**Centennial College**

**COMP 228: Java Programming**

**LAB #4 – Developing UI using JavaFx**

**Studen**t: **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Due Date: **Week 10**.

Purpose: The purpose of this Lab assignment is to:

1. Practice the use of JavaFx GUI event handling

References: Learning materials for weeks 7 and 8, textbook, and other references (if any)

Be sure to read the following general instructions carefully:

This lab should be completed individually by all the students.

YOU NEED TO SUBMIT THE FOLLOWING 2 DOCUMENTS IN THE DROPBOX TITLED LAB4:

1. THE FIRST ONE IS A WORD DOCUMENT. USE THIS DOCUMENT AND ADD SCREEN SHOTS OF THE RUNNING STATE OF EACH EXERCISE (If there are more than 1 exercise). DO NOT DELETE THE QUESTIONS. THE SCREEN SHOTS SHOULD FOLLOW EACH QUESTION AND COVER ALL THE ASPECTS/FUNCTIONALITIES OF EACH EXERCISE. AFTER THE SCREEN SHOTS PLEASE COPY THE CODE FROM THE CODE WINDOW AND PASTE THE COMPLETE CODE. DO NOT GIVE ME SCREEN SHOTS OF THE CODE. DO NOT ZIP THIS FILE AND KEEP IT SEPARATE FROM YOUR ZIPPED PROGAM FILE.

2. SUBMIT ALSO ONE ZIPPED PROJECT FILE THAT CONTAINS ALL THE EXERISES SEPARATELY INTO THE SAME DROP BOX.

This material provides the necessary information you need to complete the exercises.

You must name your Eclipse project according to the following rule:

**YourFullName\_COMP228Labnumber**

Example: **JohSmith\_COMP228Lab4**

Each exercise should be placed in a separate package named *exercise1*, *exercise2*, etc.

Submit your assignment in a **zip file** that is named according to the following rule:

**YourLastName\_COMP228Labnumber.zip**

Example: **JohSmith\_COMP228Lab4.zip**

Apply the naming conventions for variables, methods, classes, and packages:

- *variable names* start with a *lowercase* character

- *classes* start with an *uppercase* character

- **packages** use only *lowercase* characters

- *methods* start with a *lowercase* character

## Exercise 1

Write a Java application using JavaFx which allows the user to enter student information (see the sample design below).

#### 

#### The user will enter *full* *name*, *address,* *city, province, postal code, phone number and email* in *text field* controls. The student’s *major* (Computer Science or Business) will be selected from two *radio buttons*.

#### A combo box will display the list of *courses* for each program whenever the user selects the desired program.

#### A course will be added to a *list box w*henever the user selects a course from the corresponding combo box. Make sure that the user cannot add a course several times.

Additional information about the student will be provided from a group of *check boxes* (such as involvement in various activities, etc).

All the information about the student will be displayed in a text *area* component. Use simple JavaFX layout managers, such as FlowPane, BorderPane, and GridPane to create the JavaFX GUI of this application.

![A screenshot of a computer

Description automatically generated]()

package application;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.\*;

import javafx.scene.layout.BorderPane;

import javafx.scene.layout.FlowPane;

import javafx.scene.layout.GridPane;

import javafx.stage.Stage;

public class Main extends Application {

private TextField fullNameField;

private TextField addressField;

private TextField cityField;

private TextField provinceField;

private TextField postalCodeField;

private TextField phoneNumberField;

private TextField emailField;

private RadioButton csRadioButton;

private RadioButton businessRadioButton;

private ComboBox<String> courseComboBox;

private ListView<String> courseListView;

private CheckBox volunteeringCheckBox;

private CheckBox studentCouncilCheckBox;

private TextArea displayTextArea;

*@Override*

public void start(Stage primaryStage) {

primaryStage.setTitle("Student Information");

// Create text fields

fullNameField = new TextField();

addressField = new TextField();

cityField = new TextField();

provinceField = new TextField();

postalCodeField = new TextField();

phoneNumberField = new TextField();

emailField = new TextField();

// Create radio buttons

ToggleGroup programToggleGroup = new ToggleGroup();

csRadioButton = new RadioButton("Computer Science");

csRadioButton.setToggleGroup(programToggleGroup);

businessRadioButton = new RadioButton("Business");

businessRadioButton.setToggleGroup(programToggleGroup);

// Create combo box

courseComboBox = new ComboBox<>();

courseComboBox.setDisable(true);

// Create list view

courseListView = new ListView<>();

// Create check boxes

volunteeringCheckBox = new CheckBox("Volunteering");

studentCouncilCheckBox = new CheckBox("Student Council");

// Create text area

displayTextArea = new TextArea();

displayTextArea.setEditable(false);

// Add event listener to program radio buttons

csRadioButton.setOnAction(event -> setProgramCourses("Computer Science"));

businessRadioButton.setOnAction(event -> setProgramCourses("Business"));

// Create layout

GridPane gridPane = new GridPane();

gridPane.setHgap(10);

gridPane.setVgap(10);

gridPane.setPadding(new javafx.geometry.Insets(10));

gridPane.add(new Label("Full Name:"), 0, 0);

gridPane.add(fullNameField, 1, 0);

gridPane.add(new Label("Address:"), 0, 1);

gridPane.add(addressField, 1, 1);

gridPane.add(new Label("City:"), 0, 2);

gridPane.add(cityField, 1, 2);

gridPane.add(new Label("Province:"), 0, 3);

gridPane.add(provinceField, 1, 3);

gridPane.add(new Label("Postal Code:"), 0, 4);

gridPane.add(postalCodeField, 1, 4);

gridPane.add(new Label("Phone Number:"), 0, 5);

gridPane.add(phoneNumberField, 1, 5);

gridPane.add(new Label("Email:"), 0, 6);

gridPane.add(emailField, 1, 6);

gridPane.add(new Label("Program:"), 0, 7);

gridPane.add(csRadioButton, 1, 7);

gridPane.add(businessRadioButton, 1, 8);

gridPane.add(new Label("Courses:"), 0, 9);

gridPane.add(courseComboBox, 1, 9);

gridPane.add(courseListView, 1, 10);

gridPane.add(new Label("Additional Information:"), 0, 11);

gridPane.add(volunteeringCheckBox, 1, 11);

gridPane.add(studentCouncilCheckBox, 1, 12);

FlowPane buttonPane = new FlowPane();

buttonPane.setPadding(new javafx.geometry.Insets(10));

Button submitButton = new Button("Submit");

submitButton.setOnAction(event -> displayStudentInformation());

buttonPane.getChildren().add(submitButton);

BorderPane borderPane = new BorderPane();

borderPane.setCenter(gridPane);

borderPane.setRight(displayTextArea);

borderPane.setBottom(buttonPane);

primaryStage.setScene(new Scene(borderPane, 500, 400));

primaryStage.show();

}

private void setProgramCourses(String program) {

if (program.equals("Computer Science")) {

courseComboBox.setItems(javafx.collections.FXCollections.*observableArrayList*(

"Java", "C#", "Python"));

} else if (program.equals("Business")) {

courseComboBox.setItems(javafx.collections.FXCollections.*observableArrayList*(

"Economics", "Accounting", "Business Administration"));

}

courseComboBox.setDisable(false);

}

private void displayStudentInformation() {

String fullName = fullNameField.getText();

String address = addressField.getText();

String city = cityField.getText();

String province = provinceField.getText();

String postalCode = postalCodeField.getText();

String phoneNumber = phoneNumberField.getText();

String email = emailField.getText();

String program = csRadioButton.isSelected() ? "Computer Science" : "Business";

String course = courseComboBox.getValue();

String additionalInformation = "";

if (volunteeringCheckBox.isSelected()) {

additionalInformation += "Volunteering ";

}

if (studentCouncilCheckBox.isSelected()) {

additionalInformation += "Student Council";

}

String studentInformation = "Full Name: " + fullName +

"\nAddress: " + address +

"\nCity: " + city +

"\nProvince: " + province +

"\nPostal Code: " + postalCode +

"\nPhone Number: " + phoneNumber +

"\nEmail: " + email +

"\nProgram: " + program +

"\nCourse: " + course +

"\nAdditional Information: " + additionalInformation;

displayTextArea.setText(studentInformation);

}

public static void main(String[] args) {

*launch*(args);

}

}

(10 marks)

**Evaluation:**

|  |  |
| --- | --- |
| **Functionality** |  |
| Correct implementation of UI | 40% |
| Correct implementation of event handling and display of results | 40% |
| Comments, correct naming of variables, methods, classes, etc. | 5% |
| **Friendly input/output** | 15% |
| **Total** | 100% |