Project Report

Student Report Card Generator

Author: Abu Bakar

Python Intern at CodeX Computing Co.

1. Introduction

This report outlines the development and features of the Student Report Card Generator, a mini project developed during the Python internship with CodeX Computing Co. The project aims to provide an efficient way to manage and generate student report cards in multiple formats using Python.

2. Project Overview

The Student Report Card Generator is a command-line based Python application designed to manage student academic records and generate detailed report cards. Users can add students, enter their subject-wise marks, calculate grades, and generate report cards in Text, Word, and PDF formats.

3. Key Features

- Create and manage student records with name and roll number.
- Add multiple subjects and marks for each student.
- Automatic grade calculation based on average marks.
- Generate report cards in Text (txt), Word (.docx), and PDF (.pdf) formats.
- Store student data persistently using JSON.
- Input validation and duplicate roll number handling.
- User-friendly menu-driven command-line interface.

4. Technical Details

The application is developed in Python 3.x. It utilizes the following libraries:
- `python-docx`: For generating Word documents.
- `reportlab`: For generating PDF documents.
- `dataclasses` and `typing`: For structured data management and type annotations.
The student data is saved in a JSON file to ensure data persistence across sessions.

5. Functional Overview

The core functionalities of the system are implemented in two main classes:
1. `Student`: Handles student data, subject management, and grade calculation.
2. `ReportCardGenerator`: Manages student records, handles file generation, and provides a menu interface.
The application validates all inputs and gracefully handles errors such as invalid marks or duplicate roll numbers.

6. Challenges and Learning

Developing this project enhanced my understanding of Python object-oriented programming, file handling, data persistence, and document generation. One of the main challenges was formatting outputs consistently across different formats (text, Word, and PDF). This was addressed by structuring the report generation logic carefully.

7. Conclusion

The Student Report Card Generator serves as a practical demonstration of applying Python to solve real-world problems. This project helped me solidify my Python skills and introduced me to advanced concepts like external library usage, report generation, and persistent storage. It stands as a foundation for more advanced educational tools in the future.