PROJECT REPORT

Project Title: Student Performance Analyzer
Course: Build Your Own Project (VITyarthi) fundamental in AIML
Student: Siya Dubey
Registration no.-25BCY10204

2.Introduction

This project titled Student Performance Analyzer is a simple Python-based system designed to manage student records and predict their performance using rule-based logic. It is aligned with the project structure guidelines provided in the uploaded PDF.

3. Problem Statement

Students often face difficulty understanding their academic standing based solely on marks. Teachers also require a simple digital method to track performance. Manual evaluation is time-consuming and unorganized. This project provides a small-scale automated solution for storing student details, recording marks, and predicting performance categories.

4. Functional Requirements

• Add Student • Store Marks • Predict Performance (Excellent, Good, Average, Poor) • View All Students • Search a Student

5. Non-Functional Requirements

• Usability: Easy text-based interface • Performance: Fast execution with simple data • Maintainability: Clean and modular code • Reliability: Basic input validation included

6. System Architecture

User Input → Processing Logic → Performance Prediction → Output Display

7. Design Diagram

(Textual Representation) Start → Show Menu → User Selects Option → Perform Operation → Display Output → Return to Menu → Exit

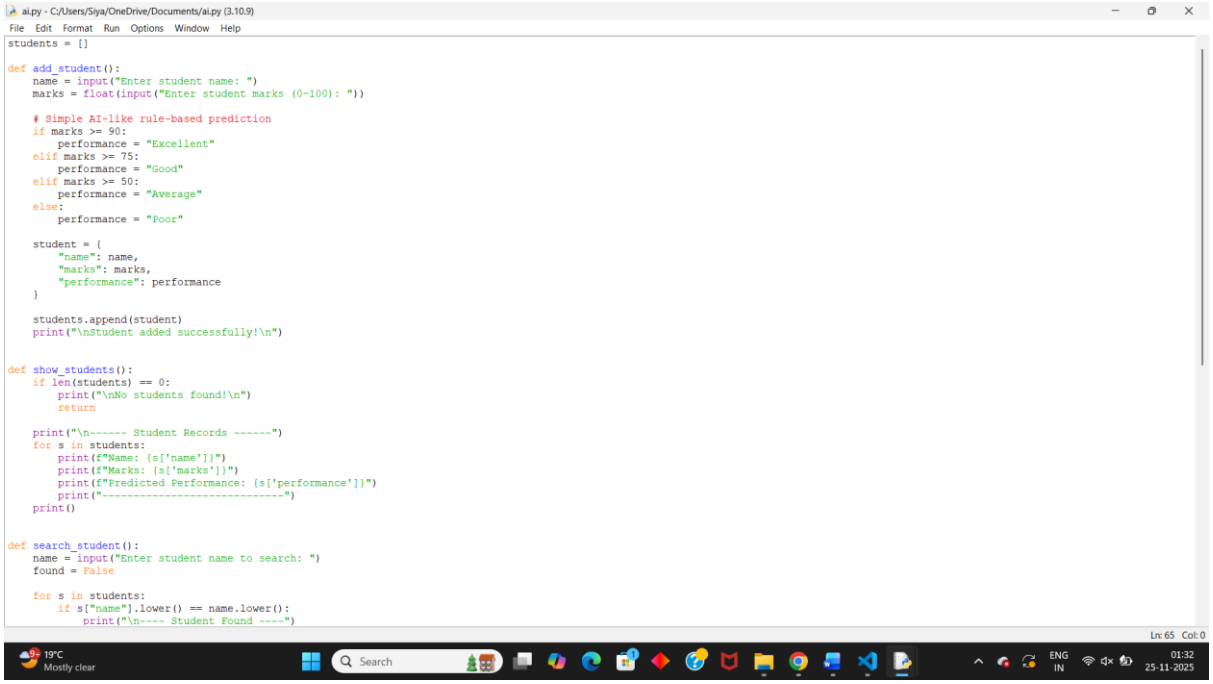
8. Design decision and rationale

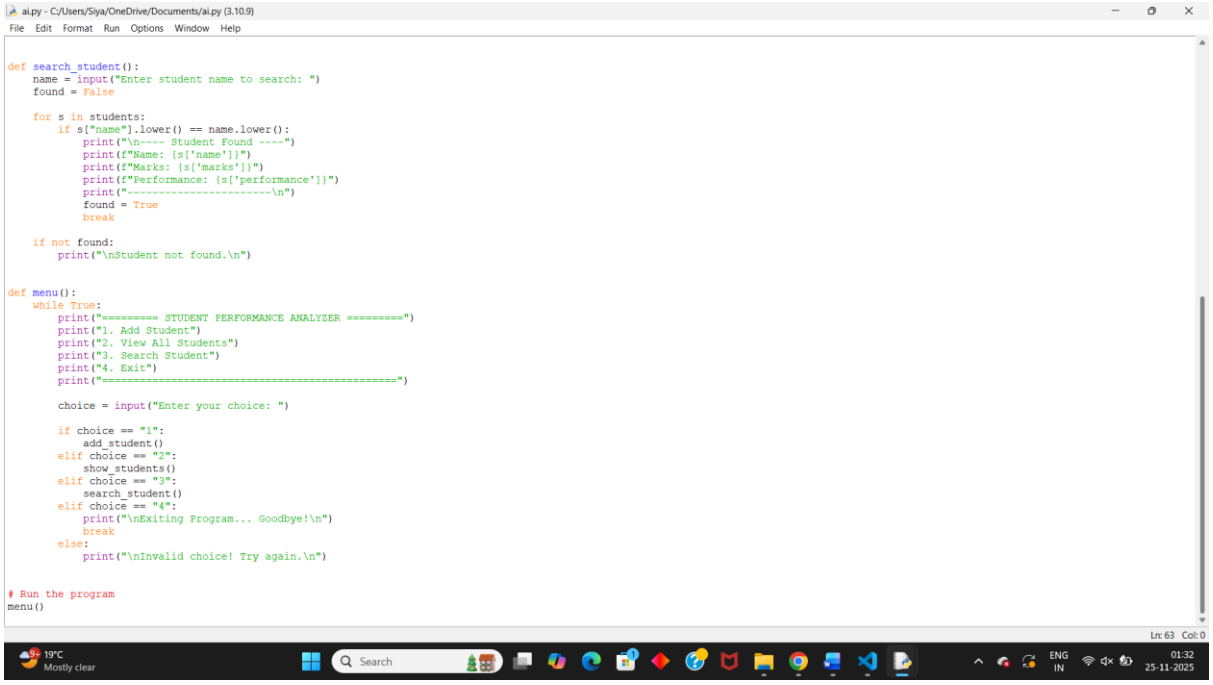
(Text Format) Use Case: Actors: User Use Cases: Add Student, View Students, Search Student, Predict Performance Sequence Flow: User → Menu → Operation → Display Result

9. Implementation Details

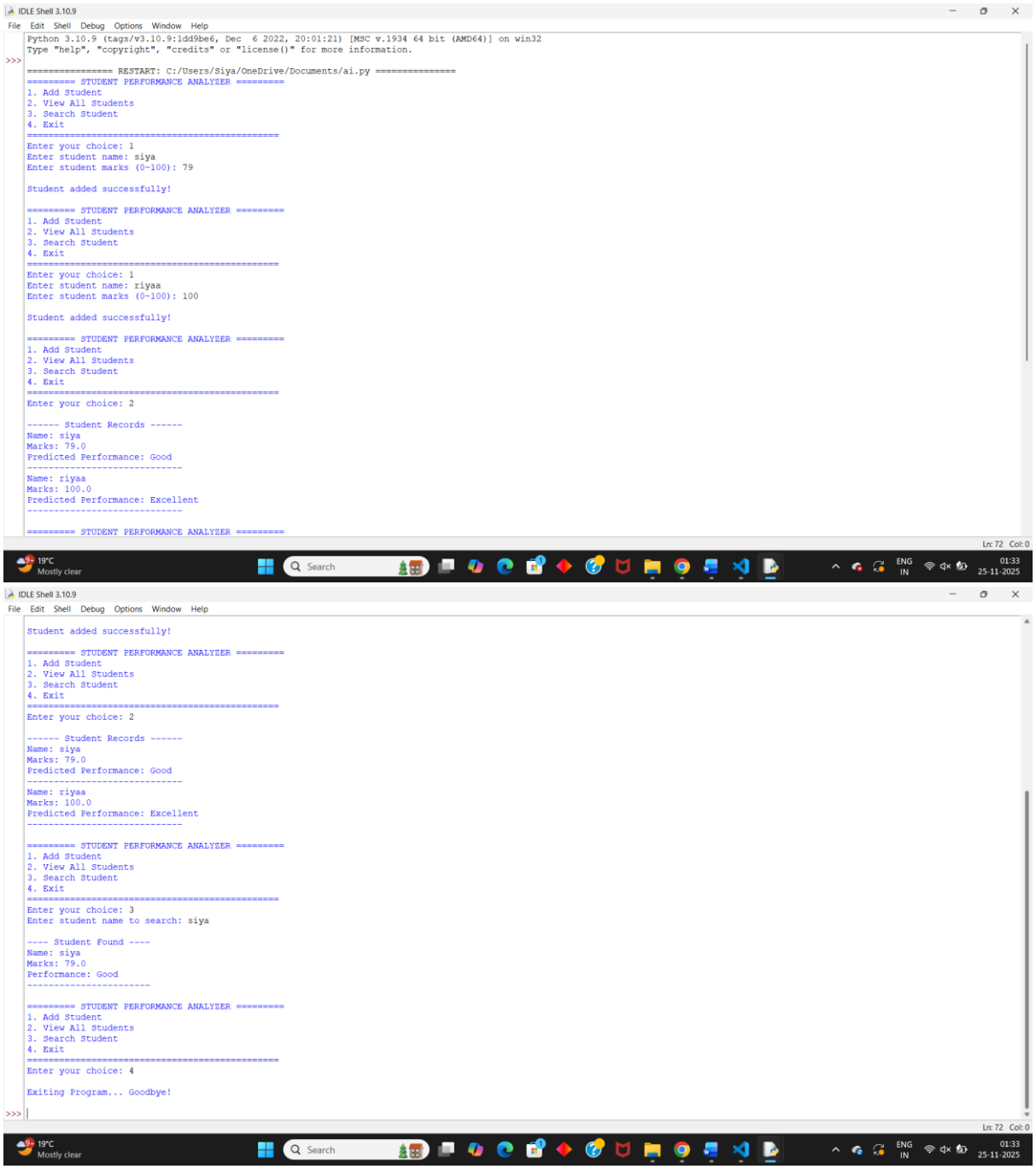
The project is implemented in a single Python file using lists and basic conditional logic. No external libraries are required. Code Summary • Menu-driven program • Student records stored as dictionaries • Rule-based prediction logic

10. Results & Screenshots





Output:-



11. Testing Approach

Manual testing performed using sample inputs

: • Marks 95 → Excellent • Marks 68 → Average • Invalid menu choices handled

12. Challenges Faced

• Designing simple yet meaningful AI-like behavior

• Ensuring clarity in user input handling

13. Learnings & key takeways

• Improved understanding of Python conditional logic

• Better understanding of structured program design

14. Future Enhancements

• Graphical User Interface (GUI)

• Database integration

• Real Machine Learning model for prediction

15. References

Class notes

Geeks for geeks