model and set header and row

Regular rStudent = (Regular) student; //Downcasting

int enrollmentId = (rStudent.getEnrollmentId()); //call get enrollment id method

String courseName = (rStudent.getCourseName()); //call get courseName method

String studentName = (rStudent.getStudentName()); //call get StudentName method

String dateOfBirth = (rStudent.getDateOfBirth()); //call get dateOfBirth method

String dateOfEnrollment = (rStudent.getDateOfEnrollment()); //call get dateOfEnrollment method

double courseDuration = (rStudent.getCourseDuration()); //call get courseDuration method

int tuitionFee = (rStudent.getTuitionFee()); //call get tuitionFee

method

int numOfModules = (rStudent.getNumOfModules()); //call get numOfModules method

int numOfCreditHours = (rStudent.getNumOfCreditHours()); //call get numofCreditHours method

double daysPresent = (rStudent.getDaysPresent()); //call get daysPresent method

//Add row on model and store object model.addRow(new

Object[]{enrollmentId,courseName,studentName,dateOfBirth,dateOfEnrollment,c ourseDuration,tuitionFee,numOfModules,

numOfCreditHours,daysPresent});

//construct tabele and add table model JTable regularTable = new JTable(model);

//Construct scrollpaner and add table

JScrollPane regularScroll = new JScrollPane(regularTable);

//add scrollpane in frame frame8.add(regularScroll);

//default operation on dispose on close

frame8.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

//Set size frame8.setSize(900, 300);

//Set visibity to true frame8.setVisible(true);

}

if(student instanceof Dropout){

//Create frame

JFrame frame9 = new JFrame("Dropout Record");

//Store tabel header in arry

String[] dropoutHeader = {"Enrollment ID", "Course Name", "Student Name", "Date Of Birth", "Date Of Enrollment", "Course Duration", "Tuition Fee",

"Number Of Remaining Modules", "Number Of Months Attended", "Date Of Dropout", "Remaining Amount"};

//Construct table model

DefaultTableModel model = new DefaultTableModel(dropoutHeader,0);

Dropout dStudent = (Dropout) student; //oencasting dropout class

as dStudent

int enrollmentId = (dStudent.getEnrollmentId()); //call get enrollment id method

String courseName = (dStudent.getCourseName()); //call get courseName method

String studentName = (dStudent.getStudentName()); //call get StudentName method

String dateOfBirth = (dStudent.getDateOfBirth()); //call get dateOfBirth method

String dateOfEnrollment = (dStudent.getDateOfEnrollment()); //call get dateOfEnrollment method

double courseDuration = (dStudent.getCourseDuration()); //call get courseDuration method

int tuitionFee = (dStudent.getTuitionFee()); //call get tuitionFee

method

int numOfRemainingModules = (dStudent.getNumOfRemainingModules()); //call get numOfRemainingModules method

int numOfMonthsAttended = (dStudent.getNumOfMonthsAttended()); //call get numofMonthsAttend method

String dateOfDropout = (dStudent.getDateOfDropout()); //call get date of dropout method

int remainigAmount = (dStudent.getRemainingAmount()); //call get remainig amount method

//Add row on model and store object model.addRow(new

Object[]{enrollmentId,courseName,studentName,dateOfBirth,dateOfEnrollment,

courseDuration,tuitionFee, numOfRemainingModules, numOfMonthsAttended, dateOfDropout, remainigAmount});

//construct tabele and add table model JTable regularTable = new JTable(model);

//Construct scrollpaner and add table

JScrollPane regularScroll = new JScrollPane(regularTable);

//add scrollpane in frame frame9.add(regularScroll);

//default operation on dispose on close

frame9.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

//Set size frame9.setSize(1000, 300);

//Set visibity to true frame9.setVisible(true);

}

}

}

});

//

//Clear button functionality clearButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

//set all textfield to empty string and combo box to index 0 studentNameTf.setText(""); birthYearsComboBox.setSelectedIndex(0); birthMonthsComboBox.setSelectedIndex(0); birthDaysComboBox.setSelectedIndex(0); enrollmentIdTf.setText("");

courseNameTf.setText(""); courseDurationTf.setText(""); enrollmentYearsComboBox.setSelectedIndex(0); enrollmentMonthsComboBox.setSelectedIndex(0); enrollmentDaysComboBox.setSelectedIndex(0); tuitionFeeTf.setText(""); numberOfModulesTf.setText(""); numberOfCreditHoursTf.setText(""); numberOfDaysPresentTf.setText(""); numberOfRemainingModulesTf.setText("");

numberOfMonthsAttendedTf.setText("");

dropoutYearsComboBox.setSelectedIndex(0); dropoutMonthsComboBox.setSelectedIndex(0); dropoutDaysComboBox.setSelectedIndex(0); remainingAmountTf.setText("");

}

});

//add panel in frame frame.add(panel);

//set size of frame frame.setSize(1500,800);

//set frame visibity true frame.setVisible(true);

//set default operation dispose on close frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

}

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

new StudentGui();

}

}