**STUDENT MANAGEMENT SYSTEM**

**Problem Statement**

Managing courses, training materials, and exams manually in educational institutions can be time-consuming, error-prone, and inefficient. Instructors often face difficulties keeping course content organized and evaluating students quickly, while students lack centralized access to study material and assessment opportunities.

**Objective**

To create a Java-based console application that provides a role-based interface for Admins and Students to manage and interact with courses, materials, and exams using Object-Oriented Programming principles.

**Technologies Used**

* Java (OOP, Collections, Abstraction)
* Java Console Input/Output
* Data Structures: ArrayList, HashMap
* Package Structure: Student\_Management\_System

**User Roles**

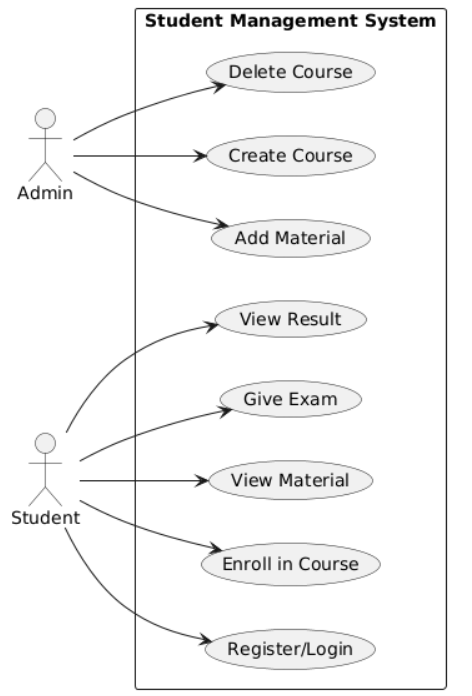
**Admin:**

* Create, update, and delete courses
* Add training materials (PDF names, web links)
* Add MCQ exam questions to courses
* View all available courses
* Protected by a hardcoded password: admin123

**Student:**

* Register with name, email, password
* View and enroll in available courses
* Access course-specific training materials
* Attempt MCQ-based exams
* View results with pass/fail status and percentage

**Use Case Diagram**



**System Architecture / Class Design**

**▪ User (abstract class)**

* Common base for Admin and Student
* Attributes: name, email, password
* Abstract method: showMenu()

**▪ Admin (extends User)**

* Creates/deletes courses
* Adds materials and exam questions

**▪ Student (extends User)**

* Enrolls in courses
* Stores and retrieves exam results
* Can access materials and give exams

**▪ Course**

* Attributes: title, description, duration, materials, exam
* Methods: addMaterial, showMaterials, get/set exam

**▪ Question**

* Contains question text, 3 options, and the correct option index (1-based)

**▪ Exam**

* A list of questions
* Conducts the exam and calculates results

**▪ Result**

* Stores score and total questions
* Calculates percentage and pass/fail status

**Program Workflow**

**1. System Initialization:**

* Admin and 3 courses are preloaded: Java Basics, Python Fundamentals, Web Development
* Each course has 2 training materials and 10 MCQ questions

**2. Role Selection:**

* User chooses to act as either Admin or Student

**Admin Flow:**

* Login using password: admin123
* Add new courses, materials, and questions
* Delete or view courses

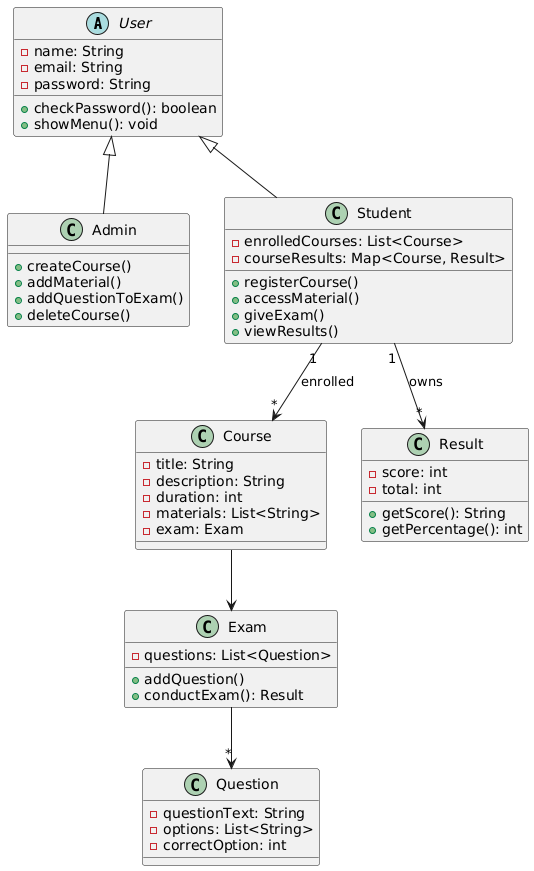
**Student Flow:**

* Register with name, email, password
* View available courses and enroll
* Access training materials of enrolled courses
* Attempt exams and view results immediately

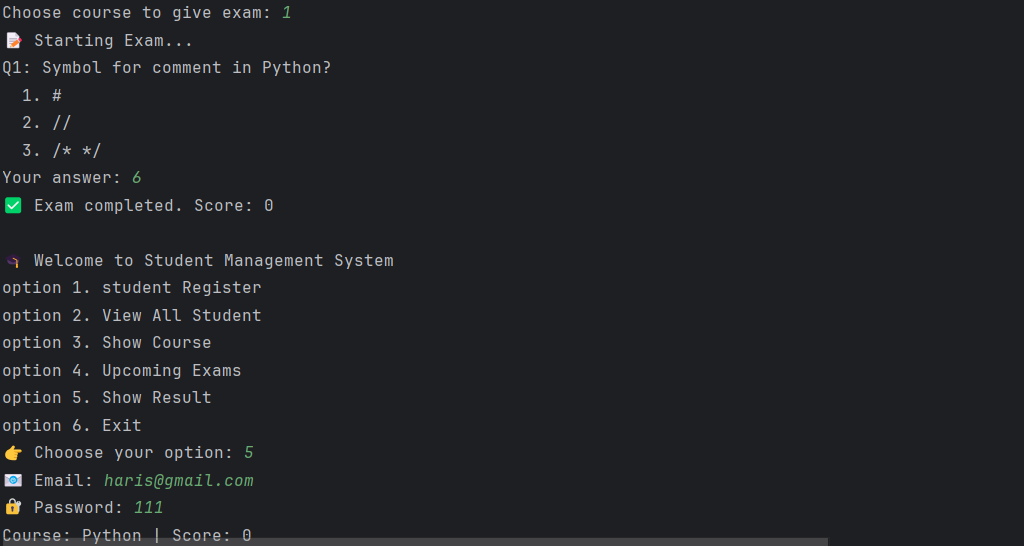
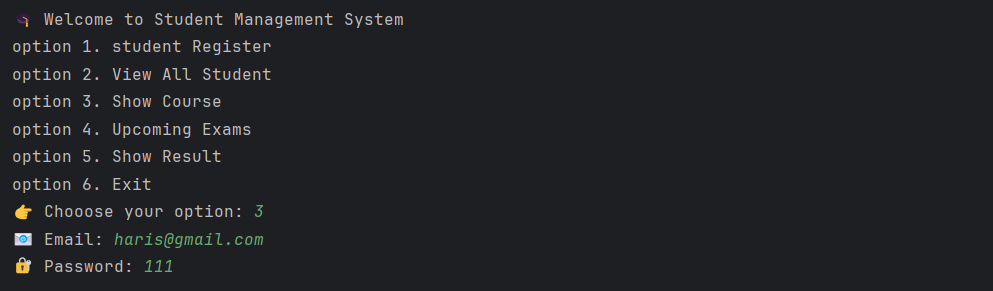
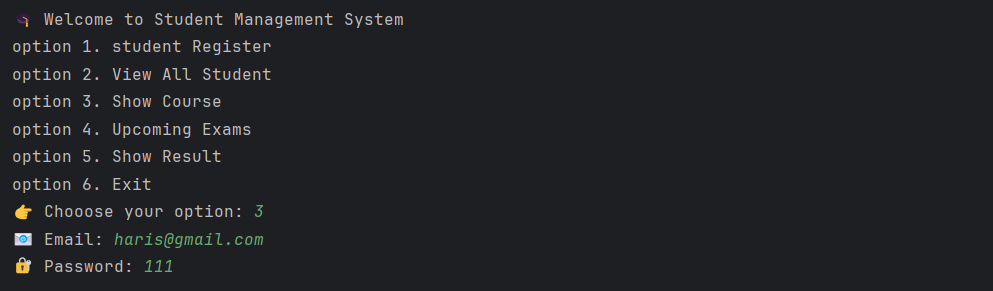
**How the System Works Together**

1. Admin creates a course.
2. Admin adds training material and questions.
3. Student enrolls in that course.
4. Student can view course's materials and take its exam.
5. Student attempts exam and gets a result.
6. Result is stored in the student record.

**Class Diagram**



**Screenshots**

****

**Conclusion**

The Student Management System offers a modular and extendable Java application that effectively demonstrates key OOP principles. It now includes data persistence for courses and exams, ensuring continuity across sessions. The system provides a functional workflow between admins and students for managing courses, materials, and assessments entirely through a console-based interface. With enhancements, it can evolve into a full-fledged learning platform.

**By:- Syed Mohammad Haris.**