

Faculty of Science

**Course**: CSCI 2020U: Software Systems Development and Integration

**Lab:** #5

**Topic:** User Interfaces 2

**Overview**

In this lab, you’ll develop a Java program with a table user interface that shows student marks, using JavaFX. It is recommended, but not required, that you use IntelliJ for this lab.

**Instructions**

You can use any operating system or environment for this laboratory assignment.

You will create a new directory (or IntelliJ IDEA project) called lab05. The first user interface demo might serve as a useful starting point, since this lab is similar. Use the following steps to complete this lab:

1. Create a class, called StudentRecord, which represents one row of data for our student. A StudentRecord should contain the following fields:
   * Student ID
   * Midterm
   * Assignments
   * Final exam
   * Final Mark
   * Letter Grade

The final mark will be calculated as a weighted average of the assignments (20%), midterm (30%), and final exam (50%).

The letter grade will be determined from the following table:

|  |  |  |
| --- | --- | --- |
| Minimum | Maximum | Letter Grade |
| 80 | 100 | A |
| 70 | 79 | B |
| 60 | 69 | C |
| 50 | 59 | D |
| 0 | 49 | F |

1. Use the following DataSource class that generates some sample StudentRecord instances:

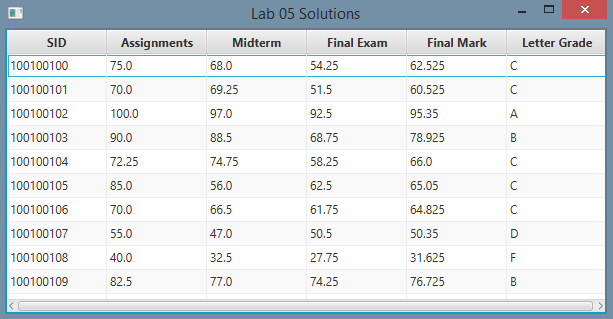
public class DataSource {  
 public static ObservableList<StudentRecord> getAllMarks() {  
 ObservableList<StudentRecord> marks = FXCollections.*observableArrayList*();

// Student ID, Assignments, Midterm, Final exam  
 marks.add(new StudentRecord("100100100", 75.0f, 68.0f, 54.25f));  
 marks.add(new StudentRecord("100100101", 70.0f, 69.25f, 51.5f));  
 marks.add(new StudentRecord("100100102", 100.0f, 97.0f, 92.5f));  
 marks.add(new StudentRecord("100100103", 90.0f, 88.5f, 68.75f));  
 marks.add(new StudentRecord("100100104", 72.25f, 74.75f, 58.25f));  
 marks.add(new StudentRecord("100100105", 85.0f, 56.0f, 62.5f));  
 marks.add(new StudentRecord("100100106", 70.0f, 66.5f, 61.75f));  
 marks.add(new StudentRecord("100100107", 55.0f, 47.0f, 50.5f));  
 marks.add(new StudentRecord("100100108", 40.0f, 32.5f, 27.75f));  
 marks.add(new StudentRecord("100100109", 82.5f, 77.0f, 74.25f));  
  
 return marks;  
 }  
}

1. Create a table that will show a list of StudentRecord objects, using our DataSource data

***Note****: This table user interface can use FXML or you can define it programmatically*

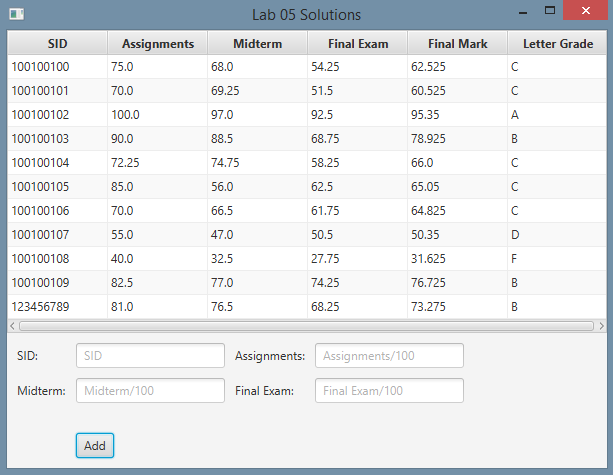
***Note****: See figure 1 for an example of the final product.*

****

**Figure 1: The running application, showing the marks table**

**Extra Challenge (Optional)**

Add a button (Add) and some text fields below for entering the Student ID, midterm, assignment, and final exam marks. Calculate the remaining values just like in step 3. When Add is pressed, these values get added to the table.

****

**Figure 2: The application, showing the fields below for adding new student records**

**How to Submit**

Show your running application to the TA to prove that you have finished this lab. The TA can provide oral feedback if you do not receive full marks for any lab assignment, but it is most appropriate to ask the TA for this feedback in a timely fashion (i.e. ask now, not at the end of the term).