

***Department of Electronics and communication engineering***

***Course Name – OOPS using Java Course code : R1UC201C Program : B.Tech ECE(AIML)***

***Second Semester(2024-25) Section : 56***

***Project title : Online Examination System Group name : Harmony Haven***

***Submitted by : Submitted to :***

***Dipshikha Mr. Ganesh***

***Harshit Aanvi***

***Yasraj***

Dipshikha Chakraborty (Admin) 24SECE1291777

Email Id :- [dipshikha.24sece1290177@galgotiasuniversity.ac.in](mailto:dipshikha.24sece1290177@galgotiasuniversity.ac.in)

Aanvi(Member) 24SECE1290094

Email Id :- [aanvi.24sece1290094@galgotiasuniversity.ac.in](mailto:aanvi.24sece1290094@galgotiasuniversity.ac.in)

Harshit(Member) 24SECE1290145

Email Id :- [harshit.24sece1290145@galgotiasuniversity.ac.in](mailto:harshit.24sece1290145@galgotiasuniversity.ac.in) Yashraj(Member)

24SECE1290098

Email Id :- [yash.24sece1290098@galgotiasuniversity.ac.in](mailto:yash.24sece1290098@galgotiasuniversity.ac.in)

Git hub repository link:-

# ACKNOWLEDGEMENT

***We would like to express our deepest gratitude to everyone who contributed to the successful completion of this project.***

***First and foremost, we extend our sincere appreciation to Mr. Ganesh for their invaluable guidance and mentorship. Their expertise, encouragement, and continuous support played a crucial role in shaping our understanding and refining our approach.***

***We would also like to acknowledge the Guvi team for their unwavering assistance, technical support, and constructive feedback throughout the project. Their insights helped enhance our workflow and optimize our outcomes.***

***A heartfelt thank you to our team members for their dedication, coordination, and collaborative spirit. The seamless teamwork, commitment to excellence, and mutual support among us were instrumental in driving this project to completion. Every individual’s efforts, contributions, and problem-solving skills ensured that we maintained efficiency and overcame challenges effectively.***

***This project stands as a testament to the power of teamwork, perseverance, and shared vision. We are truly grateful to be part of such an incredible team project.***

# PROJECT OVERVIEW

The Online Examination System is a Java-based console application designed to automate the process of conducting and evaluating exams for students. Built using OOP concepts, it provides a user-friendly interface to register students, display available exams, conduct multiple-choice quizzes, and generate results.

***Key Functionalities:-***

1. ***Student Management***
   * ***Register new students.***
   * ***Automatically generate unique student IDs.***
   * ***Store and display student details.***
2. ***Exam Management***
   * ***Create and store multiple exams(e.g.,Java,Python,C++)***
   * ***Each exam includes a set of multiple-choice questions with options and correct answers.***
3. ***Quiz Conduct***
   * ***Display questions with options.***
   * ***Allow students to take the exam by selecting the subject.***
   * ***Record student answers and calculate scores.***
4. ***Result Generation***
   * ***Evaluate the student’s answers.***
   * ***Display result summary(total questions, correct answers, wrong answers, final score)***
5. ***Menu-Driven Interface***
   * ***Provides option to:***
     + ***Register Student***
     + ***Take Exam***
     + ***View Result***
     + ***Exit***

## PROJECT STRUCTURE

*1. User (Abstract Class)*

*Attributes:*

*• userId: Unique ID.*

*• name: Name of the user.*

*Methods:*

*• getUserId()*

*• getName()*

*• Abstract method: getRole() (Implemented by child classes).*

*2. Student (Child Class of User)*

*Attributes:*

*• registeredExams: List of exams attempted.*

*Methods:*

*• getRole() returns “Student”.*

3. Exam Class

Attributes:

• examId, title, questionList.

Methods:

• addQuestion(Question q)

• getQuestions()

4. Question Class

Attributes:

• questionText, options[], correctAnswer.

Methods:

• checkAnswer(selectedOption)

5. Result Class

Attributes:

• student, exam, score.

Methods:

• calculateScore()

• generateResultReport()

6. OnlineExaminationSystem Class

Attributes:

• students, exams.

Methods:

• registerStudent(Student s)

• addExam(Exam e)

• conductExam(Student s, Exam e)

• generateResult(Student s)

7. Main Class

• Initializes sample exams and questions.

• Provides a menu to the user:

1. Register Student

2. Take Exam

3. View Result

4. Exit

**WORKFLOW**

1. Application Startup

• Initialize exams and questions.

• Display menu.

2. Menu Options

• Register Student: Adds a new student.

• Take Exam: Select an exam and answer questions.

• View Result: Shows score report.

• Exit: End application.

**OOP Concepts Used:**

• Abstraction: User class abstracted for general user behavior.

• Inheritance: Student extends User.

• Encapsulation: Private fields accessed via getters/setters.

• Polymorphism: getRole() method implemented differently.

**import java.util.\*;**

**class Question {**

**String question;**

**String[] options;**

**char answer;**

**Question(String question, String[] options, char answer) {**

**this.question = question;**

**this.options = options;**

**this.answer = answer;**

**}**

**public boolean askQuestion(Scanner sc) {**

**System.out.println("\n" + question);**

**for (int i = 0; i < options.length; i++) {**

**System.out.println((char)('A' + i) + ". " + options[i]);**

**}**

**System.out.print("Your Answer (A/B/C/D): ");**

**char userAnswer = Character.toUpperCase(sc.next().charAt(0));**

**return userAnswer == answer;**

**}**

**}**

**class User {**

**String username;**

**String password;**

**User(String username, String password) {**

**this.username = username;**

**this.password = password;**

**}**

**public boolean login(String uname, String pwd) {**

**return username.equals(uname) && password.equals(pwd);**

**}**

**}**

**public class OnlineExamSystem {**

**public static void main(String[] args) {**

**Scanner sc = new Scanner(System.in);**

**// Predefined user (you can add registration feature)**

**User user = new User("student", "pass123");**

**System.out.println("=== Welcome to Online Examination System ===");**

**// Login**

**System.out.print("Enter Username: ");**

**String uname = sc.nextLine();**

**System.out.print("Enter Password: ");**

**String pwd = sc.nextLine();**

**if (!user.login(uname, pwd)) {**

**System.out.println("Invalid Credentials! Exiting...");**

**return;**

**}**

**System.out.println("Login Successful!\n");**

**// Sample Questions**

**Question[] questions = {**

**new Question("What is the capital of France?",**

**new String[]{"Berlin", "London", "Paris", "Rome"}, 'C'),**

**new Question("Which data type is used to create a variable that should store text?",**

**new String[]{"String", "int", "char", "float"}, 'A'),**

**new Question("Which company developed Java?",**

**new String[]{"Microsoft", "Apple", "Sun Microsystems", "Google"}, 'C'),**

**new Question("Which of these is NOT a Java keyword?",**

**new String[]{"class", "interface", "extends", "define"}, 'D'),**

**new Question("What does JVM stand for?",**

**new String[]{"Java Virtual Machine", "Java Verified Machine", "Just Virtual Machine", "None of the above"}, 'A')**

**};**

**int score = 0;**

**// Exam loop**

**for (Question q : questions) {**

**if (q.askQuestion(sc)) {**

**System.out.println("Correct!");**

**score++;**

**} else {**

**System.out.println("Wrong!");**

**}**

**}**

**System.out.println("\nExam Over!");**

**System.out.println("Your Score: " + score + " out of " + questions.length);**

**// Result**

**if (score >= 3) {**

**System.out.println("Result: PASSED");**

**} else {**

**System.out.println("Result: FAILED");**

**}**

**sc.close();**

**}**

**}**

**Future Enhancements:**

**•** Persistent data storage using files or databases.

• GUI (Swing/JavaFX) Interface.

• Timer-based examination.

• Detailed analytics and progress reports.

**Conclusion:**

The Online Examination System successfully demonstrates core OOP concepts and offers a practical, extensible platform for conducting exams virtually.

***.***