AI & Data Science – Assignment 4

🏫 Build a Simple School Management System (Group Project)

🎯 Objective

As a group, you will build a **School Management System** using only the Python topics we have learned:

□ concepts

- functions
- lists
- loops
- if/else statements
- input()
- formatted strings (f-strings)

□ System Description

The program should allow the school to:

- Add students
- View students
- Search students
- Remove students
- Edit students
- Count students
- Show top student (highest marks)

All students will be stored inside **one list**.

🎁 Data Format

Use this structure:

```
students = []
```

Each student should be stored like:

```
[name, age, marks]
```

Example:

```
["John", 14, 78]
```

👥 Task Distribution

Each student writes **ONLY their assigned function**.

👤 https://github.com/royaltyfavs379 — Add Student

```
def add_student(students):
```

Tasks:

- input name
- input age
- input marks
- add to list using append()
- print success message

👤 https://github.com/ale12-bit — View Students

```
def view_students(students):
```

Tasks:

- loop through students
- display all students
- number them

Example:

```
1. John - Age: 14 - Marks: 78
```

👤 https://github.com/YvonneMwaura — Search Student

```
def search_student(students):
```

Tasks:

- ask for name
- find student using loop
- display details
- if not found → print "Student not found"

👤 https://github.com/gtrevor025-a11y — Remove Student

```
def remove_student(students):
```

Tasks:

- ask for name
- remove student from list
- print confirmation

□ https://github.com/annwambeti769-png — Count Students

```
def count_students(students):
```

Tasks:

- use len(students)
- print total number

Example:

```
Total students: 5
```

□ https://github.com/Moseti7 — Edit Student

```
def edit_student(students):
```

Tasks:

- search by name
- allow user to change age or marks
- update the list

□ https://github.com/Sir-Andie28 — Top Student (Highest Marks)

```
def top_student(students):
```

Tasks:

- loop through students
- find highest marks
- print best student

Example:

```
Top student is Mary with 92 marks
```

Starter Code

```
# ===== IMPORT STUDENT FUNCTIONS =====
from add_student import add_student
from view_students import view_students
from search_student import search_student
from remove_student import remove_student
from count_students import count_students
from edit_student import edit_student
from top_student import top_student
# ===== SHARED DATA =====
students = []
# ===== MAIN PROGRAM =====
def main():
    while True:
        print("\n===== SCHOOL MANAGEMENT SYSTEM =====")
        print("1. Add Student")
        print("2. View Students")
        print("3. Search Student")
        print("4. Remove Student")
        print("5. Count Students")
        print("6. Edit Student")
        print("7. Top Student")
        print("8. Exit")

        choice = input("Choose: ")

        if choice == "1":
            add_student(students)
        elif choice == "2":
            view_students(students)
        elif choice == "3":
            search_student(students)
        elif choice == "4":
            remove_student(students)
        elif choice == "5":
            count_students(students)
        elif choice == "6":
            edit_student(students)
        elif choice == "7":
            top_student(students)
        elif choice == "8":
            break
        else:
            print("Invalid choice")

main()
```

✅ Testing Requirements

Before submission:
- ☐ function runs without errors
- ☐ correct output
- ☐ works with multiple students
- ☐ uses loops + lists correctly

💻 Submission Instructions (VERY IMPORTANT)

Create your own branch in the project repository, upload your .py file to that branch, commit your changes, then open a Pull Request to the **main branch** for review and merging.

Each student's **filename MUST match the import**:

Function	File name they must upload
add_student	add_student.py
view_students	view_students.py
search_student	search_student.py
remove_student	remove_student.py
count_students	count_students.py
edit_student	edit_student.py
top_student	top_student.py