25.04.2022

48024 Applications Programming Assignment 2

Topics:

OO design, GUIs, MVC, tables, lists

Learning Outcomes:

This assignment supports objectives 3 - 5

Due date:

17th May 2021 - 11:59PM (Tuesday Week 12)

Weight:

25%

Individual Work

All work is individual. You must write every line of code yourself except for code copied from study module sample code, lecture sample code, tutor demos or lab code.

In most cases, you may discuss ideas, approaches and problems. However, if an assignment task is labeled as “Advanced”, you must not discuss ideas, approaches and problems. Advanced tasks are designed to test your ability to think on your own.

You **MUST NOT** let another student see your solution code, and you **MUST NOT** look at another student’s solution code. Sharing your code on public forums such as the discussion board, or Internet forums such as stackoverflow.com is not permitted. More information about Academic Misconduct can be found at:

http://www.gsu.uts.edu.au/rules/student/section-16.html

Skeleton Code

As a starting point for this assignment, you must use the skeleton code provided in Assignment 2 Description. There are two options for Java, a NetBeans version, which includes the structure required to import directly into NetBeans, and a plain version for other IDEs (which tend to be less rigid about their import requirements). **You are NOT allowed to modify the model or remove any of the starting classes or FXML files.**

Expected workload

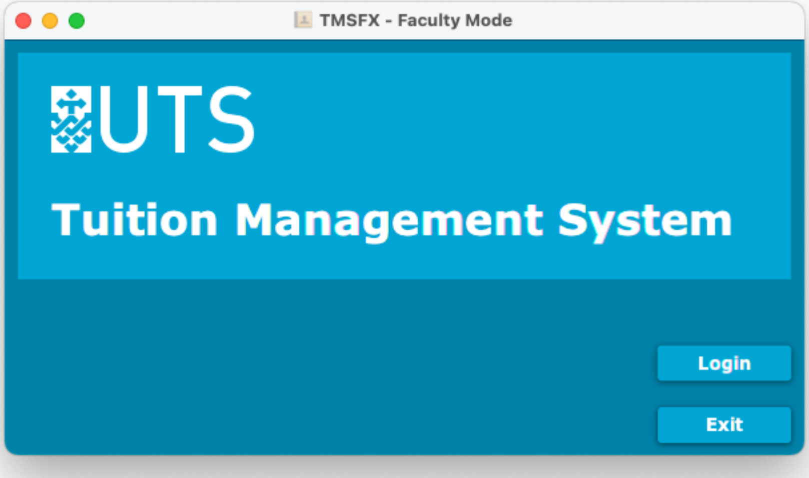
The time to do the assignment to a distinction level (i.e. a mark between 75% to 84%) has been estimated at 25 hours for a student of average ability who has completed all the tutorial and lab exercises.

Specification

The specification is presented in several parts. In this document is given a series of screen shots and textual descriptions for visual reference.

A demonstration video is also presented at https://canvas.uts.edu.au/courses/22120/assignments/91790/. This demonstration video is considered part of the specification and contains important details about the dynamic function of the assignment.

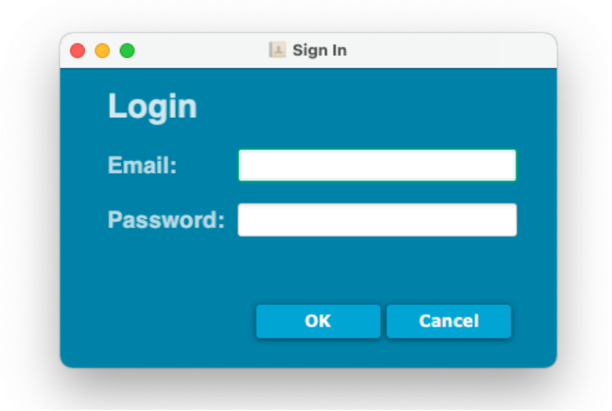
Main Menu (8%)



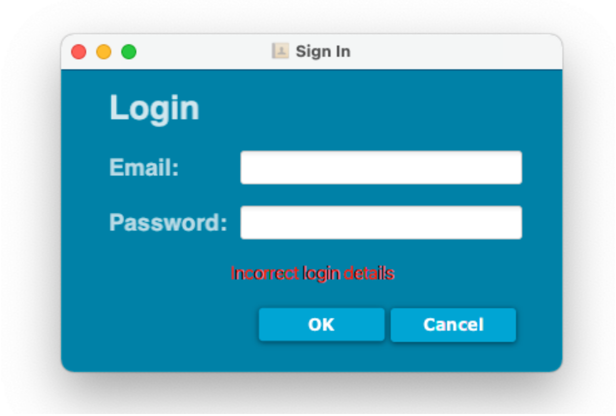
The main menu is opened when the application launches. It has "Login" button to access the Login window. It also has an exit button which shuts the entire application down. The main menu has a header section with an UTS logo and the title "Tuition Management System". The title of the window includes a small icon and the tile "TMSFX – Faculty Mode".

Login Window (12%)

The login window is launched from the main menu by opening a separate window. It has OK button to match the email and password for the user. The Cancel button closes the login window. The login window includes a same icon as the main menu and the title is set as "Sign In".

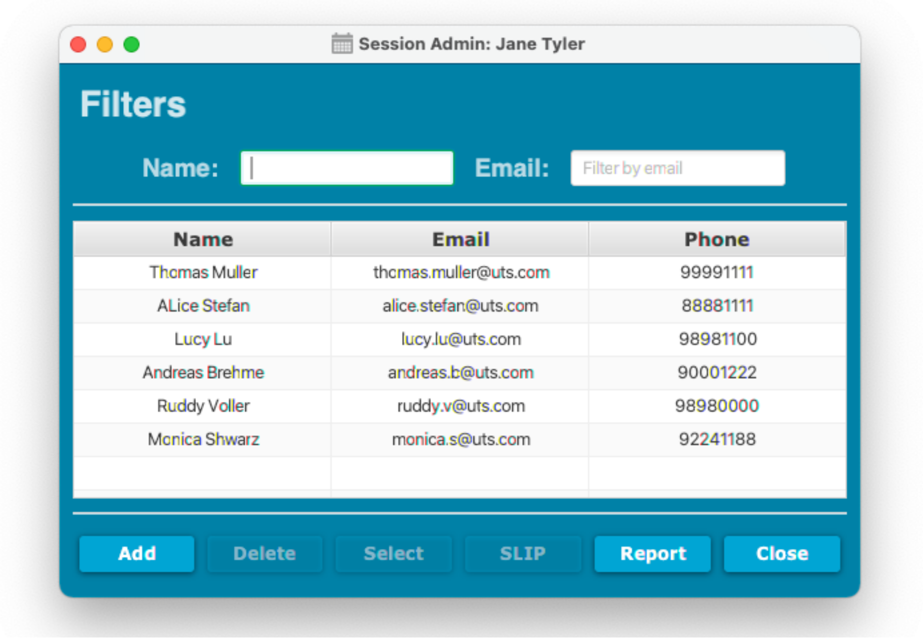


The OK button will open the admin window if the combination of the email and password is correct. If the combination is not correct, the error message “Incorrect login details” will show up. The TextField is reset to blank after click "OK".



Incorrect login details

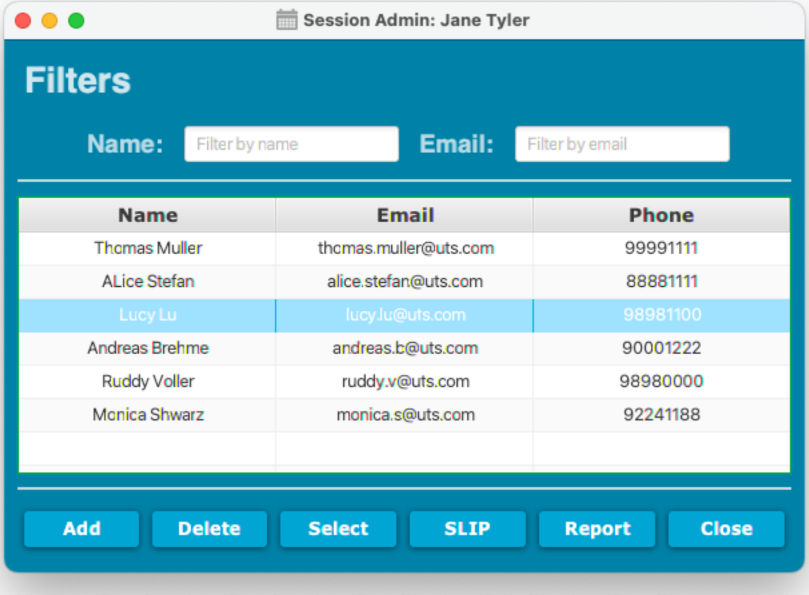
Admin Window (20%)



Name	Email	Phone
Thomas Muller	thomas.muller@uts.com	99991111
ALice Stefan	alice.stefan@uts.com	88881111
Lucy Lu	lucy.lu@uts.com	98981100
Andreas Brehme	andreas.b@uts.com	90001222
Ruddy Voller	ruddy.v@uts.com	98980000
Monica Shwarz	monica.s@uts.com	92241188

The administration window is launched from the Login window by opening a separate window. The administration window displays the student's information by a table with 3 columns "Name", "Email" and "Phone".

The administration window has buttons to open the student window in a new window to add and update students, and has buttons to open the windows to list the tuition slip and report. Only "Add", "Report" and "Close" buttons are enabled initially.



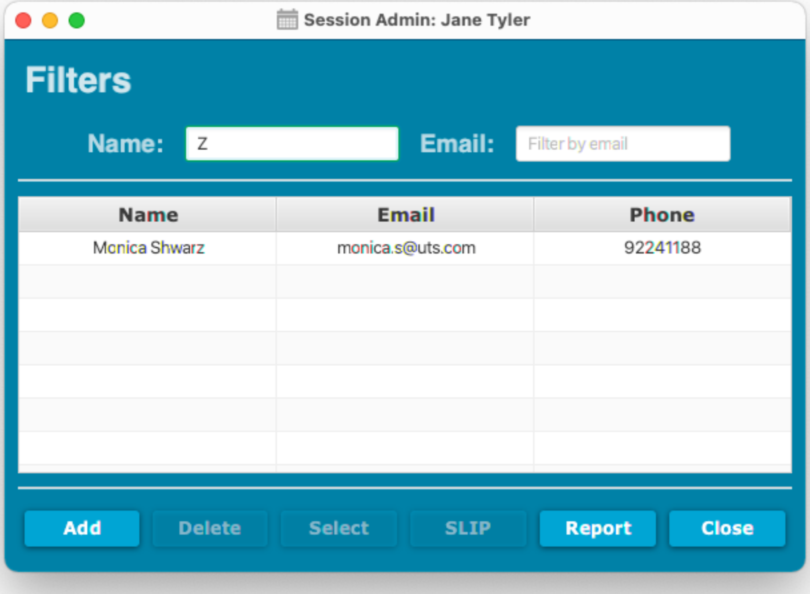
Name	Email	Phone
Thomas Muller	thomas.muller@uts.com	99991111
ALice Stefan	alice.stefan@uts.com	88881111
Lucy Lu	lucy.lu@uts.com	98981100
Andreas Brehme	andreas.b@uts.com	90001222
Ruddy Voller	ruddy.v@uts.com	98980000
Monica Shwarz	monica.s@uts.com	92241188

When click on one student record in the table, the selected record is highlighted and the "Delete" "Select" "SLIP" buttons are enabled. Click the "Delete" button will remove the selected record from the table. The "Delete" "Select" and "SLIP" buttons will become disabled after the removal.

The Close button closes the administration window. The administration window includes a "faculty" icon. The title of the window is "Session Admin" plus the name of the login user.

Admin Filter (12%)

On top of the administration window, there are two independent filters, which display the students with matched name or email. The filters work asynchronously with non-case sensitive partial matching function. Either name filer works, or email filter works, not both. For example:



Session Admin: Jane Tyler

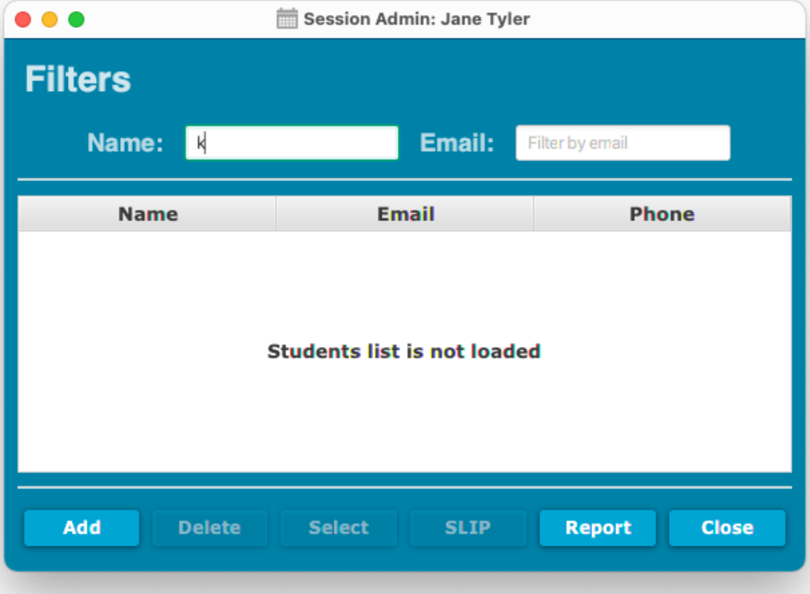
Filters

Name: Z

Email: Filter by email

Name	Email	Phone
Monica Shwarz	monica.s@uts.com	92241188

The student list prints "Students list is not loaded" when there is no matches result. The default input texts "Filter by name" and "Filter by email" are displayed when there is no input; The default input texts are disappeared when the input starts.



Session Admin: Jane Tyler

Filters

Name: k

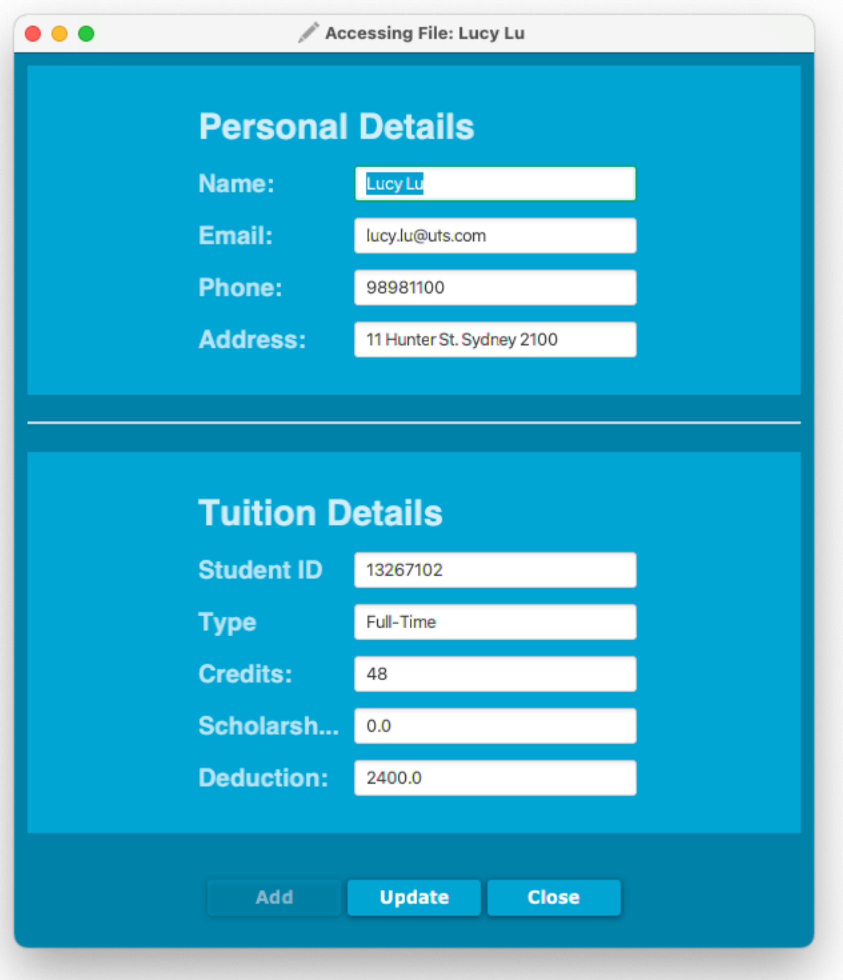
Email: Filter by email

Name	Email	Phone

Students list is not loaded

Student Window (12%)

The student window is launched from the administration window by clicking “Select” for a student record. The student window displays the information of “Personal Details” and “Tuition Details” for an existing student.



Accessing File: Lucy Lu

Personal Details

Name: Lucy Lu

Email: lucy.lu@uts.com

Phone: 98981100

Address: 11 Hunter St. Sydney 2100

Tuition Details

Student ID 13267102

Type Full-Time

Credits: 48

Scholarsh... 0.0

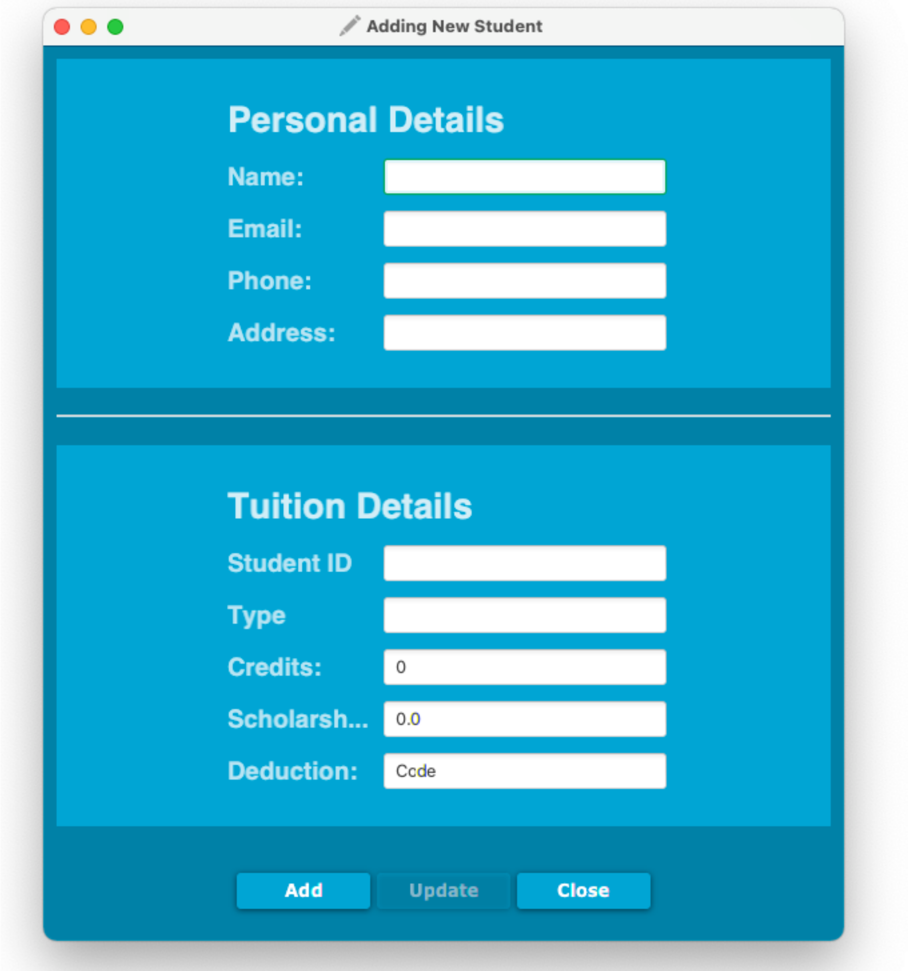
Deduction: 2400.0

The student window has buttons to update the student's information and has close button to close the student window. Only "Update" and "Close" buttons are enabled.

Update button opens the error window when the input is not valid. The button closed the window when the input is valid or there is not input. The input validation is described in validator.java and associated with the error window. Meanwhile, Admin window immediately updates the student information.

The student window includes an “edit” icon. The title of the window is “Accessing File: ” plus the name of the student.

Student Window Add (12%)



Adding New Student

Personal Details

Name:

Email:

Phone:

Address:

Tuition Details

Student ID

Type

Credits: 0

Scholarsh... 0.0

Deduction: Code

The student window is launched from the administration window by clicking “Add”. The student window displays the empty information, waiting for the input of “Personal Details” and “Tuition Details” for a new added student.

The student window has buttons to add and update student’s information, and has close button to close the student window. Only "Add" and "Close" buttons are enabled.

Add button opens the error window when the input is not valid. Add button closed the window when the input is valid. The input validation is described in validator.java and associated with the error window. Meanwhile, Admin window immediately updates the student list.

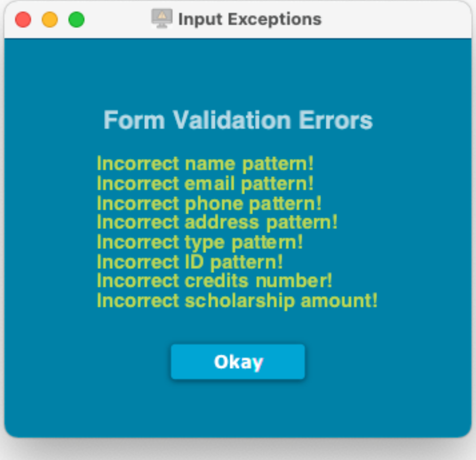
The student window includes an “edit” icon. The title of the window is “Adding New Student”.

Error Window (12%)

The error window is popped up when the input is not valid for a student record. The error window displays the input error messages based on the input validation described in validator.java.

- Name Pattern: First name and Last name are two strings with capital letters.
- Email Pattern: Domain name is appended with@ at the end of the address.
- Phone Pattern: 8 digits number
- Address Pattern: Started and ended with number, separated from the street name by space

- Type Pattern: A hyphen concatenates two strings with capital letters.
- ID Pattern: 8 digits
- Credits Pattern: a multiple number of 6
- Scholarship Pattern: a number between 0 and 100000(inclusive)
- Deduction Pattern: The input string will be compared with “2022AUT” for deduction calculation. Directly amount input will result 0.00 deduction.



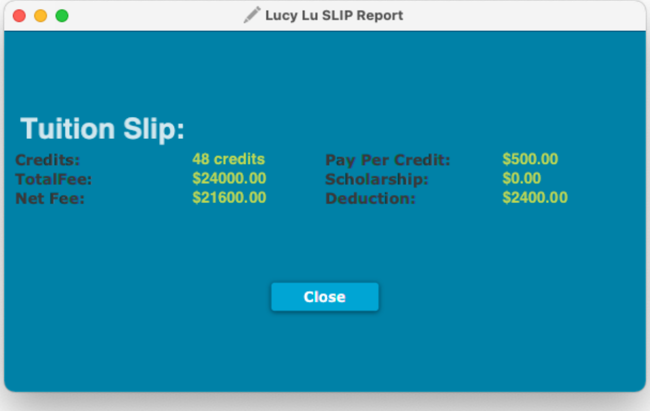
Form Validation Errors

Incorrect name pattern!
Incorrect email pattern!
Incorrect phone pattern!
Incorrect address pattern!
Incorrect type pattern!
Incorrect ID pattern!
Incorrect credits number!
Incorrect scholarship amount!

The error window has “Okay” button to close the window. The title of the window includes an “error” icon and the name is “Input Exceptions”.

Slip Window (8%)

The slip window is launched from the administration window by clicking “SLIP”. The slip window displays the tuition information for the selected student.



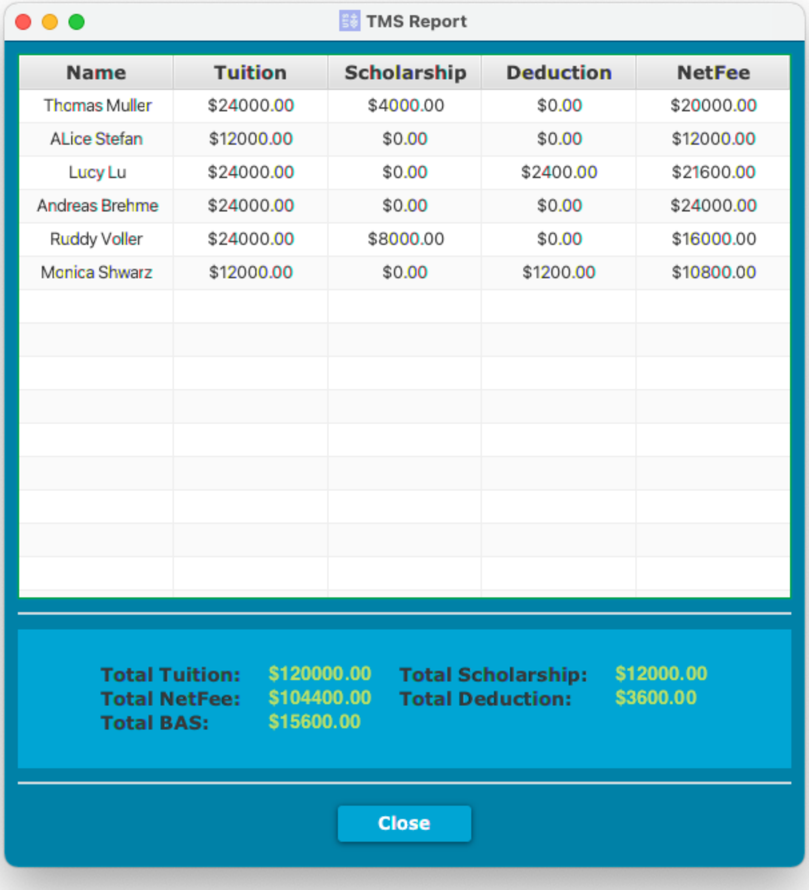
Tuition Slip:

Credits:	48 credits	Pay Per Credit:	$500.00
TotalFee:	$24000.00	Scholarship:	$0.00
Net Fee:	$21600.00	Deduction:	$2400.00

The slip window has “Close” button to close the window. The title of the window includes an “edit” icon and the name is the student’s name plus “SLIP Report”.

Report Window (12%)

The report window is launched from the administration window by clicking “Report”. The report window displays the collective tuition information for all the students.



Name	Tuition	Scholarship	Deduction	NetFee
Thomas Muller	$24000.00	$4000.00	$0.00	$20000.00
ALice Stefan	$12000.00	$0.00	$0.00	$12000.00
Lucy Lu	$24000.00	$0.00	$2400.00	$21600.00
Andreas Brehme	$24000.00	$0.00	$0.00	$24000.00
Ruddy Voller	$24000.00	$8000.00	$0.00	$16000.00
Monica Shwarz	$12000.00	$0.00	$1200.00	$10800.00

Total Tuition: $120000.00 Total Scholarship: $12000.00
Total NetFee: $104400.00 Total Deduction: $3600.00
Total BAS: $15600.00

The report window has “Close” button to close the window. The title of the window includes an “UTS” icon, and the name is “TMS Report”.

Requirements

Layout

To get full marks, you should layout your windows to look as close as possible to the screenshots. This means that you should try to duplicate the spacing between and around nodes that is shown in the screenshots, and the width and height of the nodes, and the alignment of the nodes.

Style

A CSS file is included in the skeleton code which provides all the styles used in the assignment. You don’t need to modify the CSS file.

Code

Your solution must satisfy the following code requirements:

- Your solution must follow the MVC architecture.

- Your solution must keep the package structure and class names that were provided in the skeleton code.

- The models must notify the views of changes by correctly applying the JavaFX property patterns and observable lists. Model data that can change must be observable. Model data that never changes need not be observable.

- The views must be laid out in FXML.

Submission to ED

You will need to submit the source code to ED. The marker will check the source code to make sure all the requirements have been met. Missing the source code will result 10 marks deduction of the assignment 2.

Submission to Canvas

Your assignment should be submitted as a JAR file that includes:

- All FXML, CSS and image files required to run your assignment.
- The binary files(.class) of the runnable jar file.

You can check the JAR file on a lab PC to make sure it works.

The due date won't be extended unless under extenuating circumstances. Submission extension before the last lab class will be granted automatically with a late penalty. An extension CANNOT be given after the due date. For any extension beyond one week or extension without penalty, you will need to submit a Special Consideration following the Special Consideration process for special consideration for reasons including unexpected health, family, or work problems.

Marking

You will be marked by the Canvas submission for the features that can be demonstrated to work. Your tutor will ask questions based on your demonstration in the last lab class. Aside from marks for the functionality, the marker will also check your code to ensure that all code requirements have been met. Your final mark will be a combination of marks for functionality, marks for code (See Rubric table on Canvas) and moderation marks for answering questions.

Marking the code and analyzing spoofing, cheating and plagiarism is done in the two weeks following the due date. If you are suspected of Academic Misconduct, I will forward your case to the Misconduct Committee and your mark will end with ".1". Your mark will be finalized within 2 weeks of the due date.

Online support

A FAQs (Frequently Asked Questions) page will be posted in the Assignment page on Canvas. If you have a question, check the FAQ page first, it may already be answered there. You should read the FAQ at least once before you hand in your solution, but to be safe check it every couple of days. Anything posted on the FAQ is considered to be part of the assignment specification.

If anything about the specification is unclear or inconsistent, check the FAQ first, then contact the subject coordinator who will try to make it clearer by replying to you directly and posting the common questions and answers to the FAQ. This is similar to working on the job, where you ask your client if you are unsure what has to be done, but then you write all the code to do the task.

The preferred way to ask assignment related questions is to participate in the lectures, lab activities, UPASS, and consultation sessions. Students are expected to seek help through the following steps:

Step 1: Check the FAQ
Step 2: Ask peers or via the group discussion board
Step 3: Ask your tutor or mohammed.jaddoa@uts.edu.au
Step 4: Forward to the subject coordinator

The Subject Coordinator may be contacted by email if you have matters of a personal nature to discuss, e.g., illness, personal issues or other matters of importance. All emails sent to subject coordinators, tutors or lecturers must have a clear subject line that states the subject number followed by the subject of the email [e.g. Subject 48024, Appointment Request], and must be sent from your UTS email address.

Requests for appointments outside the given consultation hours may be arranged where circumstances require, and to do so please contact the subject coordinator by email.