Use a variable to activate a Course\_Class which reads the csv file.

**Course\_Class**

**Subject\_List** Gathers names of all subjects on a course

**Student\_List** Gathers names of all students on a course

Course\_Class also contains analysis functions relevant to assignment.

Iterate through the **Student\_List** and apply the **Student\_Class**

For each student in the **Student\_List** create a student.

**Student\_Class**

Every student has a name

A field that contains a dictionary

The dictionary key is built calling the course class subject list

The dictionary value applies the corresponding grade from matching subject index value

Student class also contains functions

*The final output of the program should show the GPA of each student,*

*their highest scoring module,*

*their lowest scoring module,*

*standard deviation, median value,*

*how far from the next highest GPA they were (if not at 4.2),*

*and the letter grade of each module result.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Letter Grade** | **4.2 Scale** | **Percent Grade** |  |
| A+ | 4.2 | 99.96 | 100 |
| A | 4 | 94.08 | 99.95 |
| A- | 3.8 | 88.2 | 94.07 |
| B+ | 3.6 | 82.32 | 88.19 |
| B | 3.4 | 76.44 | 82.31 |
| B- | 3.2 | 70.56 | 76.43 |
| C+ | 3 | 64.68 | 70.55 |
| C | 2.8 | 58.8 | 64.67 |
| C- | 2.6 | 52.92 | 58.79 |
| D+ | 2.4 | 47.04 | 52.91 |
| D | 2.2 | 41.16 | 47.03 |
| D- | 2 | 35.28 | 41.15 |
| E+ | 1.8 | 29.4 | 35.27 |
| E | 1.6 | 23.52 | 29.39 |
| E- | 1.4 | 17.64 | 23.51 |
| F+ | 1.2 | 11.76 | 17.63 |
| F | 1 | 5.88 | 11.76 |
| Fail | <1 | 0 |  |

There should also be a live mode where the user can enter a set of marks and modules from the command line and have the program calculate the GPA.