

## COMP 1011: Advance Java Programming

### Assignment #4

**Due Date:** (Mid-night – 11.59 pm) 1st April

**Marks/Weightage:** 30/10%

**End Date:** (Mid-night – 11.59 pm) 4<sup>th</sup> April with 20% deduction/penalty. No exceptions.

**Purpose:** The purpose of this assignment is to:

- Practice the use of JDBC/JavaFx, SceneBuilder and MySql Database

**References:** Learning materials for chapter 24 of textbook, and other internet references (if any).

**IDE:** Eclipse Java Enterprise Developer edition – 2022-12/2023-03 and Windows 10/11 OS.

**Step1:** At the start of eclipse, you must name your Eclipse workspace according to the following rule:

YourFullName\_COMP1011SectionNumber\_A04

For Example: JohnSmith\_COMP1011Sec002\_A04 (if your section is Sec002)

**Step 2:** And after that, you must name your Eclipse project according to the following rule:

YourFullName\_COMP1011SectionNumber

For Example: JohnSmith\_COMP1011Sec002

**Step 3:** And your package should be named as follows:

YourFullName\_SectionNumber\_Ex01

For Example: johnsmith\_sec002\_ex01

#### Submission/Upload Instructions:

After you complete, run and test your code, you need to do the following:

- a) Close the eclipse, go to your **workspace** folder which you created in Step 1.
- b) Zip it up. You should get a zip file like this– JohnSmith\_COMP1011Sec002\_A04.zip. **You should only be submitting it in the .zip file format (and not .rar or any other format)**
- c) Upload this zip file using the **Assignment-04** link in blackboard.

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

- *variable names* start with a *lowercase* character for the first word and uppercase for every other word
- *classes* start with an *uppercase* character of every word
- **packages** use only *lowercase* characters
- *methods* start with a *lowercase* character for the first word and uppercase for every other word

**Note:** Late submissions are accepted until up to 3 days past due date with 20% deductions. After that no submission will be considered.

**Exercise 01:**

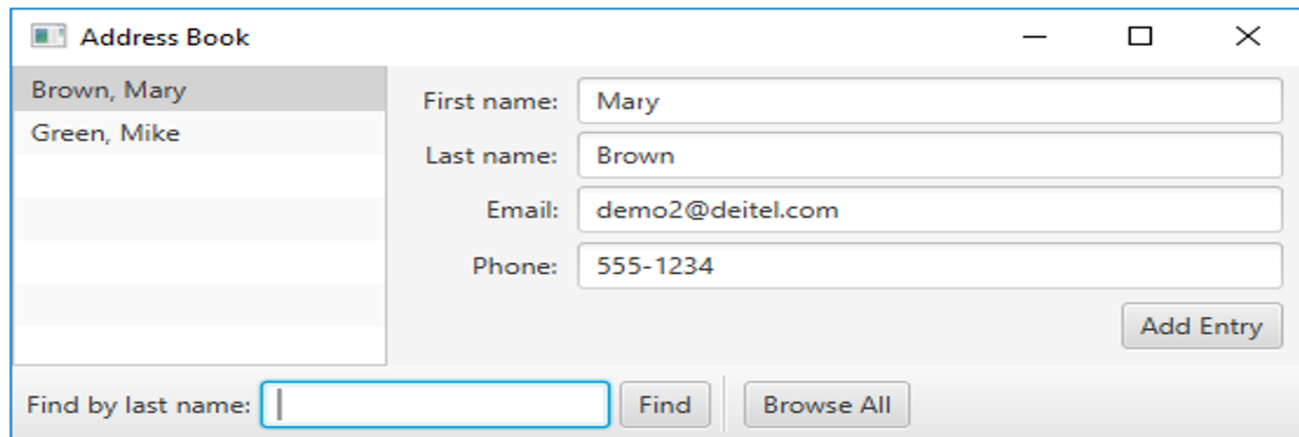
[5@6=30 marks]

Code examples are provided (refer the zip file - [addressbook\\_example.zip](#) in Blackboard) and covered in the class.

You need to create Address Book database, addressbook.sql file is included in the zip file. Also provided the .fxml file for GUI. You are required to implement the following as shown in the screen shot.

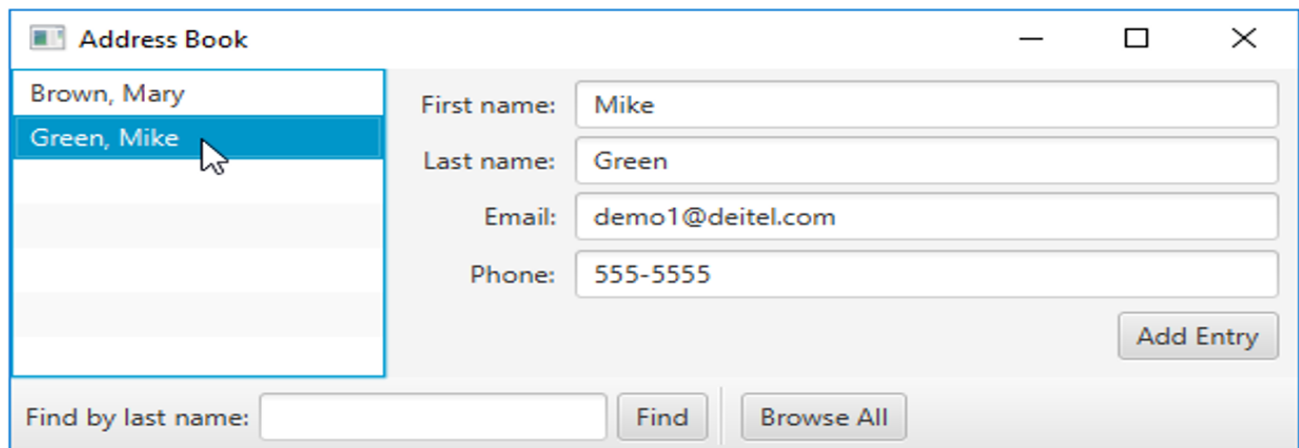
For the database, if you want to use Oracle, you can use that instead of MySQL.

a) Initial Entries in the database



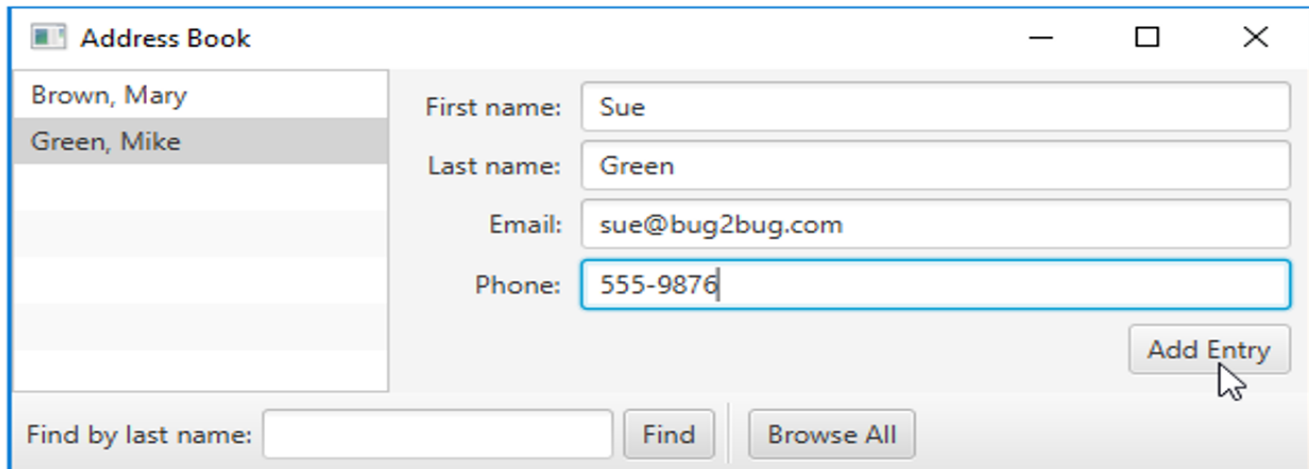
The screenshot shows a Java Swing window titled "Address Book". On the left is a list box containing two entries: "Brown, Mary" (selected) and "Green, Mike". To the right of the list box are four text input fields: "First name:" with "Mary", "Last name:" with "Brown", "Email:" with "demo2@deitel.com", and "Phone:" with "555-1234". Below these fields is an "Add Entry" button. At the bottom of the window is a "Find by last name:" text field, a "Find" button, and a "Browse All" button.

b) Viewing the entry for Green Mike



The screenshot shows the same "Address Book" window. The list box now has "Green, Mike" selected, and a mouse cursor is pointing at it. The text input fields on the right now display the details for "Green, Mike": "First name:" is "Mike", "Last name:" is "Green", "Email:" is "demo1@deitel.com", and "Phone:" is "555-5555". The "Add Entry" button is still present. The "Find by last name:" field and "Find" and "Browse All" buttons are also visible at the bottom.

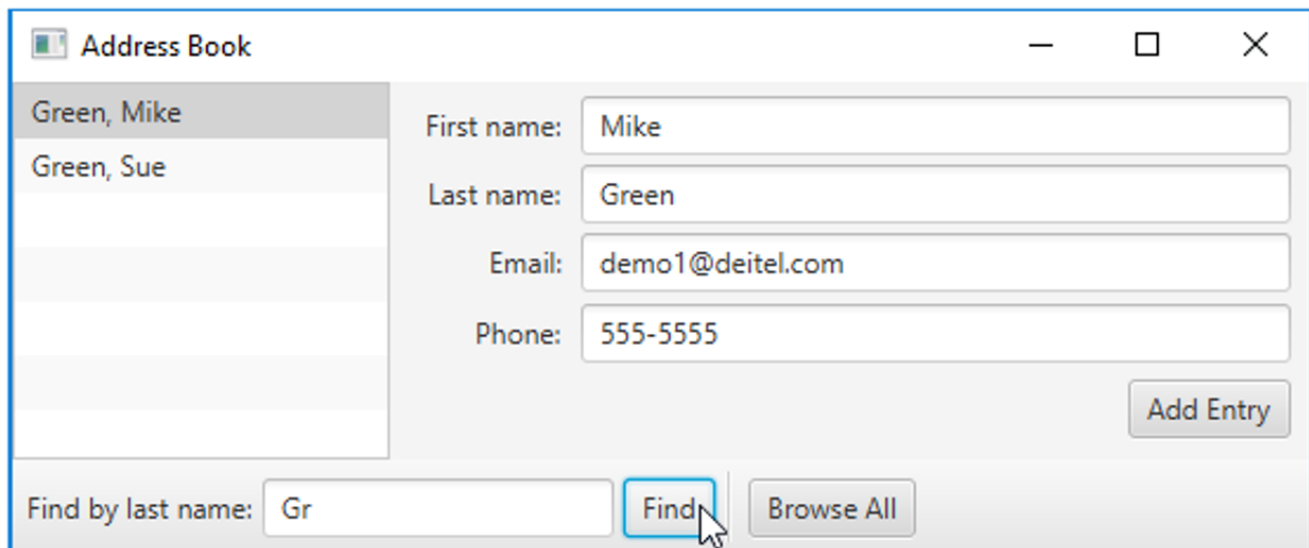
c) Adding the entry for Sue Green



The Address Book window displays a list of contacts on the left and a form for editing an entry on the right. The list contains 'Brown, Mary' and 'Green, Mike'. The form is filled with Sue Green's information: First name: Sue, Last name: Green, Email: sue@bug2bug.com, and Phone: 555-9876. The 'Add Entry' button is highlighted with a mouse cursor.

Find by last name:	Find	Browse All

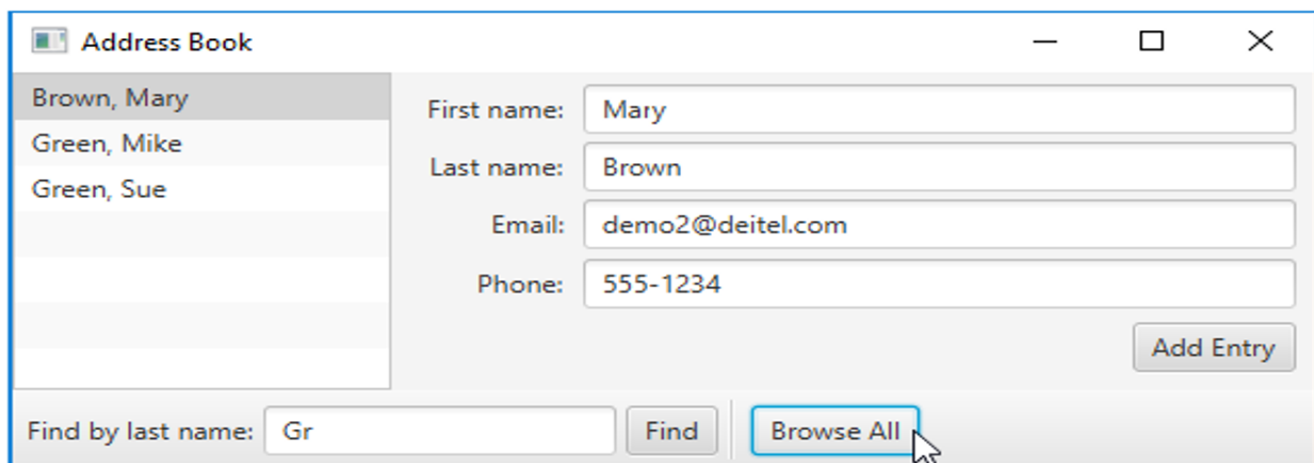
d) Searching the entry for Gr



The Address Book window shows the search results for 'Gr'. The list on the left now contains 'Green, Mike' and 'Green, Sue'. The form on the right displays Mike Green's information: First name: Mike, Last name: Green, Email: demo1@deitel.com, and Phone: 555-5555. The 'Find' button is highlighted with a mouse cursor.

Find by last name:	Find	Browse All
Gr		

e) Returning the complete list by clicking the Browse All



The Address Book window shows the complete list of contacts. The list on the left contains 'Brown, Mary', 'Green, Mike', and 'Green, Sue'. The form on the right displays Mary Brown's information: First name: Mary, Last name: Brown, Email: demo2@deitel.com, and Phone: 555-1234. The 'Browse All' button is highlighted with a mouse cursor.

Find by last name:	Find	Browse All
Gr		

**Evaluation/Rubric:**

<b>Functionality</b>	
(a) Correct implementation of business requirements, GUI, code logic, functionality, getting data from database etc.	90%
(b) Correct implementation of exception handling and intended results	5%
(c) Comments, correct naming of variables, methods, classes, etc.	2.5%
<b>(d) Friendly input/output</b>	2.5%
<b>Total</b>	100%