AI-Assisted Smart Attendance & Performance Tracker

Project Overview

The **Smart Attendance & Performance Tracker** is a console-based Python application designed to manage student records, track attendance, record marks, and generate basic analytics. The system follows a modular architecture and uses JSON for persistent data storage.

Objectives

- Maintain student records (Roll No, Name, Semester)
- Track attendance and calculate attendance percentage
- Record academic performance (marks)
- Provide automated remarks and warnings
- Generate simple analytics for insights
- Ensure clean, modular, and maintainable code

Features

1. Student Management

- Add new students with validation
- Prevent duplicate roll numbers
- View individual student reports
- Display all students

2. Attendance Management

- Mark total and attended lectures
- Automatic attendance percentage calculation
- Attendance shortage warning if below 75%

3. Performance Management

- Enter marks (0–100)
- AI-style performance remarks:
- **Good** ($\geq$ 75)
- **Average** (50–74)
- **Needs Improvement** (< 50)

4. Analytics

- Total number of students
- Average attendance
- Average marks

Project Structure

```
Smart Attendance/
│
├── models/
│   └── student.py
│
├── services/
│   └── attendance_tracker.py
│
├── ui/
│   └── display.py
│
├── data/
│   └── student_data.json
│
└── main.py
```

Module Description

1. `student.py`

- Defines the `Student` class
- Stores student data
- Contains logic for:
- Attendance percentage
- Performance remark
- Attendance warning

2. `attendance_tracker.py`

- Core business logic
- Manages students dictionary
- Handles:
- Validation
- Attendance updates
- Marks entry

- Analytics
- JSON save/load

3. `display.py`

- Handles all console output
- Displays menus and formatted reports

4. `main.py`

- Entry point of the application
- Menu-driven user interaction
- Calls appropriate services based on user choice

Menu Options

Choice	Description
1	Add Student
2	Mark Attendance
3	Enter Marks
4	View Student Report
5	Display All Students
6	View Analytics
7	Exit

Validation Rules

- Roll Number: Numeric (1–9999)
- Name: Cannot be empty
- Semester: Between 1 and 8
- Attendance: Attended $\leq$ Total lectures
- Marks: Between 0 and 100

Data Storage

- Uses **JSON file** (`student_data.json`)
- Automatically saved after every update
- Automatically loaded on application start

Sample Output (Student Report)

```
Roll No      : 58
Name         : Yash
Semester     : 2
Attendance   : 80.0%
Status       : ✓ Satisfactory
Marks        : 72
Performance  : Average
```

How to Run

1. Ensure Python 3.7+ is installed
2. Open terminal in project folder
3. Run:

```
python main.py
```

Advantages

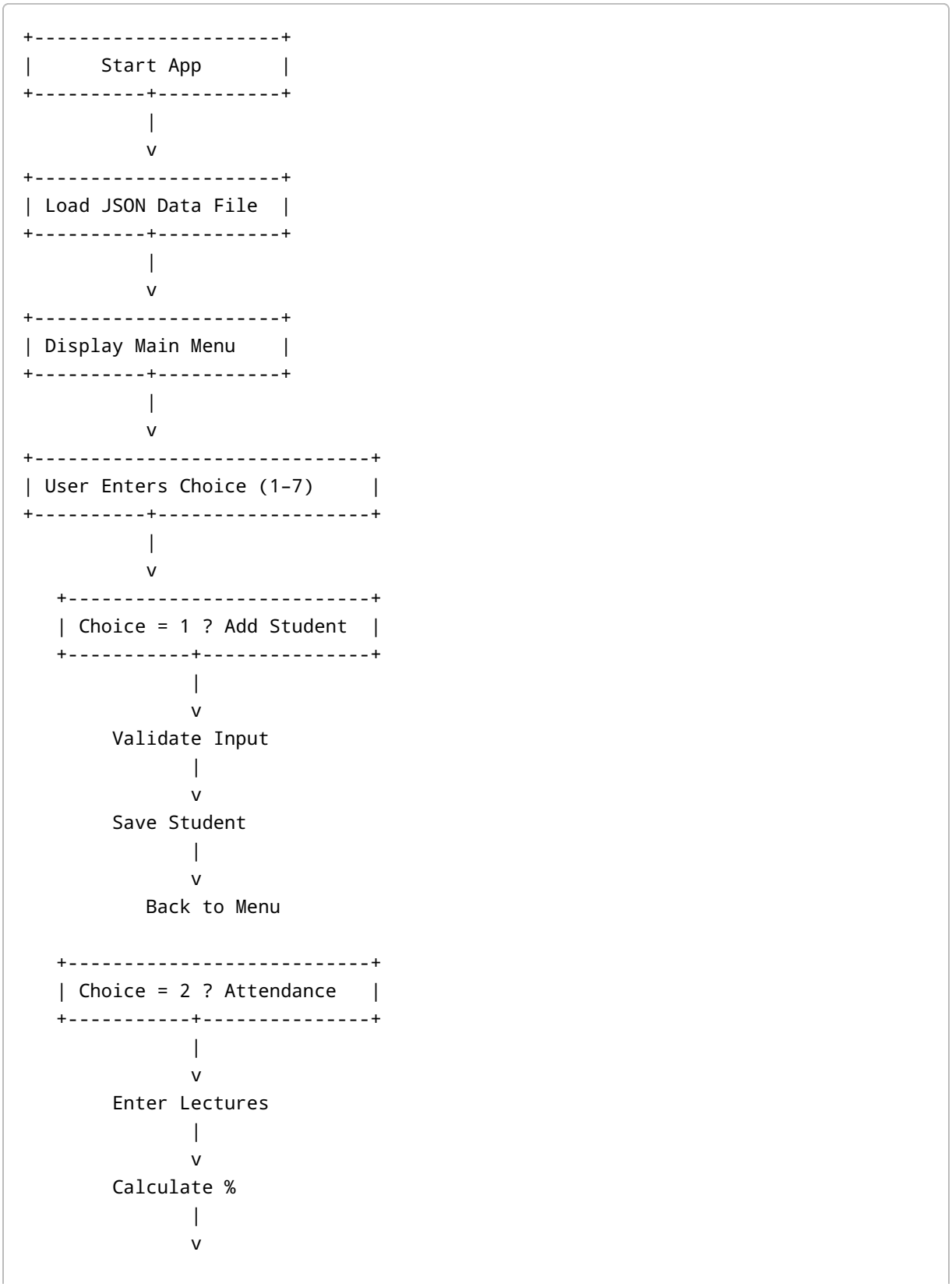
- Modular and readable code
- Easy to extend (GUI / Web / Database)
- Beginner-friendly and interview-ready
- Demonstrates AI-assisted logic

Conclusion

This project demonstrates a clean and structured approach to building a real-world student management system using Python. The modular design ensures scalability and maintainability while providing essential academic tracking features.

Status: Project Completed ✅

Flow Chart (Program Execution)



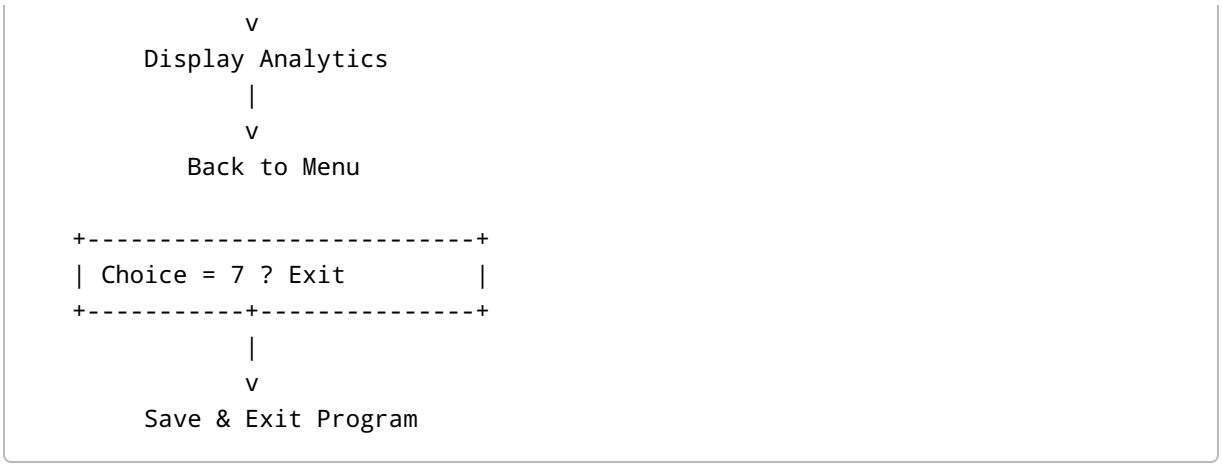
```
        Save Data
            |
            v
        Back to Menu

+---------------------------+
| Choice = 3 ? Enter Marks  |
+-----------+---------------+
            |
            v
     Validate Marks
            |
            v
     Save Marks
            |
            v
        Back to Menu

+---------------------------+
| Choice = 4 ? View Report  |
+-----------+---------------+
            |
            v
     Fetch Student
            |
            v
     Display Report
            |
            v
        Back to Menu

+---------------------------+
| Choice = 5 ? View All     |
+-----------+---------------+
            |
            v
     Display Students
            |
            v
        Back to Menu

+---------------------------+
| Choice = 6 ? Analytics    |
+-----------+---------------+
            |
            v
     Calculate Summary
            |
```


v
Display Analytics
|
v
Back to Menu
+---------------------------+
| Choice = 7 ? Exit |
+-----------+---------------+
|
v
Save & Exit Program