Project Report

Submitted By - Shreem Bhargava

Registration Number - 25MIM10011

Course - Introduction to problem solving and programming

Course Code – CSE1021

Slot- A21+A22+A23

Faculty – Dr. Lakshmi D

Class Id - BL2024260100806

Flipped Course - Python Essentials (VITyarthi)

Project Title: Relative Grade Calculator Using Python (Google Colab based)

1. Introduction

Grade calculation for students become difficult when the number of students increases.

This project helps to automatically calculate grade of students using Python. Instead of manual grading, the system uses mean and standard deviation to assign relative grades.

The project works completely on Google Colab and produces an Excel file with grade columns as output.

2. Problem Statement

Manual grading requires time and effort from teachers.

Errors can happen while calculating average, mean, and SD.

To solve this problem, a Python-based automated tool is developed to calculate grades directly from marks in Excel

3. Objective

- To automate grade calculation using relative grading
- To apply grade rules for 5 subjects
- To generate a final Excel file with grade columns
- To simplify teachers workload

4. Functional Requirements

a. Upload Excel file
b. Read marks for S1–S5
c. Calculate mean and SD

d. Apply grade logic

e. Export Excel with Grade_S1 … Grade_S5

5. Non-Functional Requirements

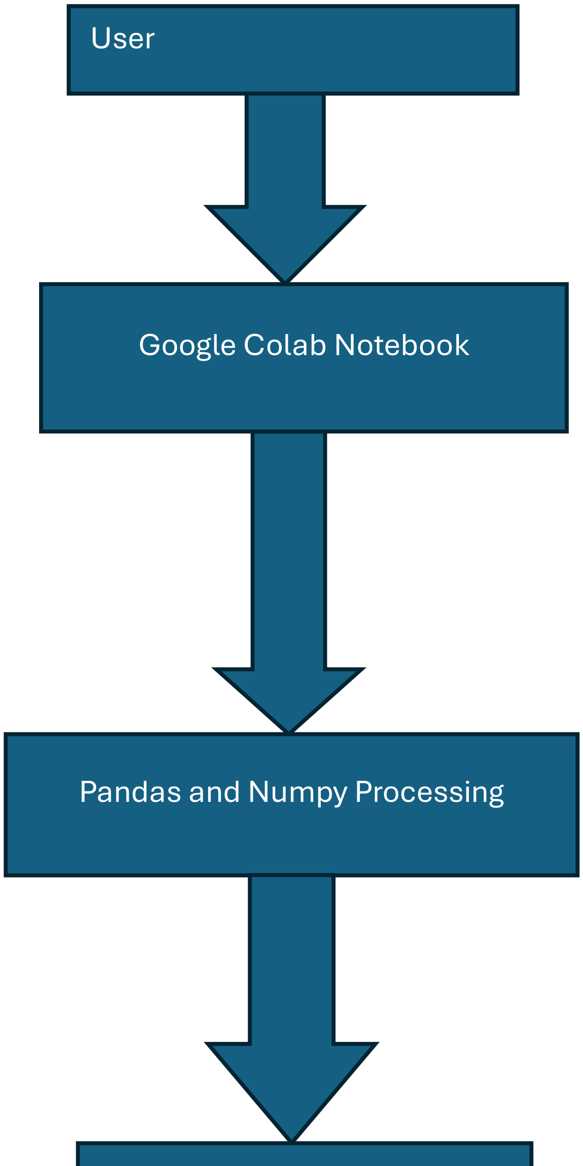
a. Accuracy: Correct mean, SD and grade

b. Usability: Easy upload and download

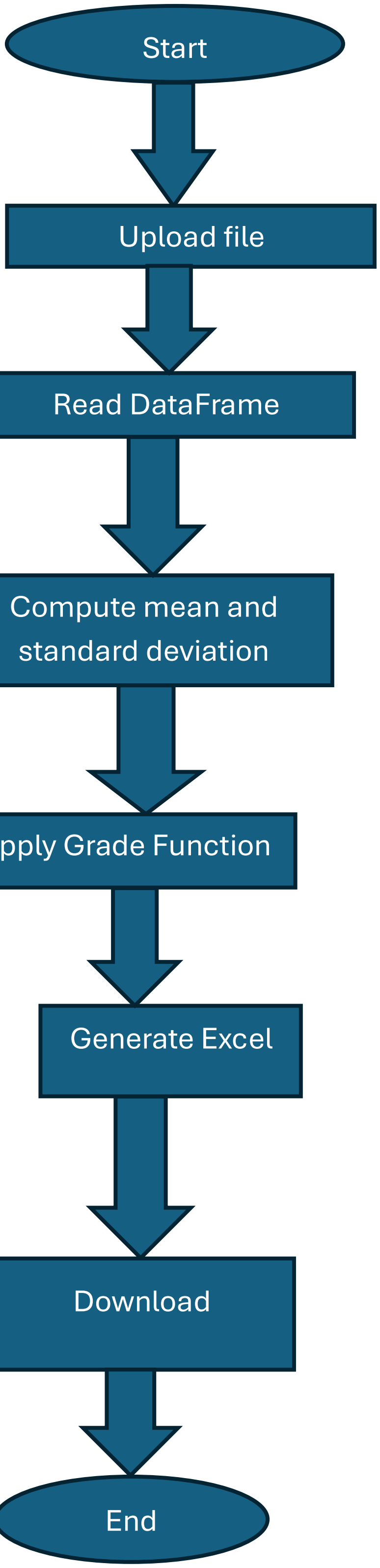
c. Efficiency: Processes all students quickly

d. Reliability: Same output every time

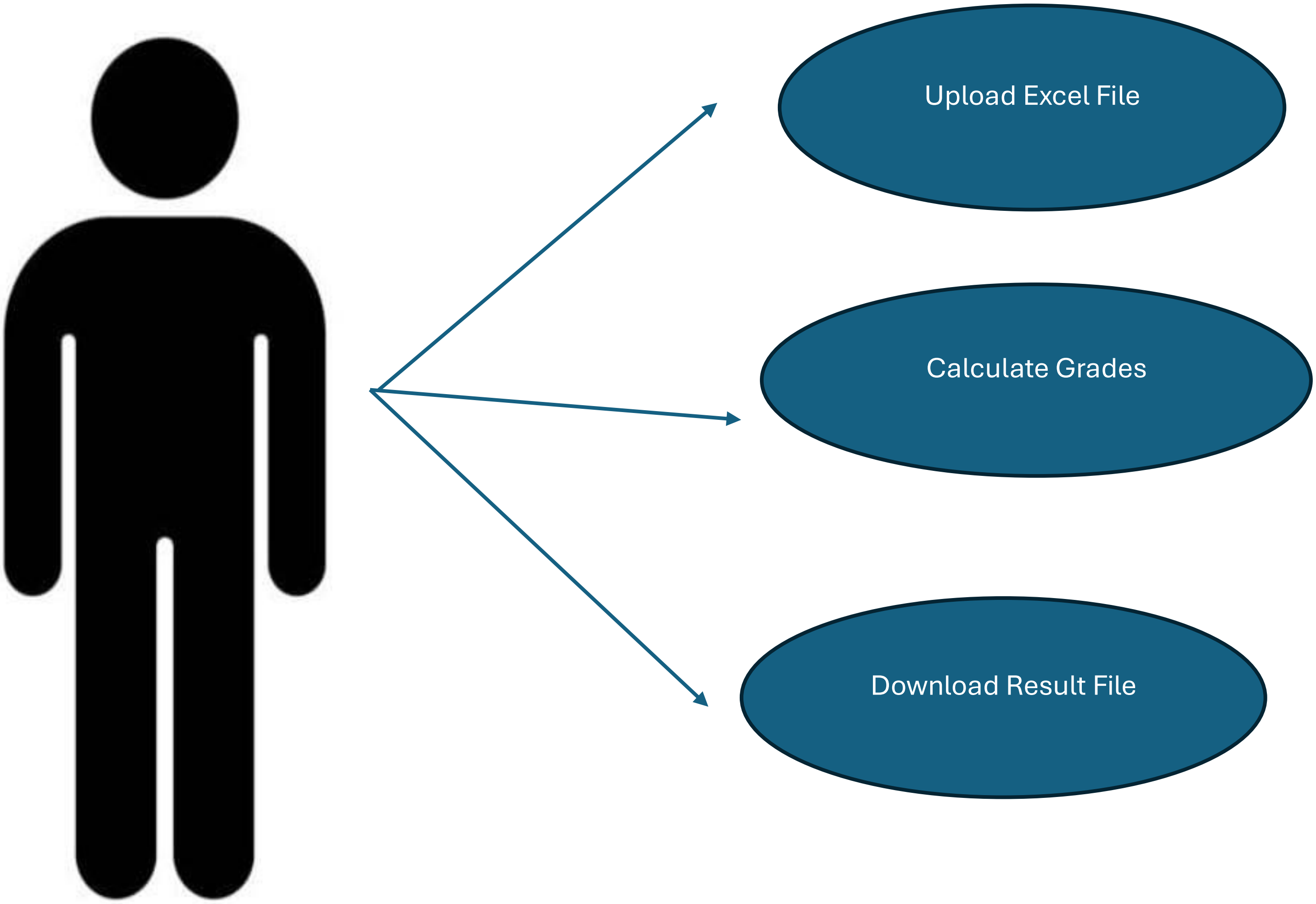
6. System Architecture



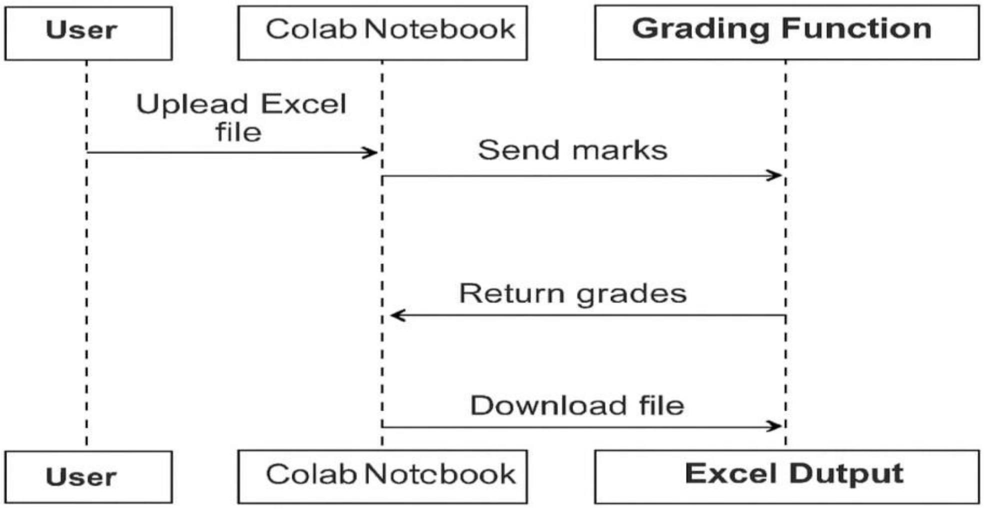
7. Workflow Diagram



8. Use Case Diagram



9. Sequence Diagram



10. Implementation Details

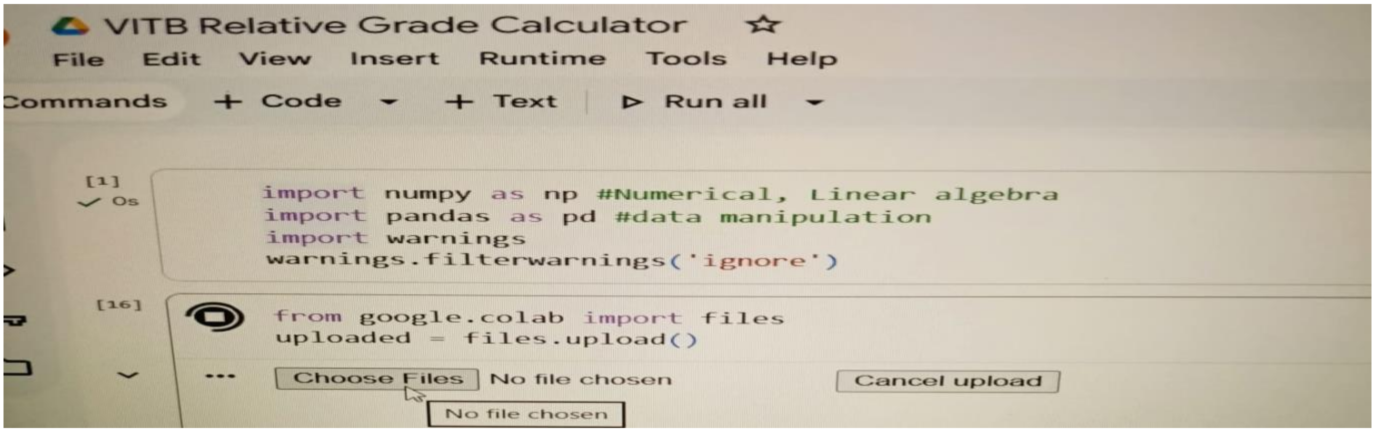
The project uses Google Colab for execution. The pandas is used to read the Excel File.

The mean and standard deviations for each subject are calculated.

The *get_grade() function* assigns grades based on conditions.

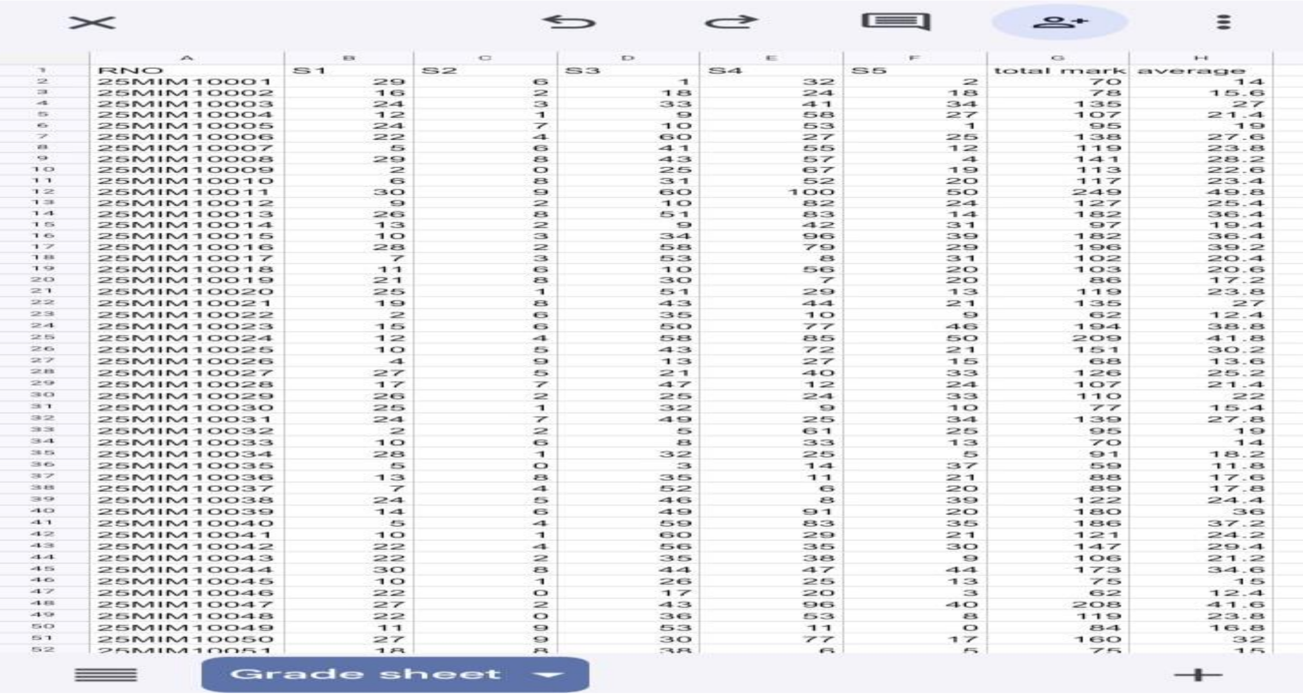
The final DataFrame is exported to a new Excel file.

11. Screenshots

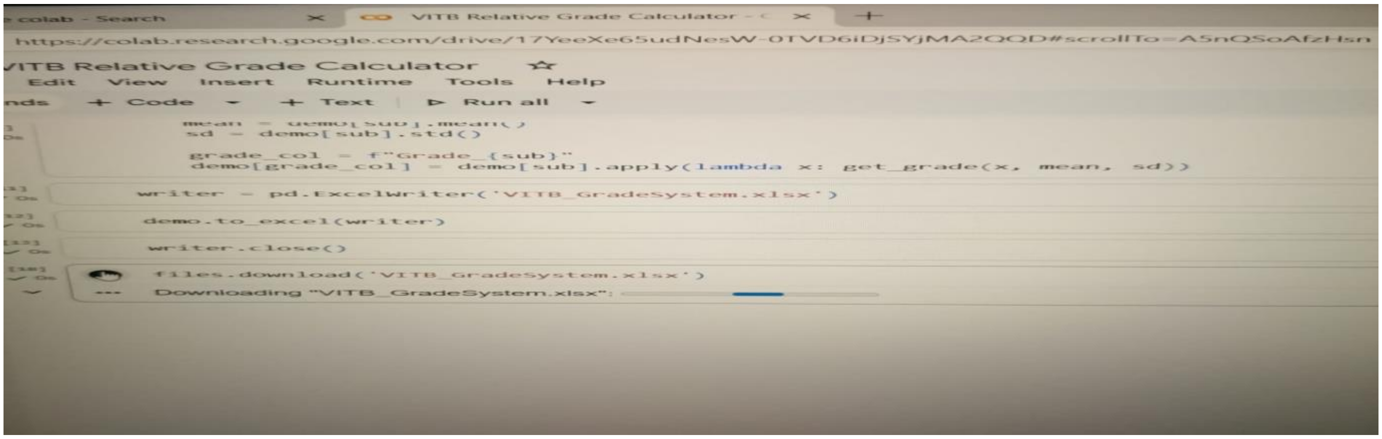


```
import numpy as np #Numerical, Linear algebra
import pandas as pd #data manipulation
import warnings
warnings.filterwarnings('ignore')
```

```
from google.colab import files
uploaded = files.upload()
```



RNO	S1	S2	S3	S4	S5	total mark	average
25MIM10001	29	6	1	32	2	70	14
25MIM10002	16	2	18	24	18	78	15.6
25MIM10003	24	3	33	41	34	135	27
25MIM10004	12	1	9	58	27	107	21.4
25MIM10005	24	7	10	53	1	95	19
25MIM10006	22	4	60	27	25	138	27.6
25MIM10007	5	6	41	55	12	119	23.8
25MIM10008	29	8	43	57	4	141	28.2
25MIM10009	2	0	25	67	19	113	22.6
25MIM10010	6	8	31	52	20	117	23.4
25MIM10011	30	9	60	100	50	249	49.8
25MIM10012	9	2	10	82	24	127	25.4
25MIM10013	26	8	51	83	14	182	36.4
25MIM10014	13	2	9	42	31	97	19.4
25MIM10015	10	3	34	96	39	182	36.4
25MIM10016	28	2	58	79	29	196	39.2
25MIM10017	7	3	53	8	31	102	20.4
25MIM10018	11	6	10	56	20	103	20.6
25MIM10019	21	8	30	7	20	86	17.2
25MIM10020	25	1	51	29	13	119	23.8
25MIM10021	19	8	43	44	21	135	27
25MIM10022	2	6	35	10	9	62	12.4
25MIM10023	15	6	50	77	46	194	38.8
25MIM10024	12	4	58	85	50	209	41.8
25MIM10025	10	5	43	72	21	151	30.2
25MIM10026	4	9	13	27	15	68	13.6
25MIM10027	27	5	21	40	33	126	25.2
25MIM10028	17	7	47	12	24	107	21.4
25MIM10029	26	2	25	24	33	110	22
25MIM10030	25	1	32	9	10	77	15.4
25MIM10031	24	7	49	25	34	139	27.8
25MIM10032	2	2	5	61	25	95	19
25MIM10033	10	6	8	33	13	70	14
25MIM10034	28	1	32	25	5	91	18.2
25MIM10035	5	0	3	14	37	59	11.8
25MIM10036	13	8	35	11	21	88	17.6
25MIM10037	7	4	52	6	20	89	17.8
25MIM10038	24	5	46	8	39	122	24.4
25MIM10039	14	6	49	91	20	180	36
25MIM10040	5	4	59	83	35	186	37.2
25MIM10041	10	1	60	29	21	121	24.2
25MIM10042	22	4	56	35	30	147	29.4
25MIM10043	22	2	35	38	9	106	21.2
25MIM10044	30	8	44	47	44	173	34.6
25MIM10045	10	1	26	25	13	75	15
25MIM10046	22	0	17	20	3	62	12.4
25MIM10047	27	2	43	96	40	208	41.6
25MIM10048	22	0	36	53	8	119	23.8
25MIM10049	11	9	53	11	0	84	16.8
25MIM10050	27	9	30	77	17	160	32
25MIM10051	18	8	38	6	5	75	15



colab - Search
VITB Relative Grade Calculator
https://colab.research.google.com/drive/17YeeXe65udNesW-0TVD6iDjSYjMA2QQD#scrollTo=A5nQSoAfzHsn
VITB Relative Grade Calculator
Edit View Insert Runtime Tools Help
+ Code + Text Run all

```
        mean = demo[sub].mean()
        sd = demo[sub].std()

        grade_col = f"Grade_{sub}"
        demo[grade_col] = demo[sub].apply(lambda x: get_grade(x, mean, sd))
```

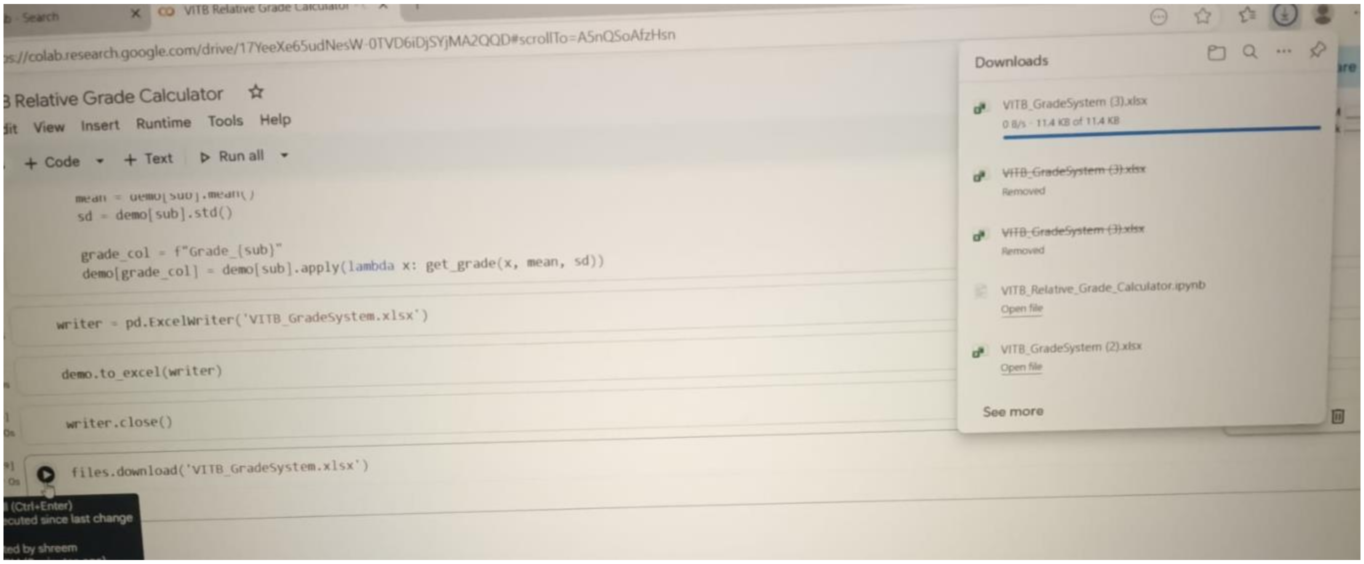
```
writer = pd.ExcelWriter('VITB_GradeSystem.xlsx')
```

```
demo.to_excel(writer)
```

```
writer.close()
```

```
files.download('VITB_GradeSystem.xlsx')
```

```
Downloading "VITB_GradeSystem.xlsx":
```



Search
VITB Relative Grade Calculator
https://colab.research.google.com/drive/17YeeXe65udNesW-0TVD6iDjSYjMA2QQD#scrollTo=A5nQSoAfzHsn
Relative Grade Calculator
dit View Insert Runtime Tools Help
+ Code + Text Run all

```
        mean = demo[sub].mean()
        sd = demo[sub].std()

        grade_col = f"Grade_{sub}"
        demo[grade_col] = demo[sub].apply(lambda x: get_grade(x, mean, sd))
```

```
writer = pd.ExcelWriter('VITB_GradeSystem.xlsx')
```

```
demo.to_excel(writer)
```

```
writer.close()
```

```
files.download('VITB_GradeSystem.xlsx')
```

(Ctrl+Enter)
ecuted since last change
ted by shreem

Downloads

VITB_GradeSystem (3).xlsx
0 B/s · 11.4 KB of 11.4 KB

~~VITB_GradeSystem (3).xlsx~~
Removed

~~VITB_GradeSystem (3).xlsx~~
Removed

VITB_Relative_Grade_Calculator.ipynb
Open file

VITB_GradeSystem (2).xlsx
Open file

See more

12. Results

The system produces a new Excel file named VITB_GradeSystem.xlsx.

It contains subjects S1–S5, total, average, and new grade columns.

13. Testing

Tested with:

- High marks
- Low marks (<40)
- Average marks
- Random values

The function correctly produced S, A, B, C, D, E, and F.

14. Challenges

Understanding mean–SD logic

Adding fail condition

Applying lambda function

15. Learnings

Pandas DataFrame handling

Using Numpy for mean/SD

Using Google Colab for automation

Writing grade logic function

16. Future Enhancements

Add visualization

Add Tkinter UI (optional)

Add student name mapping

Store data in database

17. Conclusion

The project successfully automates relative grading using Python.

It reduces manual work and produces error-free results.

Google Colab made the implementation simple and accessible.