Student Management System –

Console-Based Java Project

# 1. Title

Student Management System (Console-Based using Java OOPs)

# 2. Problem Statement

In educational institutions, managing student data, courses, subjects, and assessments manually can be inefficient and error-prone. The absence of an automated system leads to increased workload for administrators and confusion among students about their course enrollments and exam results. Therefore, there is a need for a simplified and structured student management solution that can be operated through a console-based application.

# 3. Objectives

* To build a console-based Java application for managing student records using Object-Oriented Programming (OOP).
* To allow two types of users: Admin and Student.
* To enable the Admin to:

- Add and manage courses.

- Add and manage subjects under each course.

- View all registered students.

- View results of student examinations.

* To enable the Student to:

- Register using their personal details.

- View and enroll in available courses.

- Select subjects from the chosen course.

- Attempt an MCQ-based examination.

- View exam score and pass/fail status.

# 4. Implementation Design

## a. Technologies Used:

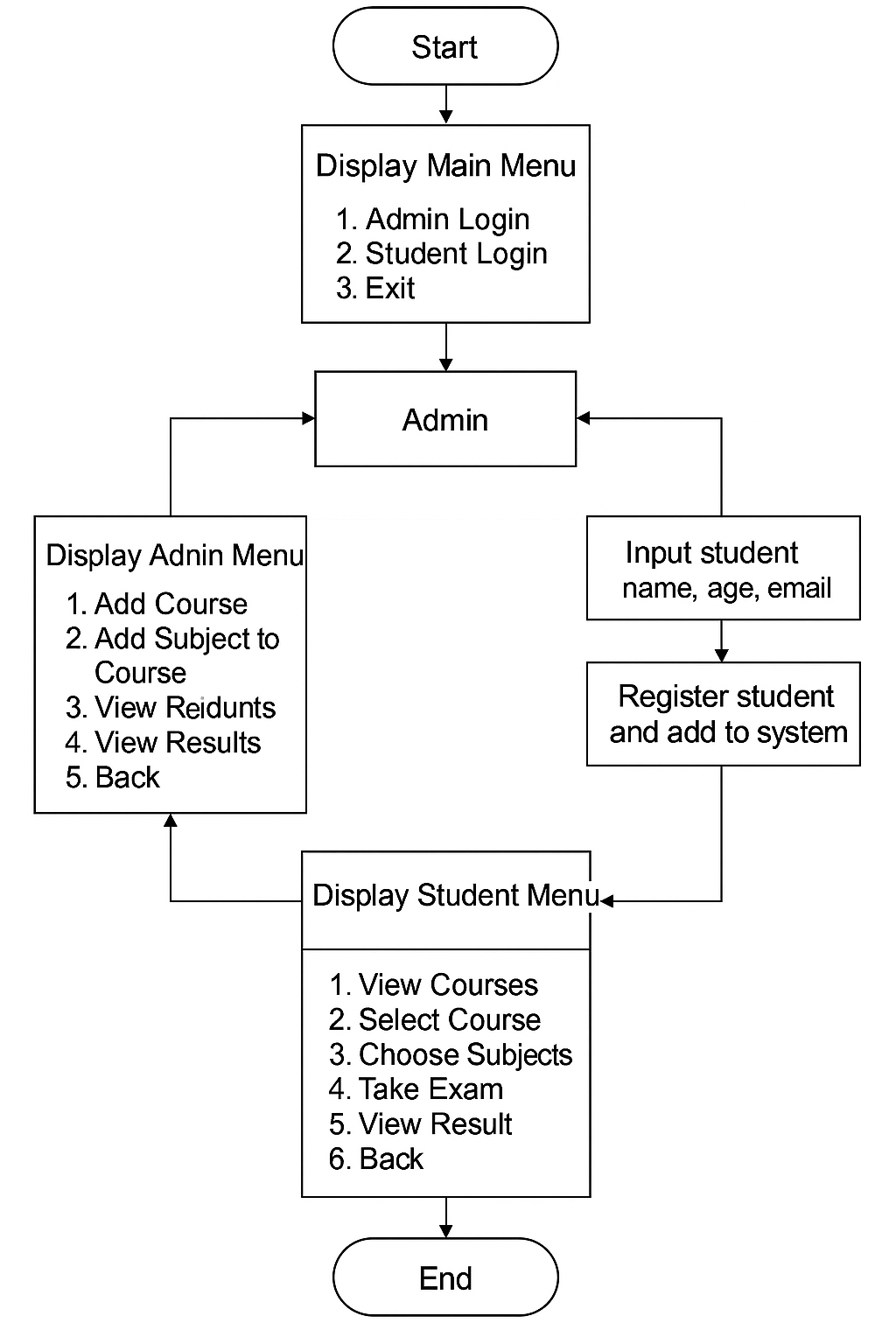
* Programming Language: Java
* Concepts: OOP (Abstraction, Inheritance, Encapsulation, Polymorphism)
* Development Environment: Any Java IDE (e.g., IntelliJ IDEA, Eclipse)

## b. System Components:

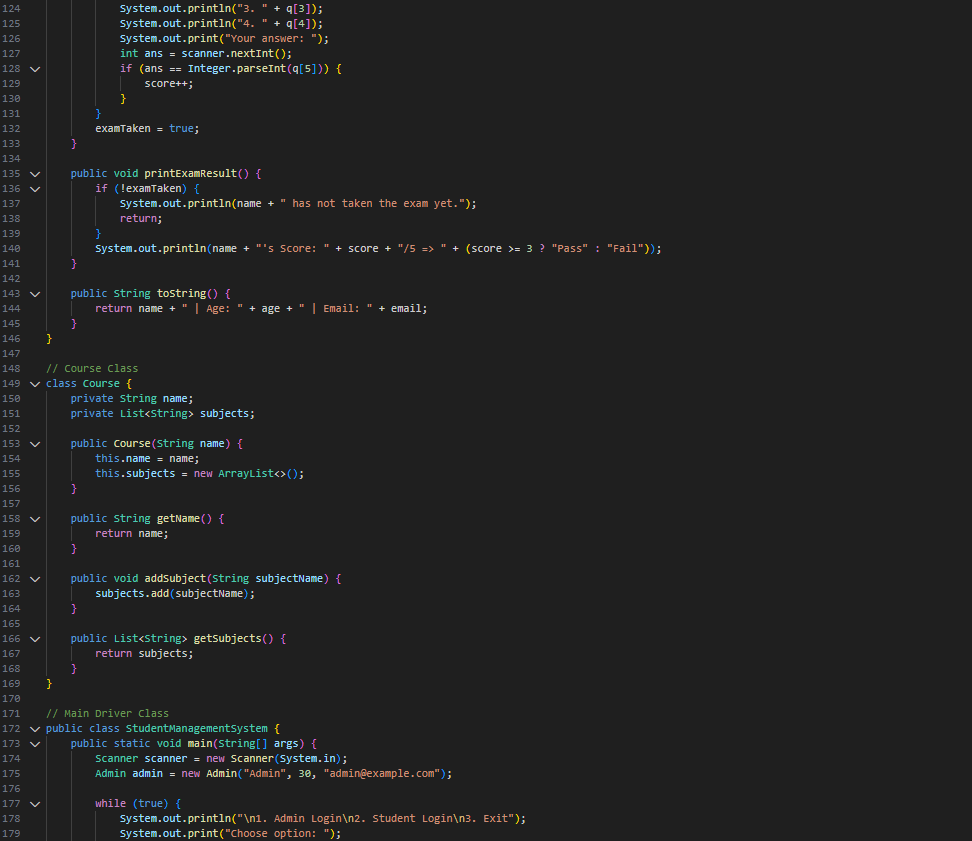
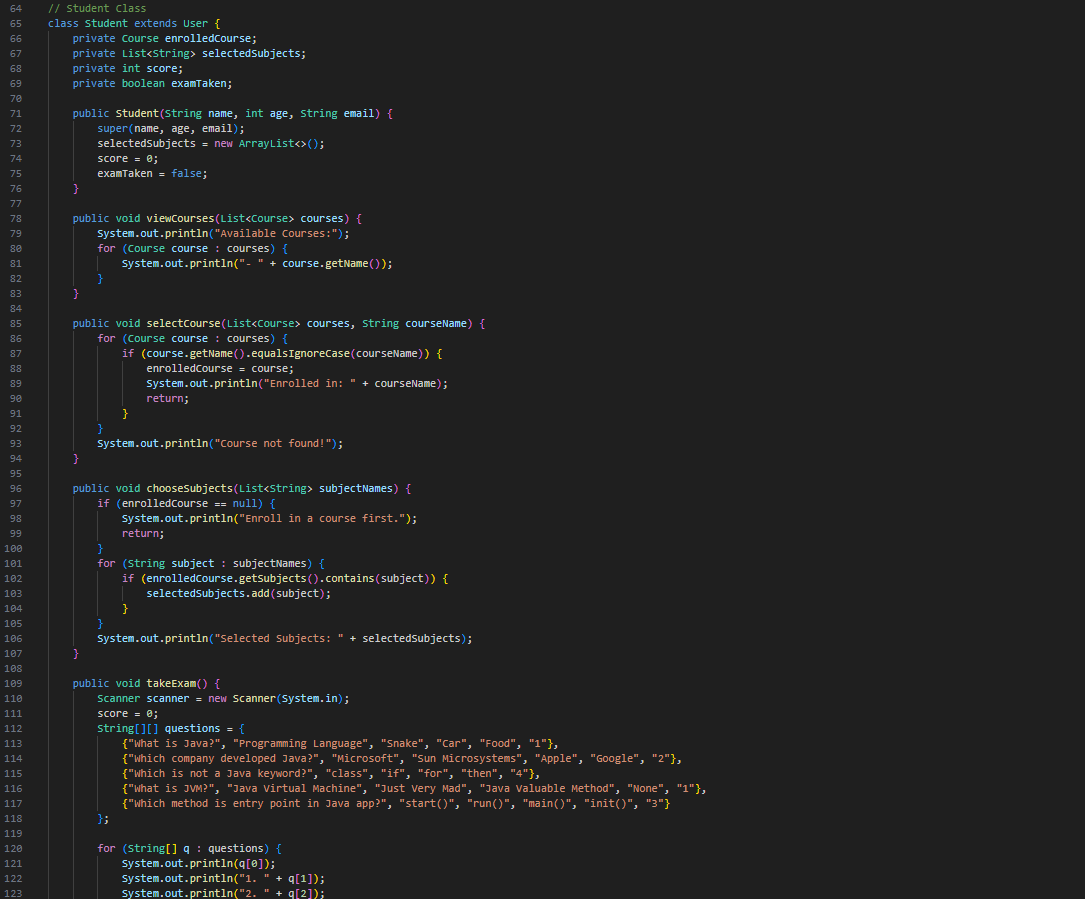
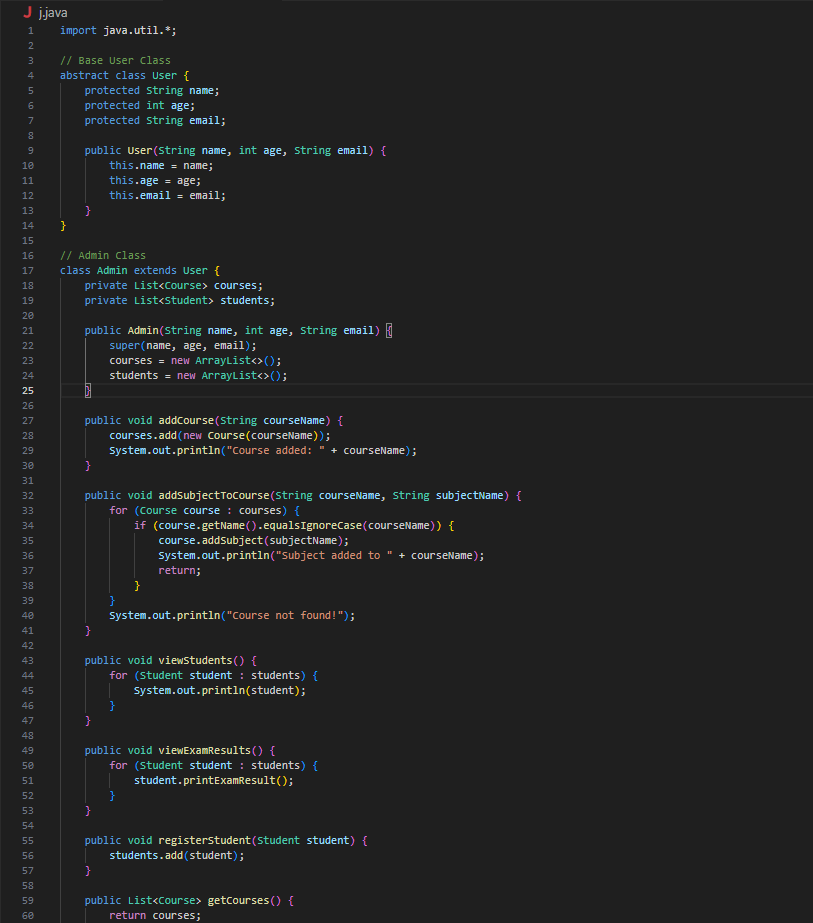
* Abstract Class `User`: Parent class for both Admin and Student. Stores common attributes like name, age, and email.
* Class `Admin`: Inherits from User. Can add/manage Course and Subject. Can register/view Student and their exam results.
* Class `Student`: Inherits from User. Can view/enroll in Course, select Subject, and take a 5-question MCQ exam. Stores score and exam status.
* Class `Course`: Contains course name and a list of subjects. Supports addition of subjects and retrieval of subject list.
* Class `StudentManagementSystem`: Contains the main method. Handles login options and menus for Admin and Student.

## c. Exam Module:

Contains 5 pre-defined MCQ questions with 4 options each. Automatically calculates score and determines pass/fail status based on answers.

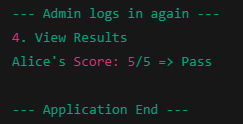


# 5. Actual Code and Sample Output



# 

# 



# 6. Conclusion

The Student Management System successfully demonstrates the use of OOP concepts in Java to solve real-world problems. It provides a basic yet effective interface for managing student data, courses, subjects, and assessments in an educational setting. Though console-based, it models the foundational structure of a scalable school/college management system and can be further enhanced with GUI, database support, or web integration in future iterations.