Lab 7 Assignment – 3

Team Name:

Runtime Error

Team Members’ names:

Yash Bhatia

Akshay Makhija

Akash Chhabria

What is the problem you plan to solve?

In universities, managing and maintaining academic tasks can be challenging and time consuming. Our team plans to develop a Student Management System, designed to support educators and administrators in efficiently organizing and accessing student data, such as attendance, grades, and personal information. This centralized platform streamlines the process of managing student information, offering improved insights into student performance.

Here are the key issues our project addresses:

1. Student Record Management: By consolidating all necessary data in a single location, the system simplifies the management of student records for educators and administrators.
2. Attendance Monitoring: It facilitates the tracking of student attendance, enabling the generation of detailed attendance reports and highlighting patterns of absences. This data can be leveraged to enhance student engagement and intervene when necessary.
3. Performance Analysis: The system aids in monitoring student progress, providing insights into grades and identifying areas needing improvement. It supports tailored instructional strategies and personalized student support.
4. Enhanced Communication: Offering a shared platform for educators, administrators, and parents, the system fosters better communication regarding student progress and essential updates, promoting collaboration in the educational journey.
5. Streamlined Administrative Processes: By automating routine administrative duties such as report generation and grade recording, the system reduces the administrative burden, allowing educators to concentrate more on teaching and fostering student success.

In summary, the Student Management System enhances the management of student-related data, supporting educators and administrators in elevating student support and educational achievements.

What is the idea to solve the problem:

The Student Management System would include various modules such as attendance management, grade management, student profile management, and communication management. Each module would be accessible through a tab or menu in the GUI.

The attendance management module of the system would enable educators and administrators to log each student's attendance and compile detailed attendance summaries. The grade management feature would facilitate the entry of grades, allow access to grade summaries, and highlight areas where students may require additional support. Additionally, the student profile management function would provide a mechanism for storing and retrieving vital student data, including personal and academic information. The communication management aspect would establish a channel for educators, administrators, and parents to exchange information regarding students' academic performance, attendance, and other significant details.

To enhance both functionality and aesthetic appeal, JavaFX could be utilized to craft a user interface that is both adaptable and engaging, featuring easy-to-navigate controls and an intuitive layout.

What topics of CSYE6200 will be covered?

JavaFX: As a widely recognized open-source GUI toolkit for the Java programming language, JavaFX will be utilized to craft the Student Management System's interface, ensuring it is both intuitive and engaging for users.

Class Definition: Within Java, a class acts as a template for objects, detailing their attributes and actions. For the Student Management System, class definitions will articulate the properties and functionalities of various elements like student records.

Inheritance/Polymorphism: Inheritance enables a Java class to acquire properties from another class. This feature will be employed to outline the hierarchical relationships among the different classes within the Student Management System. Polymorphism, which allows objects to assume various forms, will be leveraged to enhance the system's flexibility and expandability.

Abstract Classes/Interfaces: These constructs are fundamental in Java for creating a cohesive structure. Abstract classes will serve as a foundation for other classes to build upon, while interfaces will dictate a set of methods that must be implemented by a class. They will be used to standardize the functionality across the Student Management System's components, making the code more modular and reusable.

Generics/Collections/Iterators: These concepts are pivotal for managing data in Java. Generics will specify the types of objects within collections, facilitating the management and access of database-stored data through collections and iterators.

Lists: Representing a sequence of objects, lists will be employed to organize and manipulate data such as student records, grade lists, or attendance records.

Stacks: Operating on a last-in-first-out (LIFO) basis, stacks will be used to track the Student Management System's navigation history, enabling users to revert to previously viewed pages efficiently.

Queues/Priority Queues: These structures, which organize objects based on first-in-first-out (FIFO) and priority, respectively, will be applied to manage data like sort grades by priority.

Sets/Maps: Sets, which prevent duplicate entries, will be used for storing unique data like a student's Id. Maps, organizing data in key-value pairs, will manage associations such as student IDs to their profile details or grades.

Tools used:

1. Eclipse: Eclipse will serve as our primary integrated development environment (IDE) for crafting Java code for the Student Management System. It will be instrumental in writing, compiling, and debugging our project.
2. Git Repository: Utilizing Git, a version control system, will enable streamlined management and collaboration on our source code, facilitating the handling of various code versions within our team.
3. Scene Builder: This visual layout tool for JavaFX will be employed to construct the Student Management System's interface, simplifying the design process, and enhancing efficiency.
4. Documentation Tools: For documentation and collaborative work, we'll leverage Google Docs.

What is the schedule of your project?