Department of Computer Science & Engineering, SDMCET, Dharwad-2

AOOP Assignment Submission Report

[Submitted as part of CTA Assignment No-2]

Course:	Advanced Object-Oriented Programming	Course Code:	18UCSE508
Semester:	V	Division:	A

Submitted by:

USN:	2SD20CS081	Name:	PRESHITA DESAI

1. Problem Definition:

Q1. Write a Java program to build the GUI application using JavaFX for the following requirements:

a) Read user name and password using appropriate JavaFX controls.

b) Validate the input. If the user name and password are matched with the assumed values, then display the welcome scene with proper text.

c) If username and password don't match, then raise appropriate exception.package application;

2. Java Program:

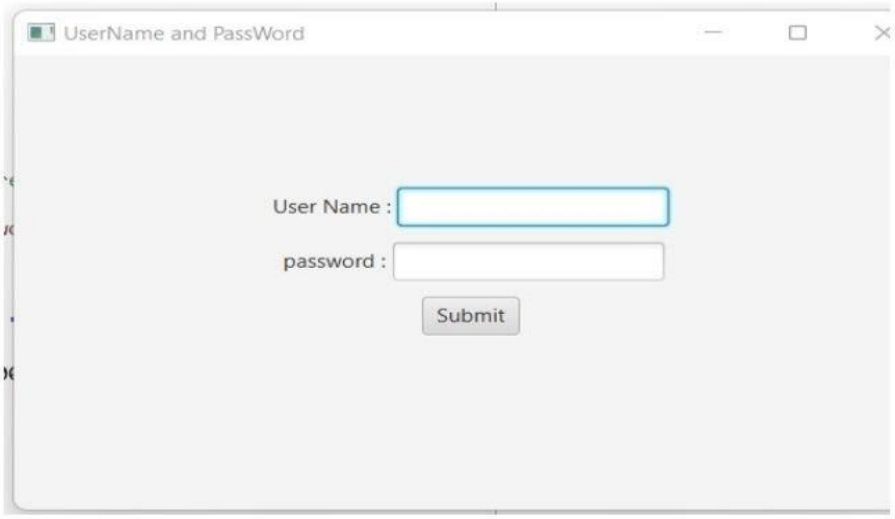
```
package application;


  /*Java program to build GUI application using javaFx
   * Date:15-10-22
   * USN:2SD20CS007
   */
        import javafx.application.Application;
        import javafx.geometry.Pos;
        import javafx.scene.Scene;
        import javafx.scene.control.Button;
        import javafx.scene.control.Label;
        import javafx.scene.control.PasswordField;
        import javafx.scene.control.TextField;
        import javafx.scene.layout.FlowPane;
        import javafx.scene.layout.HBox;
        import javafx.scene.layout.VBox;
        import javafx.stage.Stage;
```

```
public class UserNamePass extends Application {
	public static void main(String[] args) {
	launch(args);
	}
	@Override
	public void start(Stage myStage) {
	// TODO Auto-generated method stub
	myStage.setTitle("UserName and PassWord");

	VBox vbox = new VBox();
	HBox hbox = new HBox();

	Label label = new Label("User Name : ");
	TextField tf = new TextField();

	// layout for component
	HBox hbox2 = new HBox();

	Label label2 = new Label(" password : ");
	PasswordField pass = new PasswordField();

	// to keep components center
	hbox.setAlignment(Pos.CENTER);
	hbox2.setAlignment(Pos.CENTER);

	//adding components to the horizontal layout
	hbox.getChildren().addAll(label,tf);
```

```
hbox2.getChildren().addAll(label2,pass);

// creating the button
Button btn = new Button("Submit");

// label for show results
Label label1 = new Label("");

// assumed value for validation
String username = "20cs107";
String password = "preshita";
// setting action on button
btn.setOnAction(e -> {
// getting the values from the field
String EUsername = tf.getText();
String Epassword = pass.getText();


// if entered username and password are equal then create a new welcome
//Scene
if(username.equals(EUsername) && password.equals(Epassword)) {
// label1.setText(" : WELCOME : ");
FlowPane flowpane = new FlowPane();
flowpane.setAlignment(Pos.CENTER);
Label welcome = new Label(": Welcome :");
flowpane.getChildren().add(welcome);
Scene myScene1 = new Scene(flowpane,500,300);
myStage.setScene(myScene1);
```

```
}else {
try {
throw new MyException();
}catch(MyException e1){
label1.setText(e1.toString());
}
}
});

// adding horizontal components to the main vertical layout
vbox.getChildren().addAll(hbox,hbox2,btn,label1);

// adding layout to the scene
Scene myScene = new Scene(vbox,500,300);

// spacing between the vertical components
vbox.setSpacing(10);
vbox.setAlignment(Pos.CENTER);

myStage.setScene(myScene);

myStage.show();
}
}
class MyException extends Exception{
public String toString() {
return "Invaid UserName and Password";
}}
```

3. Screen Shots of Execution:



Problem Definition:

Q2. Write a Java program to build the GUI application using JavaFX for the following requirements:

a) Create a Menu control to display the menu items: File, Edit & Help.

b) Create sub menus in the order: File → New, Open & Save. Edit → Cut, Copy & Paste. Help → Help Centre, About Us

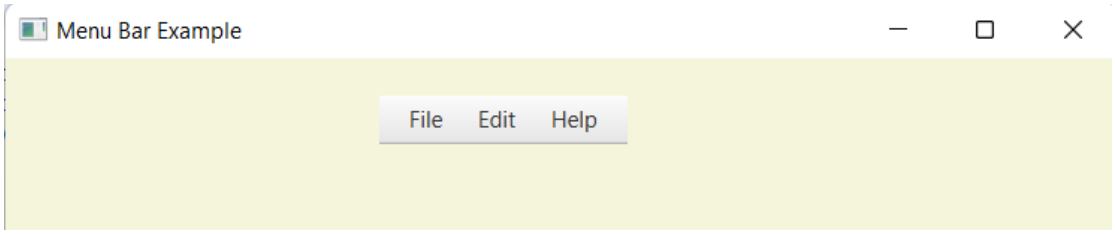
1. Java Program:

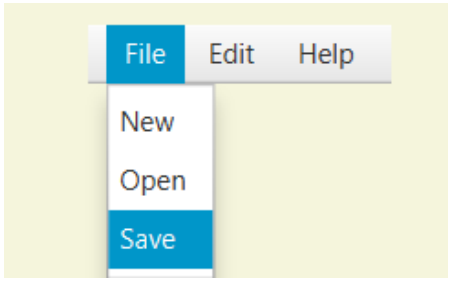
```
package application;


/*Java program to build GUI application using javaFx
 a) Create a Menu control to display the menu items: File, Edit & Help.
 b) Create sub menus in the order: File → New, Open & Save. Edit → Cut, Copy &  Paste.
Help → Help Centre, About Us
 * Date:15-10-22
 * USN:2SD20CS007
 */


import javafx.application.Application;
import javafx.scene.Group;
import javafx.scene.Scene;
import javafx.scene.control.Menu;
import javafx.scene.control.MenuBar;
import javafx.scene.control.MenuItem;
import javafx.scene.paint.Color;
import javafx.stage.Stage;
public class MenuBar1 extends Application {
```

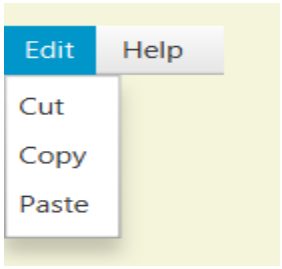
```
public void start(Stage stage) {
//Creating file menu
Menu file = new Menu("File");
//Creating file menu items
MenuItem item1 = new MenuItem("New");
MenuItem item2 = new MenuItem("Open");
MenuItem item3 = new MenuItem("Save");
//Adding all the menu items to the file menu
file.getItems().addAll(item1, item2, item3);
//Creating edit menu
Menu edit = new Menu("Edit");
//Creating fileList menu items
MenuItem item6 = new MenuItem("Cut");
MenuItem item7 = new MenuItem("Copy");
MenuItem item8 = new MenuItem("Paste");
//Adding all the items to File List menu
edit.getItems().addAll(item6, item7, item8);
//Creating help menu
Menu help = new Menu("Help");
MenuItem item9 = new MenuItem("Help center");
MenuItem item10 = new MenuItem("About Us");
help.getItems().addAll(item9, item10);
//Creating a menu bar
MenuBar menuBar = new MenuBar();


menuBar.setTranslateX(200);
menuBar.setTranslateY(20);
//Adding all the menus to the menu bar
```

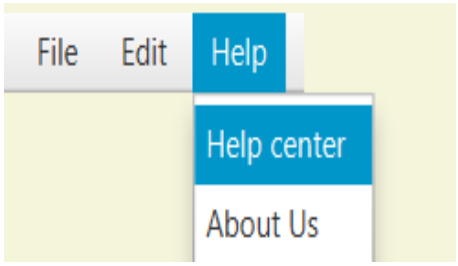
```
menuBar.getMenus().addAll(file, edit, help);
//Setting the stage
Group root = new Group(menuBar);
Scene scene = new Scene(root, 595, 200, Color.BEIGE);
stage.setTitle("Menu Bar Example");
stage.setScene(scene);
stage.show();
}
public static void main(String args[]){
launch(args);
}
}
```

2. Screen Shots of Execution:









1. Problem Definition:

Q3. Write a Java program to build the GUI application using JavaFX for the following requirements:

a) Create Context menu involving the menu items in the order: New & View.

b) Create sub menus for the above main context menu:

New → File, Folder & Image. View → Large, Medium & Small.

The context menu must be displayed on right-click of the mouse button.

2. Java Program:

```
/* Java program to build the GUI application using JavaFX for the following requirements:
a) Create Context menu involving the menu items in the order: New & View.
b) Create sub menus for the above main context menu: New->File, Folder & Image.
View → Large, Medium & Small.
The context menu must be displayed on right-click of the mouse button
* Date:15-10-22
* USN:2SD20CS007
*/
package application;
import java.io.FileNotFoundException;
import javafx.application.Application;
import javafx.geometry.Insets;
import javafx.scene.Group;
import javafx.scene.Scene;
import  javafx.scene.control.Button;
import javafx.scene.control.ContextMenu;
import javafx.scene.control.MenuItem;
```

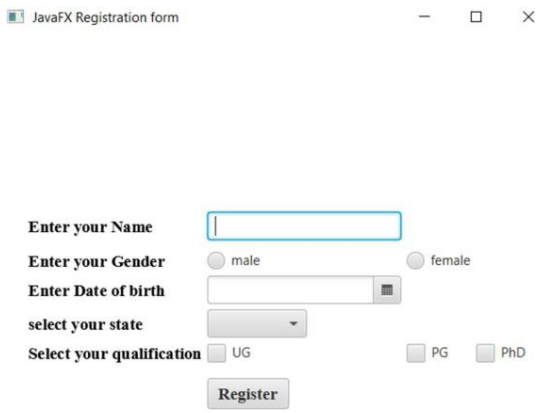
```
//import javafx.scene.control.TextField;
import javafx.scene.layout.HBox;
import javafx.scene.paint.Color;
import javafx.stage.Stage;
public class CustomMenuItem extends Application {
 public void start(Stage stage) throws FileNotFoundException {
 //Creating the image view
 Button button1 = new Button("new");
 Button button2 = new Button("view");
 //TextField textField = new TextField();
 //Creating a context menu
 ContextMenu contextMenu1 = new ContextMenu();
 //Creating the menu Items for the context menu
 MenuItem item1 = new MenuItem("file");
 MenuItem item2 = new MenuItem("folder");
 MenuItem item3 = new MenuItem("image");
 contextMenu1.getItems().addAll(item1, item2,item3);
 //Adding the context menu to the button and the text field
 ContextMenu contextMenu2 = new ContextMenu();
 //Creating the menu Items for the context menu
 MenuItem item11 = new MenuItem("large");

 MenuItem item21 = new MenuItem("medium");
 MenuItem item31 = new MenuItem("small");
 contextMenu2.getItems().addAll(item11, item21,item31);
```

```
// textField.setContextMenu(contextMenu);
button1.setContextMenu(contextMenu1);
button2.setContextMenu(contextMenu2);
HBox layout = new HBox(20);
layout.setPadding(new Insets(15, 15, 15, 100));
layout.getChildren().addAll( button1,button2);
//Setting the stage
Scene scene = new Scene(new Group(layout), 595, 150, Color.BEIGE);
stage.setTitle("CustomMenuItem");
stage.setScene(scene);
stage.show();
}
public static void main(String[] args){
launch(args);
}
}
```

3. Screen Shots of Execution:

1. Problem Definition:

Q4. Write a JavaFX program that produces the following output when executed and displays Dialog Box 9 (as shown in Figure.2) on click of Register button (as shown in Figure.1):



2. Java Program:

```
package application;


/*JavaFX program that produces the following output when executed and displays Dialog Box
* Date:15-10-22
* USN:2SD20CS007
*/
import javafx.application.Application;
import javafx.geometry.Insets;
import  javafx.geometry.Pos;
import javafx.scene.control.Dialog;
import javafx.scene.control.DialogPane;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.CheckBox;
import javafx.scene.control.ChoiceBox;
```

```
import javafx.scene.control.DatePicker;
import javafx.scene.layout.BorderPane;
//import javafx.scene.control.Button;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.control.ButtonType;
import javafx.scene.control.Label;
//import javafx.scene.control.Label;
//import javafx.scene.control.ListView;
import javafx.scene.control.RadioButton;

import javafx.scene.layout.GridPane;
import javafx.scene.text.Text;
import javafx.scene.control.TextField;
import javafx.scene.control.ToggleGroup;
//import javafx.scene.control.ToggleButton;

import javafx.stage.Stage;

public class RegistrationForm extends Application {
 @Override
 public void start(Stage stage) {
 //Label for name
 BorderPane root = new BorderPane();
 stage.setTitle(" JavaFX Registration form");
 // label headerLabel = new Label("Registration Form");
 Label label = new Label("Employee Registration Form");
 // Object root;
```

```
root.setTop(label);
//root.setAlignment(label, Pos.CENTER);

Text nameLabel = new Text("Enter your Name");

//Text field for name
TextField nameText = new TextField();

//Label for date of birth


Text dobLabel = new Text("Enter Date of birth");

//date picker to choose date
DatePicker datePicker = new DatePicker();

//Label for gender
Text genderLabel = new Text("Enter your Gender");

//Toggle group of radio buttons
ToggleGroup groupGender = new ToggleGroup();
RadioButton maleRadio = new RadioButton("male");
maleRadio.setToggleGroup(groupGender);
RadioButton femaleRadio = new RadioButton("female");
femaleRadio.setToggleGroup(groupGender);


Text selectyourqualificationLabel = new Text("Select your qualification");
```

```
//check box for education
CheckBox ugCheckBox = new CheckBox("UG");
ugCheckBox.setIndeterminate(false);

//check box for education

CheckBox pgCheckBox = new CheckBox("PG");
pgCheckBox.setIndeterminate(false);
CheckBox phdCheckBox = new CheckBox("PhD");

phdCheckBox.setIndeterminate(false);


//Label for location
Text locationLabel = new Text("select your state");


//Choice box for location

ChoiceBox locationchoiceBox = new ChoiceBox();
locationchoiceBox.getItems().addAll
("Karnataka", "Tamilnadu", "Delhi", "Mumbai", "AP");

Button buttonRegister = new Button("Register");

//Creating a Grid Pane
GridPane gridPane = new GridPane();
```

```
//Setting size for the pane

gridPane.setMinSize(500, 500);

//Setting the padding
gridPane.setPadding(new Insets(10, 10, 10, 10));

//Setting the vertical and horizontal gaps between the columns
gridPane.setVgap(5);

gridPane.setHgap(5);

//Setting the Grid alignment
gridPane.setAlignment(Pos.CENTER);

//Arranging all the nodes in the grid
gridPane.add(nameLabel, 0, 0);
gridPane.add(nameText, 1, 0);

gridPane.add(dobLabel, 0, 3);
gridPane.add(datePicker, 1, 3);

gridPane.add(genderLabel, 0, 2);
gridPane.add(maleRadio, 1, 2);
gridPane.add(femaleRadio, 2, 2);
// gridPane.add(reservationLabel, 0, 3);
//gridPane.add(yes, 1, 3);
```

```
gridPane.add(selectyourqualificationLabel , 0, 5);
gridPane.add(ugCheckBox, 1, 5);
gridPane.add(pgCheckBox, 2, 5);
gridPane.add(phdCheckBox,3, 5);

gridPane.add(locationLabel, 0, 4);
gridPane.add(locationchoiceBox, 1, 4);



gridPane.add(buttonRegister, 1, 8);

//Styling nodes
buttonRegister.setStyle(
"-fx-font: normal bold 15px 'serif' " );

nameLabel.setStyle("-fx-font: normal bold 15px 'serif' ");
dobLabel.setStyle("-fx-font: normal bold 15px 'serif' ");
genderLabel.setStyle("-fx-font: normal bold 15px 'serif' ");

selectyourqualificationLabel.setStyle("-fx-font: normal bold 15px 'serif' ");

locationLabel.setStyle("-fx-font: normal bold 15px 'serif' ");



gridPane.setStyle("-fx-background-color: white;");
```

```
buttonRegister.setOnAction(e->{
// creating a dialog box
Dialog dialog = new Dialog();
dialog.setTitle("Registration Successful");
dialog.setHeaderText("Registration Status");
dialog.setContentText("Employee Registration is successful");

// adding image to the dialog box
// Image img = new Image("",50,50,true,true);
//ImageView imageview = new ImageView(img);

//
//dialog.setGraphic(imageview);

// adding button to the dialog box
dialog.getDialogPane().getButtonTypes().add(ButtonType.OK);
dialog.show();
});


Scene scene = new Scene(gridPane);


// stage.setTitle("Registration Form");

//Adding scene to the stage
stage.setScene(scene);
```

```
//Displaying the contents of the stage
stage.show();
}
public static void main(String args[]){
launch(args);
}
```

3. Screen Shots of Execution: