**Lab Exercise 4: GUI and Event Handling**

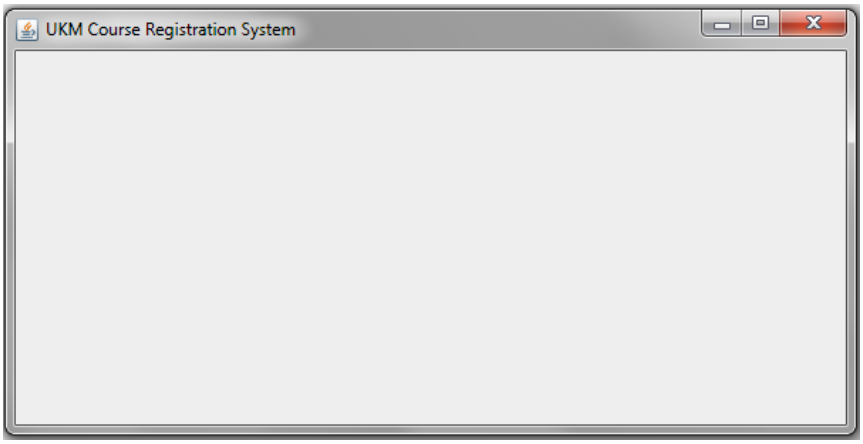
**A**

## Problem Description

Previously you had developed a Course Registration System for UKM. Although the system works perfectly as required, the system users had complained that the system's user interface is very dull and boring. So, now the client wants you to upgrade the system's user interface and make it attractive and more user-friendly.

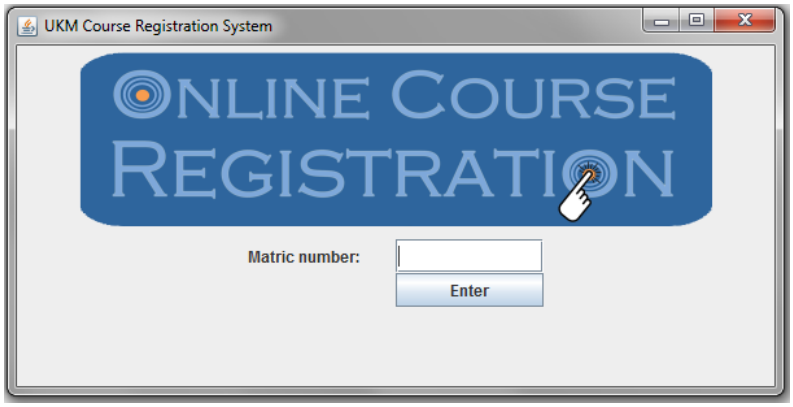
## Stage 1

1. Create a Java project named `Lab4-A-Stage1-Frame`.
2. Define a class named `LoginScreen` that will produce the following GUI.




## Stage 2

1. Create a Java project named `Lab4-A-Stage2-GUI1`.
2. Copy class `LoginScreen` from your Stage 1, and modify the class so that it will produce the following GUI.



Note:

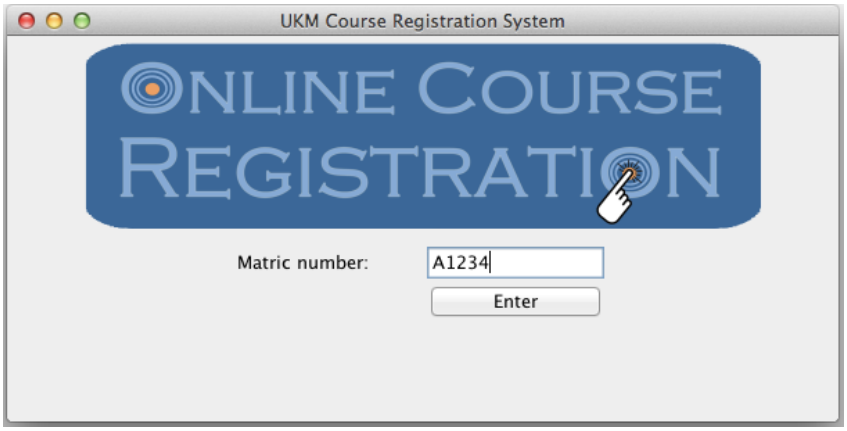
- You may refer to Java eclipse project "Lab4-Example1" in folder "examples" to see how to display image.
- The file of the image in the above GUI is in folder "image".
- At this stage you do not have to implement event handling – so, nothing happens when "Enter" button is clicked.

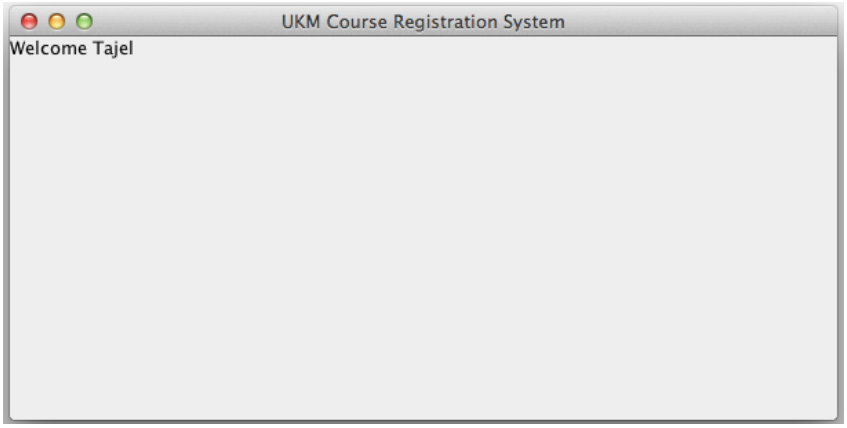
**Stage 3**

1. Create a Java project named `Lab4-A-Stage3-Event1`.
2. Copy class `Course`, `MainMenuScreen`, `Student` and `StudentRegister` from folder "stage3and4" into your Java project.
3. Copy class `LoginScreen` from your Stage 2.
4. Copy-and-paste the following line of code into class `LoginScreen` as instance variable.

```
private StudentRegister register = new StudentRegister();
```

5. Modify class `LoginScreen` so that when a valid matric number is entered (valid matric numbers are "A1234" and "A1001") and "Enter" button is clicked, `LoginScreen` will disappear and `MainMenuScreen` will appear, as shown below.

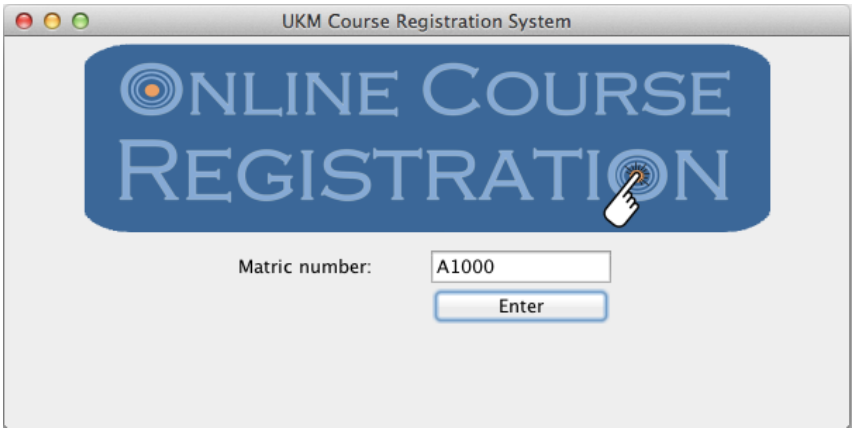


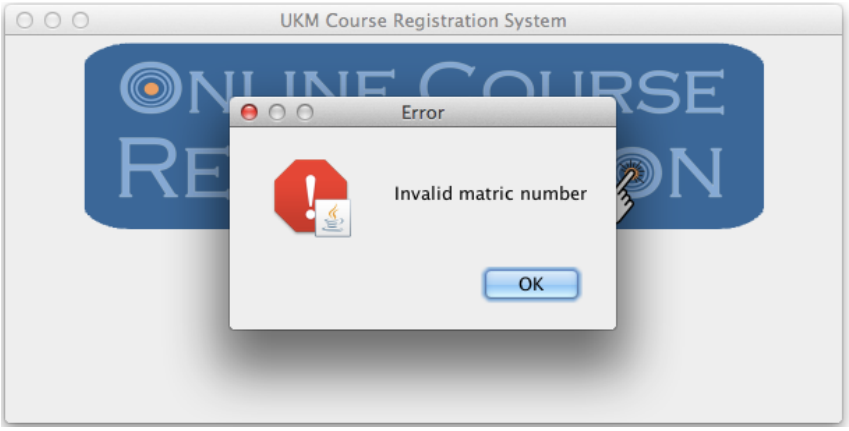



Note:

- You MUST use method `isRegisteredStudent` in class `StudentRegister` to check whether the input is a valid matric number.
- You MUST use method `getStudent` in class `StudentRegister` to get an object of `Student` associated with the entered matric number, and pass the `Student` object and `LoginScreen` object to (next frame) `MainMenuScreen`.
- You may refer to Java eclipse project "Lab4-Example2" in folder "examples" to see how to make current screen disappears and new screen apprears (i.e., moving from one screen to another).

6. Modify the class so that when invalid matric number is entered and "Enter" button is clicked, a pop-up error message will be displayed, as shown below.



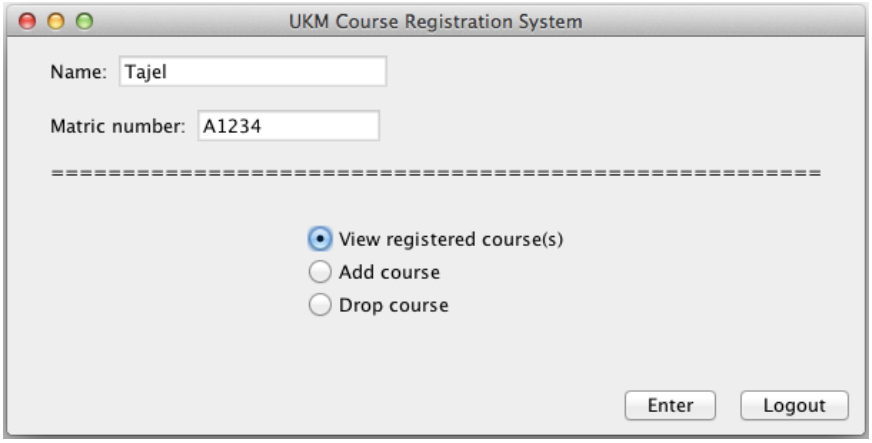


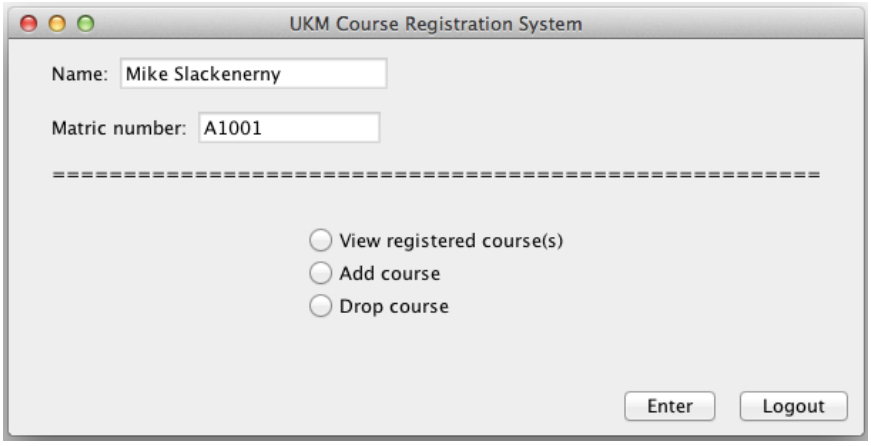

Note:

- You may refer to Java eclipse project "Lab4-Example2" in folder "examples" to see how to display pop-up error message.

**Stage 4**

1. Create a Java project named `Lab4-A-Stage4-GUI2`.
2. Copy class `Course`, `MainMenuScreen`, `Student` and `StudentRegister` from folder "stage3and4" into your Java project.
3. Copy class `LoginScreen` from your Stage 3.
4. Modify class `MainMenuScreen` so that it will produce the following GUI, after a valid matric number is entered (in this example, "A1234" or "A1001"), and "Enter" button in the previous screen is clicked.



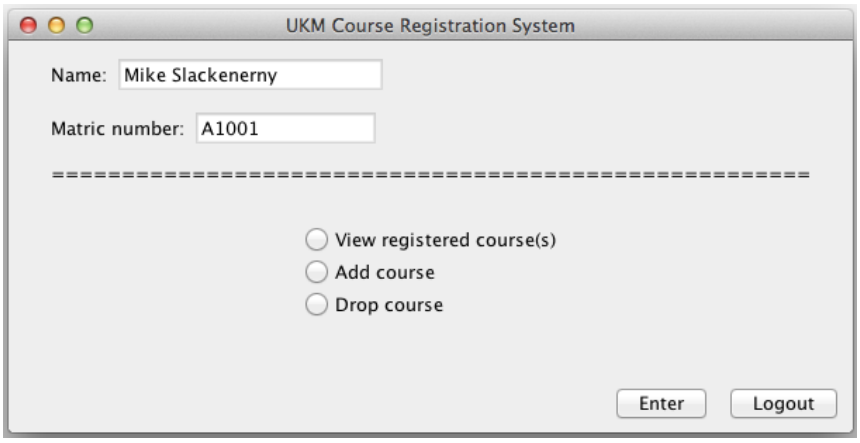


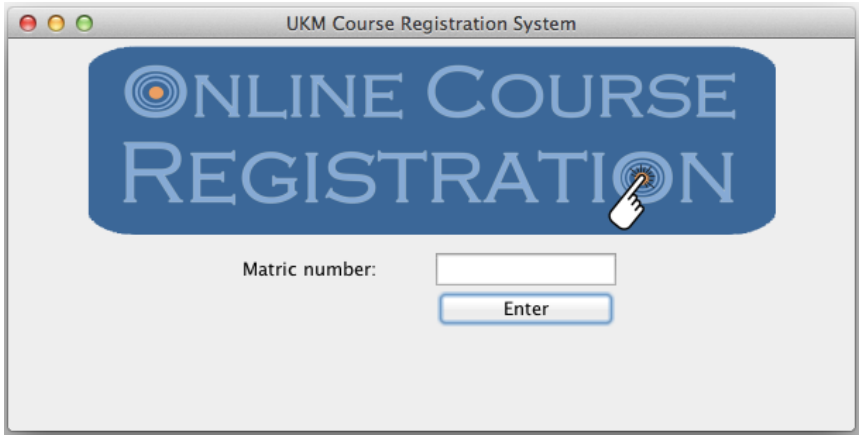

Note:

- Although it is not clear in the figures above, all text fields are actually not editable.
- Do not forget to set so that only one radio button can be selected at a time.
- You MUST use method `getName` in class `Student` to get the name of the student with a respective matric number.
- You do not have to implement event handling – so, nothing happens when "Enter" button or "Logout" button is clicked.

**Stage 5**

1. Create a Java project named `Lab4-A-Stage5-Event2`.
2. Copy class `AddCourseScreen`, `Course`, `DropCourseScreen`, `Student`, `StudentRegister` and `ViewCourseScreen` from folder "stage5" into your Java project.
3. Copy class `LoginScreen` and `MainMenuScreen` from Stage 4.
4. Modify class `MainMenuScreen` so that when "Logout" button is clicked, `MainMenuScreen` will disappear and `LoginScreen` will appear, as shown below.

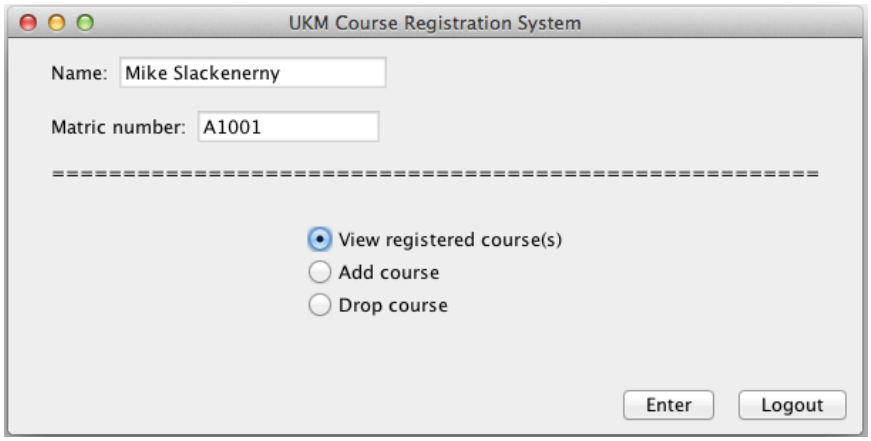


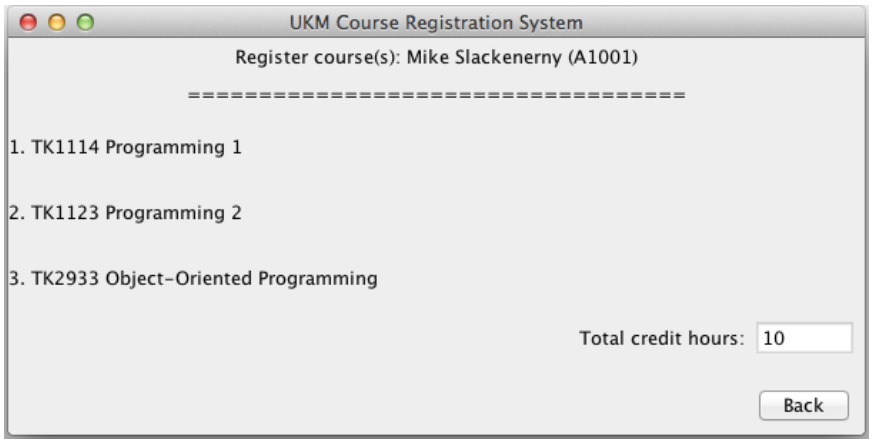



Note:

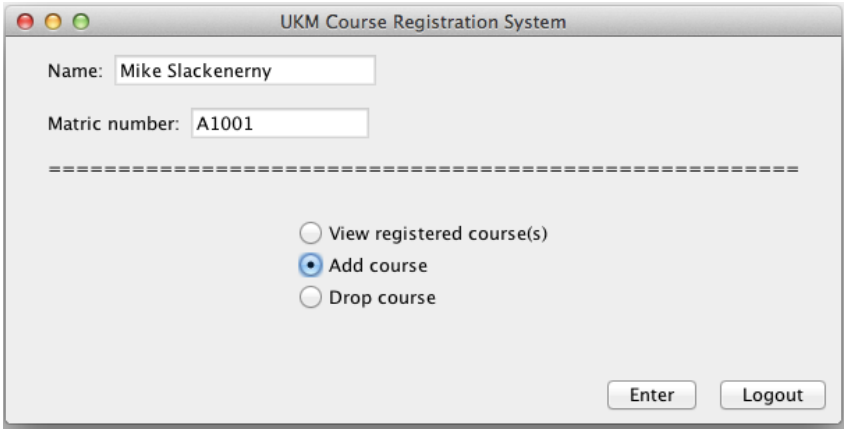
You may refer to Java eclipse project "Lab4-Example3" in folder "examples" to see how to return to previous screen.

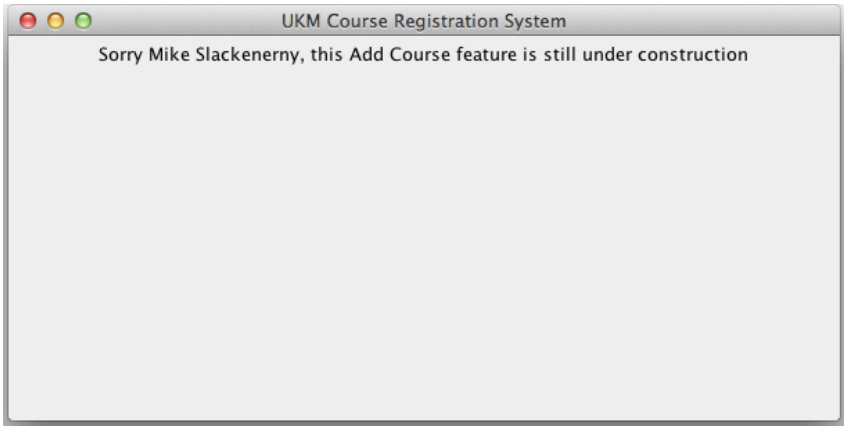
5. Modify class `MainMenuScreen` so that when radio button "View registered course(s)" is selected and "Enter" button is clicked, `MainMenuScreen` will disappear and `ViewCourseScreen` will appear, as shown below.



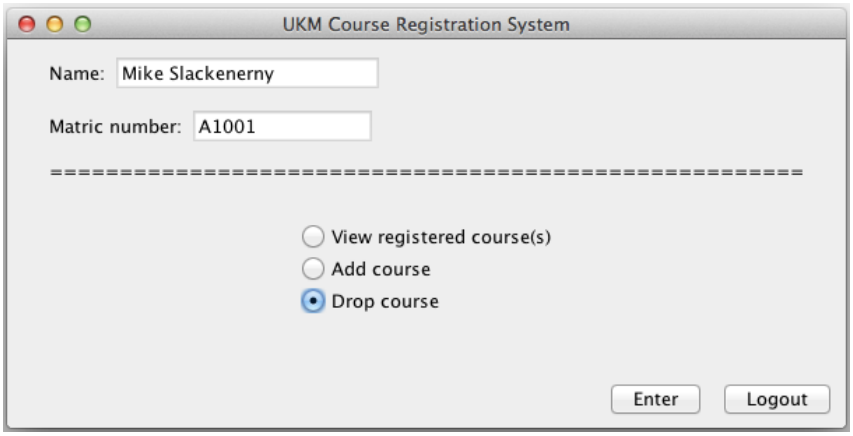



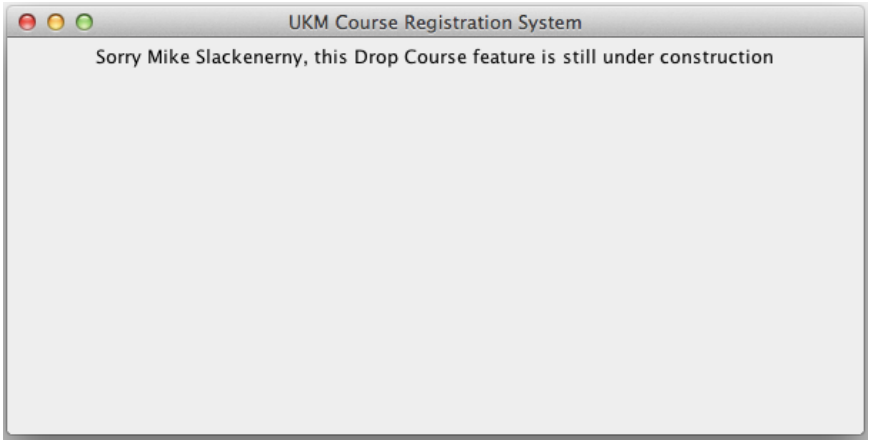
6. Modify class `MainMenuScreen` so that when radio button "Add course" is selected and "Enter" button is clicked, `MainMenuScreen` will disappear and `AddCourseScreen` will appear, as shown below.




7. Modify class `MainMenuScreen` so that when radio button "Drop course" is selected and "Enter" button is clicked, `MainMenuScreen` will disappear and `DropCourseScreen` will appear, as shown below.




UKM Course Registration System
Sorry Mike Slackenerny, this Drop Course feature is still under construction