Seneca College

Applied Arts & Technology

SCHOOL OF COMPUTER STUDIES

Workshop 3

Due Date: March 8th, 2024

INSTRUCTIONS

- *This workshop must be completed individually without any outside collaboration. All work must be your own. Copying or reproducing the work done by others (in part or in full) or letting others to copy or reproduce your own work is subject to significant grade reduction or getting no grade at all and/or being treated as academic dishonesty under the College's Academic Dishonesty Policy.*

- *Your goal is to finish the design part (using the Scene Builder is optional if you want to use it) during the lab time, unless otherwise specified by your instructor.*

- *The backend coding for your design can be submitted as a part of DIY.*

- *Your application must compile and run upon download to receive any mark.*

- *To submit the workshop, please follow the Submission Guideline provided at the end of this document.*

- *You must submit your workshop by the due date. Late submissions policy is specified in the Academic Procedures for Evaluations document available through the class plan on Blackboard.*

Description:
The third workshop lets the student practice about designing a simple JavaFx screen using Scene builder, making properties, bindings, and listeners.

Task:
Typically, banks offer car loans for periods ranging from one to eight years (12 to 96 months).

Borrowers repay the loans in weekly/ Bi-weekly/ monthly installments. The amount of each weekly/ Bi-weekly/ monthly payment is based on the length of the loan, the amount borrowed, down payment and the interest rate.

Create an app that allows the customer to enter the type of the vehicle, age of the vehicle, price of a car, the down-payment amount, the loan's monthly interest rate and the frequency.

Use a Slider control for the years of the car loan with a minimum value of 12 and a maximum value of 96 months. Each tick mark should designate a 12-month period (e.g., 12, 24, 48 etc. months options).

The app should display the loan's duration in months and the monthly payments, formatted as currency.

1. Create the following JavaFX user interface, making use of a HBox and GridPane layouts will be easy, or you can also use AnchorPane.
2. The view needs to show,
    a. Type of Vehicle:
        i. Car
        ii. Truck
        iii. Family Van
    b. Age of Vehicle:
        i. New
        ii. Used
    c. Price of the Vehicle.

d. Down payment.
e. Interest rate.
f. Loan payment period: Should be a slider min value 12 month max 96 month. Slider ticks should be monthly.
g. Laon payment frequency
    i. Weekly
    ii. Bi-weekly
    iii. Monthly
h. Your estimated fixed rate loan payment: Format in dollar amount. Should also print if the payment is monthly/ bi-weekly/ weekly depending on what user chose.
i. Clear button.
j. Calculate button.
k. Saved rates button: upon pressing this the current rate should be saved in an appropriate data structure of your choice.
l. Show saved rates button: upon clicking a new window/ dialog should open and display the saved rates as a ListView and upon selecting from the list should show the saved rate.

Note: save rate are only available as long as application is running upon closing the application it should clear all the rates and the list should be empty.

- Implement the EventHandlers for buttons.
- Implement the ChangeListener for the slider.

More discussion on design and solution will be done during the lab.

Workshop Header

/***
Workshop #
Course:*<subject type> - Semester*
Last Name:*<student last name>*
First Name:*<student first name>*
ID:*<student ID>*
Section:*<section name>*
This assignment represents my own work in accordance with Seneca Academic Policy.
Signature
Date:*<submission date>*
***/

Code Submission Criteria:

Please note that you should have:

- Appropriate indentation.
- Proper file structure
- Follow java naming convention
- Do Not have any debug/ useless code and/ or files in the assignment

Deliverables and Important Notes:

All these deliverables are supposed to be uploaded on the blackboard once done.

- Design should be discussed/ created during the lab. 10%.
- Complete the code behind as the part of your DIY. 60%
- Submit a **reflect.txt** file with the submission. 10%
  Questions to be answered for the reflection:
  - Class Design
    - Why you have single class or multiple classes in your design.
  - Data structure
    - Why choosing your data structure (used in the solution).
    - Compare your chosen data structure with at least one more data structure.
      - Your comparison should be talking about performance.
- Video submission explaining core code pointers and showing the full working application (3 – 8 minutes max). 20%
- All submission goes to Black Board.
- Your submission should include

    - Video file with audio
    - Reflect.txt file
    - Complete zipped project.
- Late submissions would result in additional 10% penalties for each day or part of it.

Remember that you are encouraged to talk to each other, to the instructor, or to anyone else about any of the workshops, but the final solution may not be copied from any source.