

Student Management System (Python Mini Project)

Project Overview

You are required to build a **console-based Student Management System (SMS)** using **core Python functions**. This system allows users to **add, view, search, update, and delete** student records easily from the terminal. All data must be stored in a **text file** for persistence, and every operation should include **input validation** and **user feedback**.

Key Python Concepts Used

- Variables & Data Types
- Lists and Dictionaries
- If–Else Conditions and Loops
- **Functions** (for modular programming)
- File Input/Output (read, write, update data)
- User Input Validation (numeric and text checks)

Project Requirements

1. Create a **menu-driven console program** that runs until the user exits.
2. Implement separate **functions** for each operation:
 - Add a new student
 - View all students
 - Search student by ID
 - Update student record
 - Delete student record
3. Store all student data in a text file named students.txt.
4. Each record in the file must follow this format:
5. ID, Name, Age, Grade, Section
6. Validate user input for correct types and empty values.
7. Display messages and formatted output for user clarity.

Core Features

- **AddNewStudent:**
Input new student details after validating all fields and save to file.
- **ViewAllStudents:**
Read from file and display all records in a neat table format.
- **SearchStudentbyID:**
Locate a specific student record using their unique ID.
- **UpdateStudentRecord:**
Modify one or more details of an existing student and rewrite the file.
- **DeleteStudent:**
Remove a record by ID and update the file accordingly.
- **Exit:**
Close the program gracefully with a thank-you message.

Input Validation Rules

- **Student ID:** Must be numeric and unique.
- **Name & Section:** Cannot be empty or contain only spaces.
- **Age:** Must be an integer greater than 0.
- **Grade:** Must be a single uppercase letter (A–F).
- Show a **clear error message** and re-prompt user for invalid inputs.

Expected Console Output Example:

```
===== Student Management System =====

1. Add New Student
2. View All Students
3. Search Student by ID
4. Update Student
5. Delete Student
6. Exit
Enter your choice: 1
Enter Student ID: 101
Enter Name: Ali Ahmed
Enter Age: 17
Enter Grade: A
Enter Section: Blue
Student added successfully!
Enter your choice: 2

ID Name Age Grade Section
-----
101 Ali Ahmed 17 A Blue
```