EDS Minor Project

University Dataset

Guided By : Aarti Deshpande Mam

Team Member

Div: B Batch: B3

251	Amol Popalghat
254	Ayush Rewatkar
261	Aditya Bhange

Index

Introduction

Details of Dataset

Data Manipulation

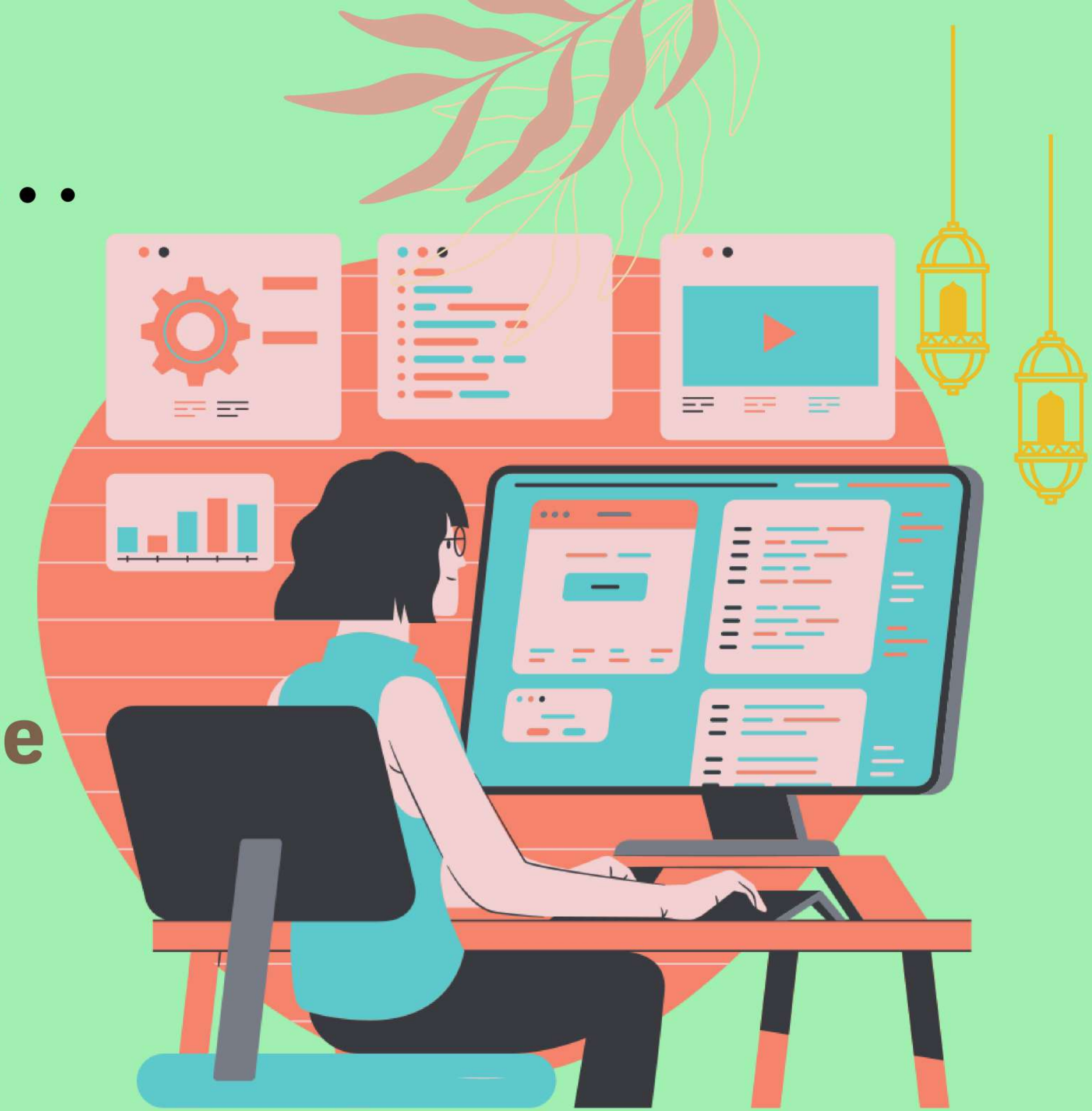
Data Visualization

Predictive Technique

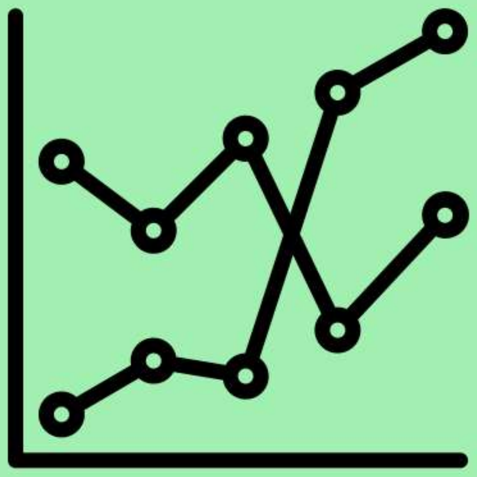
Application

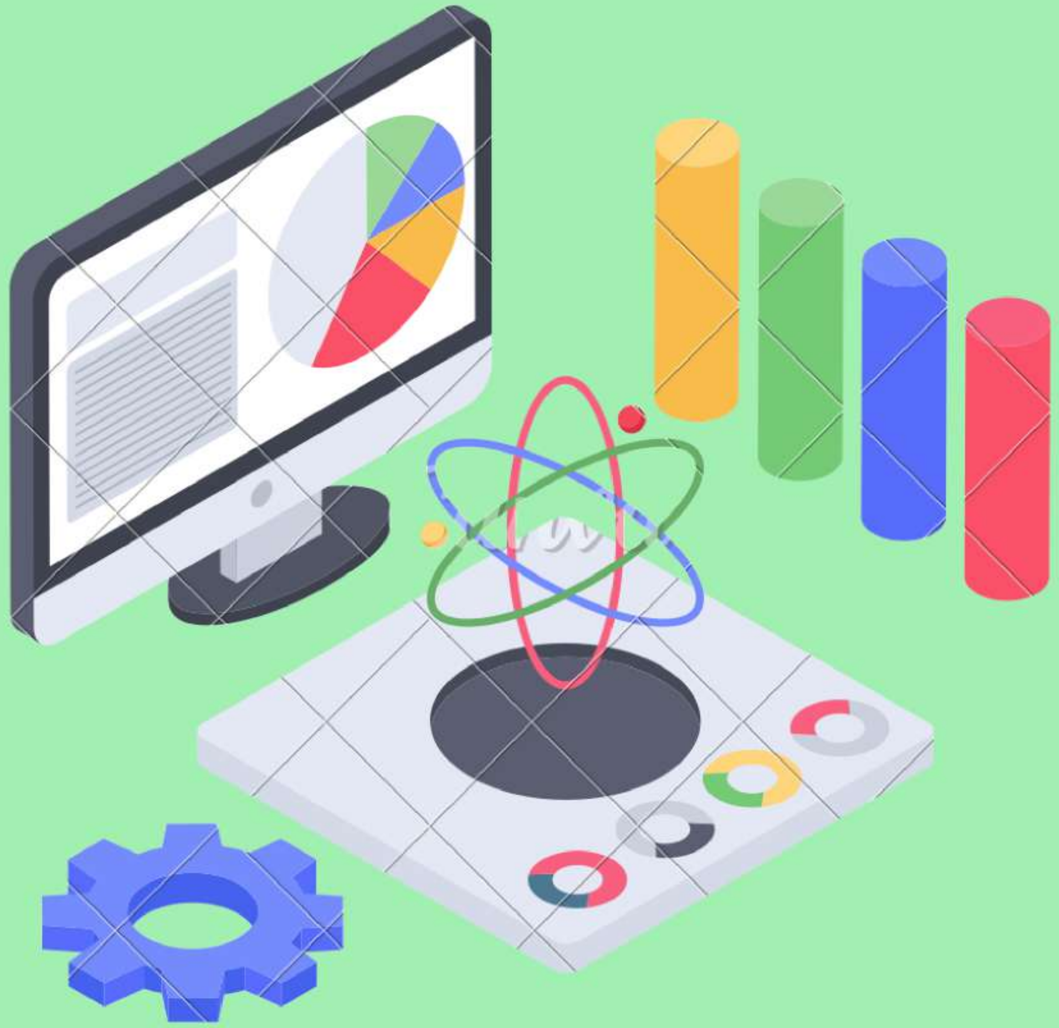
Referances

Conclusion

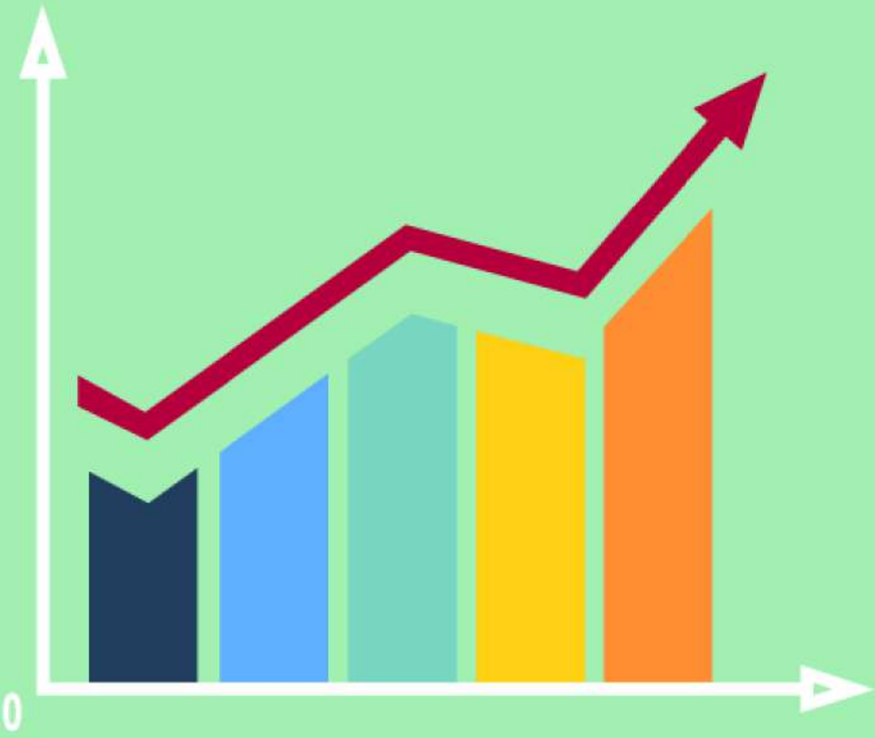


Introduction





- Data science is the study of data.
- Data scientists find patterns in data.
- Programming, statistics, machine learning are essential.
- Data science combines math and programming.
- Insights guide decision making



Details of Dataset

NAME : UNIVERSITY

Number of features: 9

Number of records: 5211

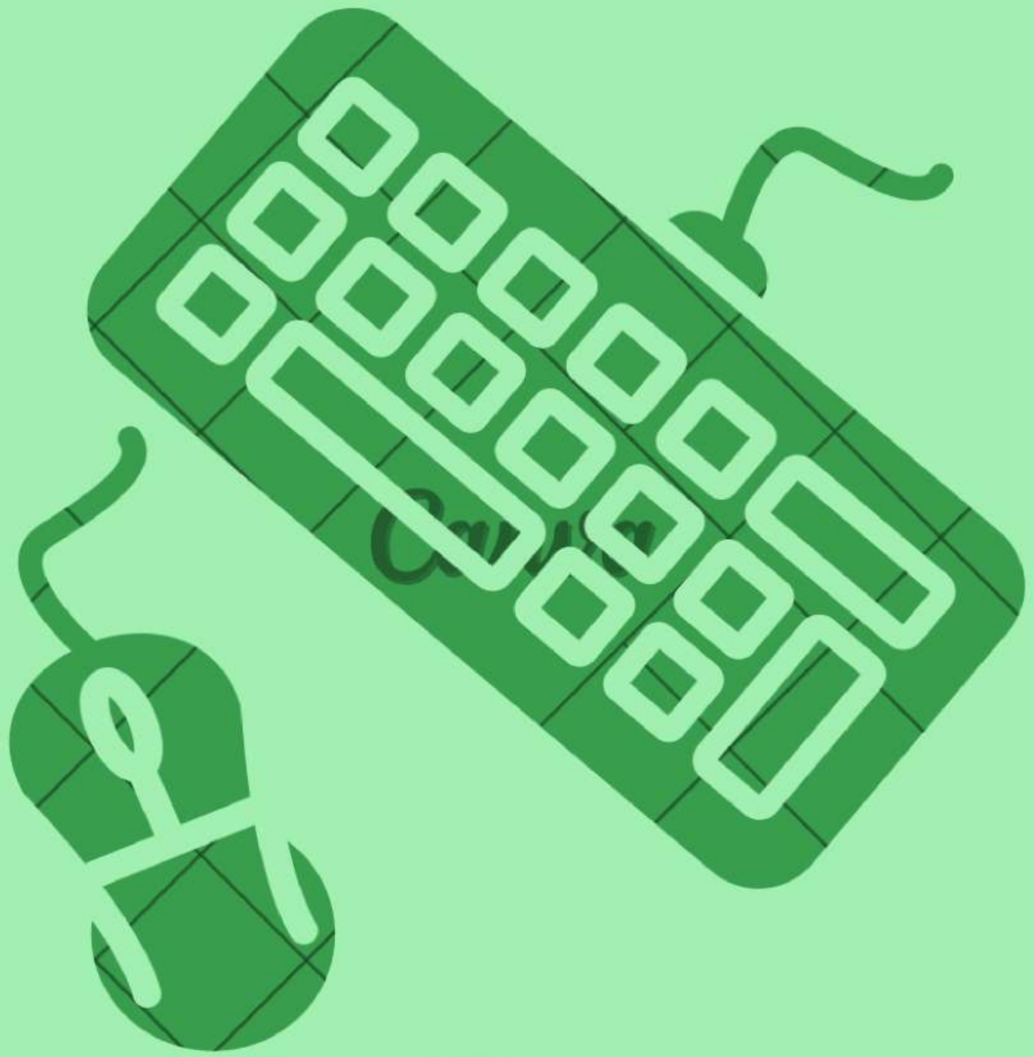
Data Manipulation

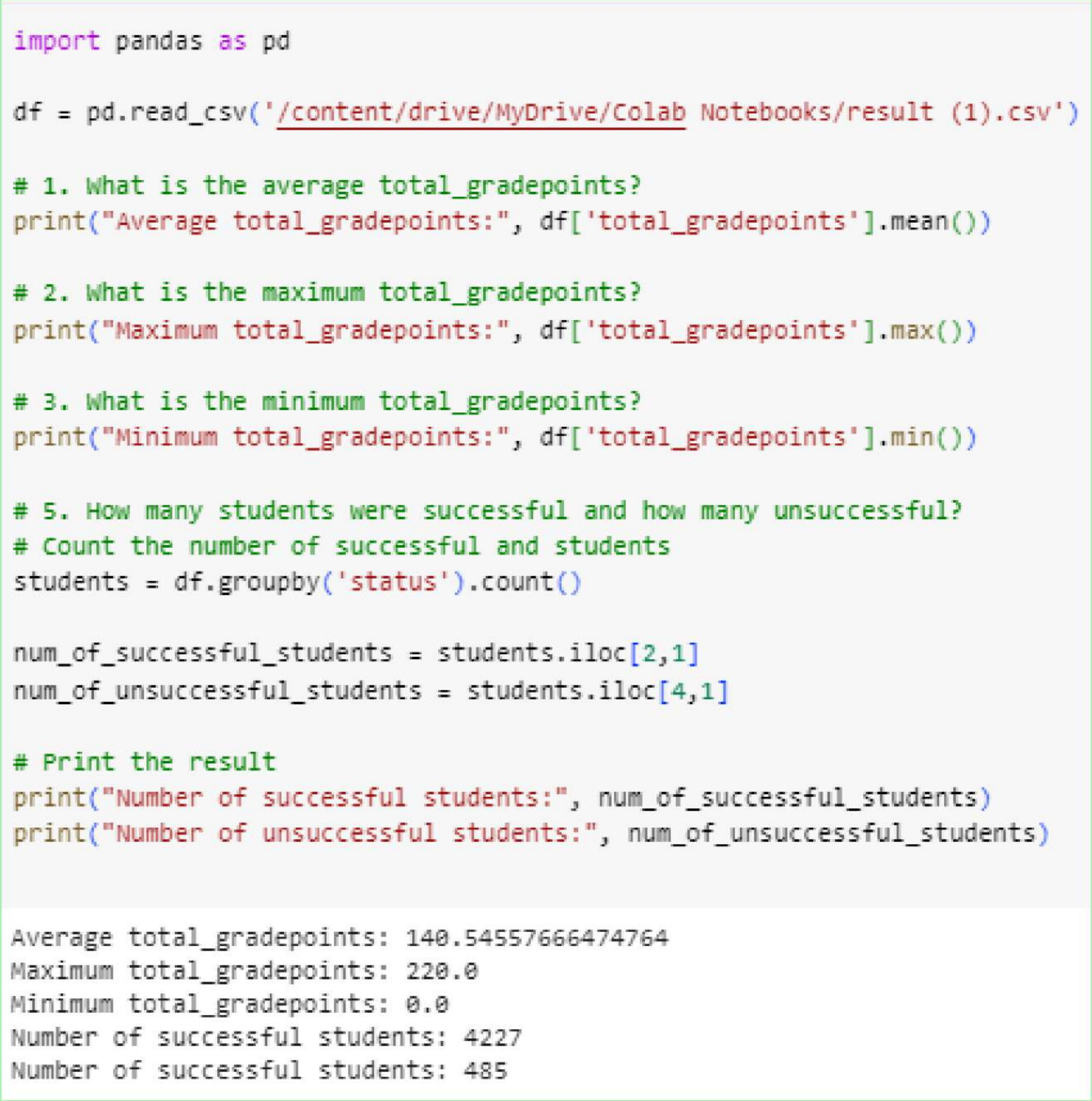
- Data manipulation prepares raw data.
- It ensures data quality and consistency.
- Missing values can be imputed.
- Outliers can be addressed.
- It makes data suitable for analysis.

Data Visualization

- Data visualization represents data visually.
- It communicates complex concepts.
- It reveals patterns and trends.
- It supports decision-making.
- It transforms complex data.

Data Manipulation

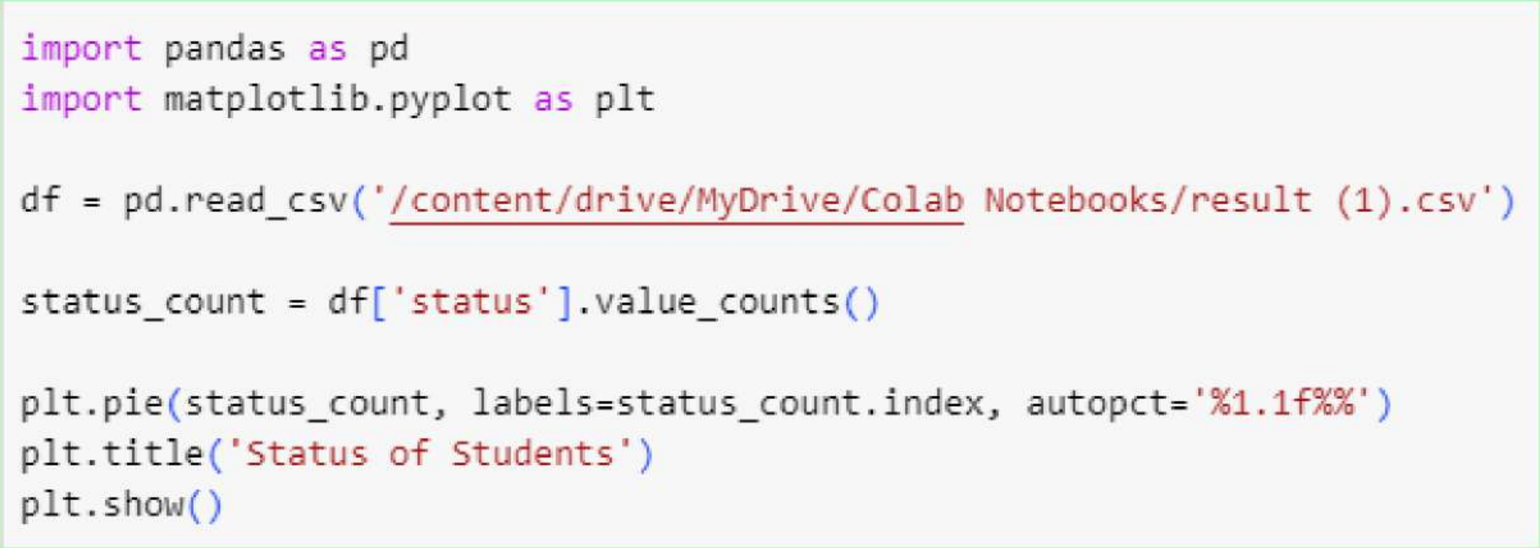




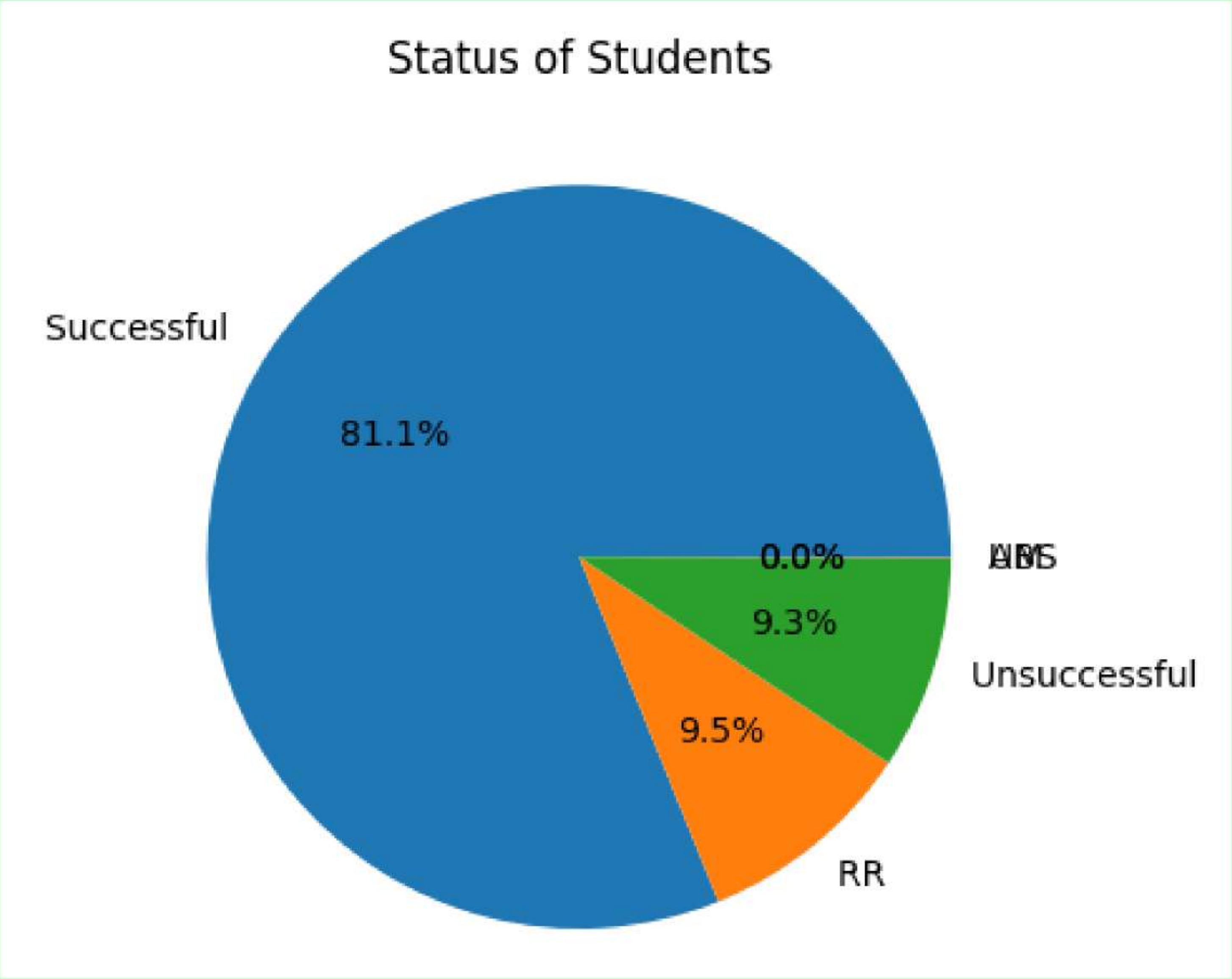
```
import pandas as pd

df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/result (1).csv')

# 1. What is the average total_gradepoints?
print("Average total_gradepoints:", df['total_gradepoints'].mean())

# 2. What is the maximum total_gradepoints?
print("Maximum total_gradepoints:", df['total_gradepoints'].max())

# 3. What is the minimum total_gradepoints?
print("Minimum total_gradepoints:", df['total_gradepoints'].min())

# 5. How many students were successful and how many unsuccessful?
# Count the number of successful and students
students = df.groupby('status').count()

num_of_successful_students = students.iloc[2,1]
num_of_unsuccessful_students = students.iloc[4,1]

# Print the result
print("Number of successful students:", num_of_successful_students)
print("Number of unsuccessful students:", num_of_unsuccessful_students)
```

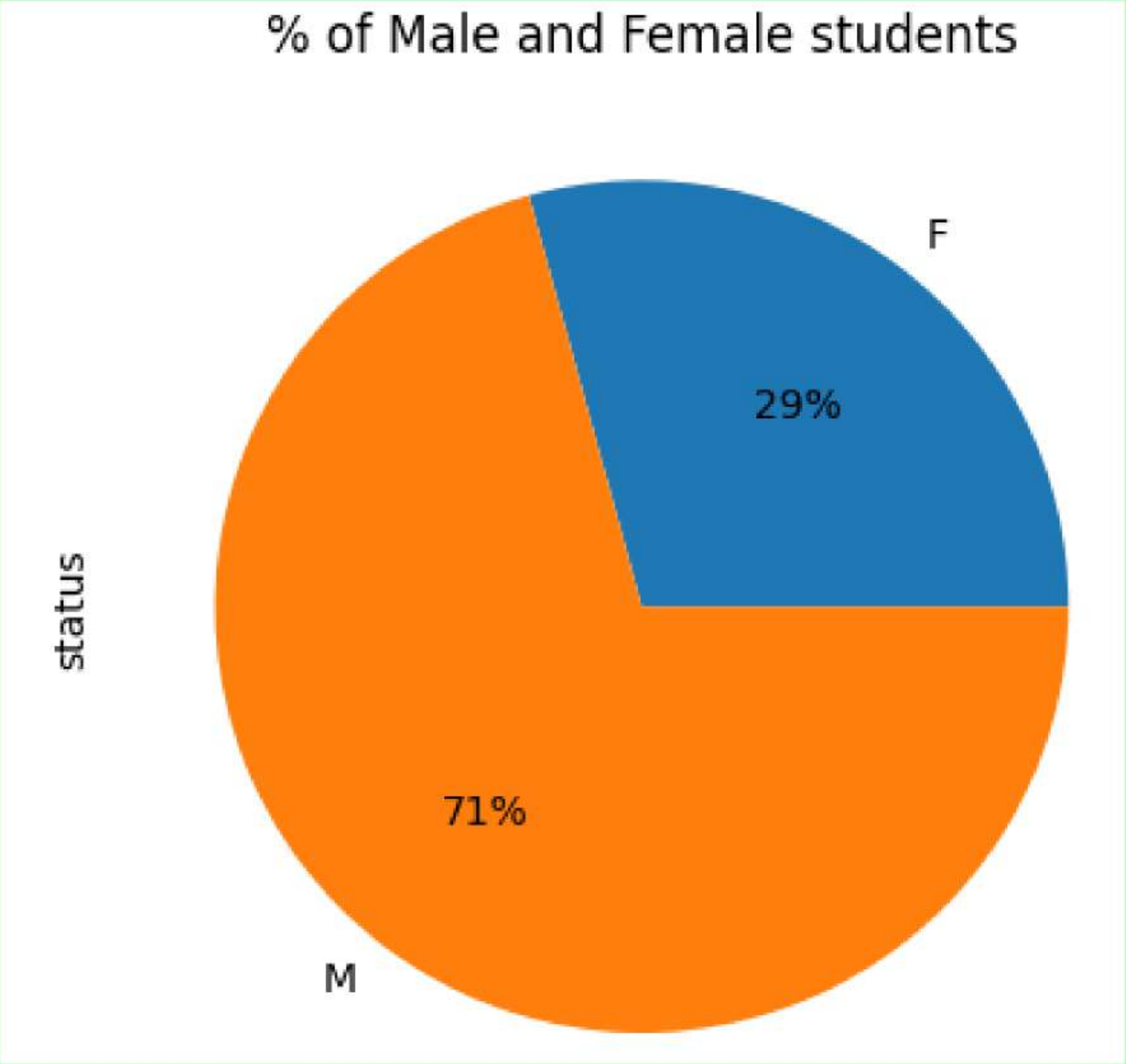
```
Average total_gradepoints: 140.54557666474764
Maximum total_gradepoints: 220.0
Minimum total_gradepoints: 0.0
Number of successful students: 4227
Number of successful students: 485
```

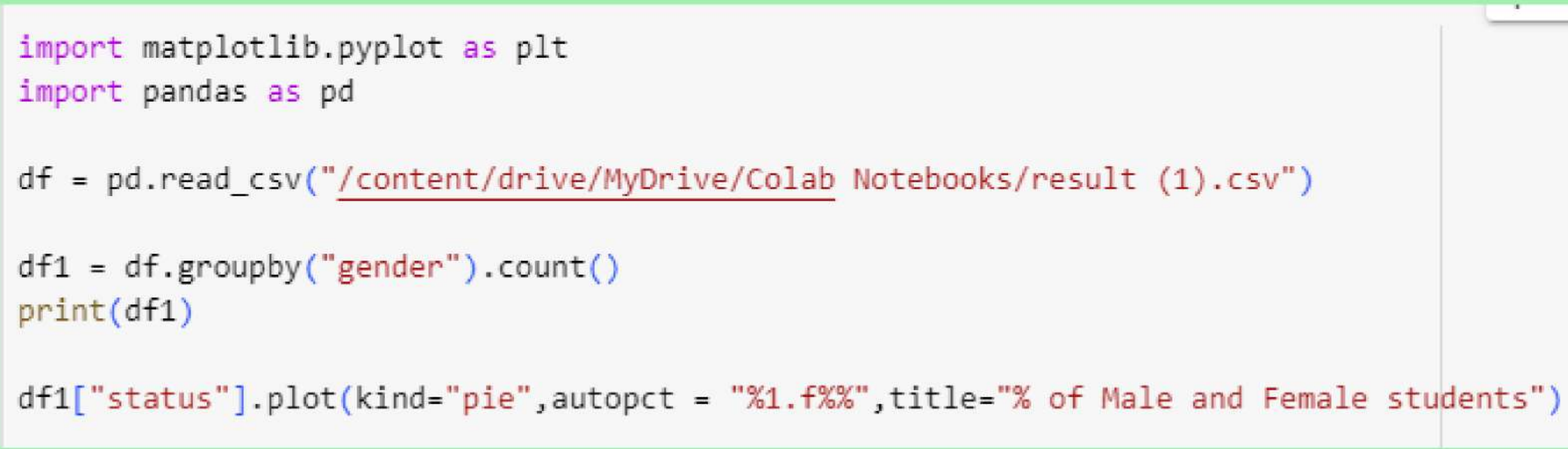
Data Visualization



```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/result (1).csv')

status_count = df['status'].value_counts()

plt.pie(status_count, labels=status_count.index, autopct='%1.1f%%')
plt.title('Status of Students')
plt.show()
```

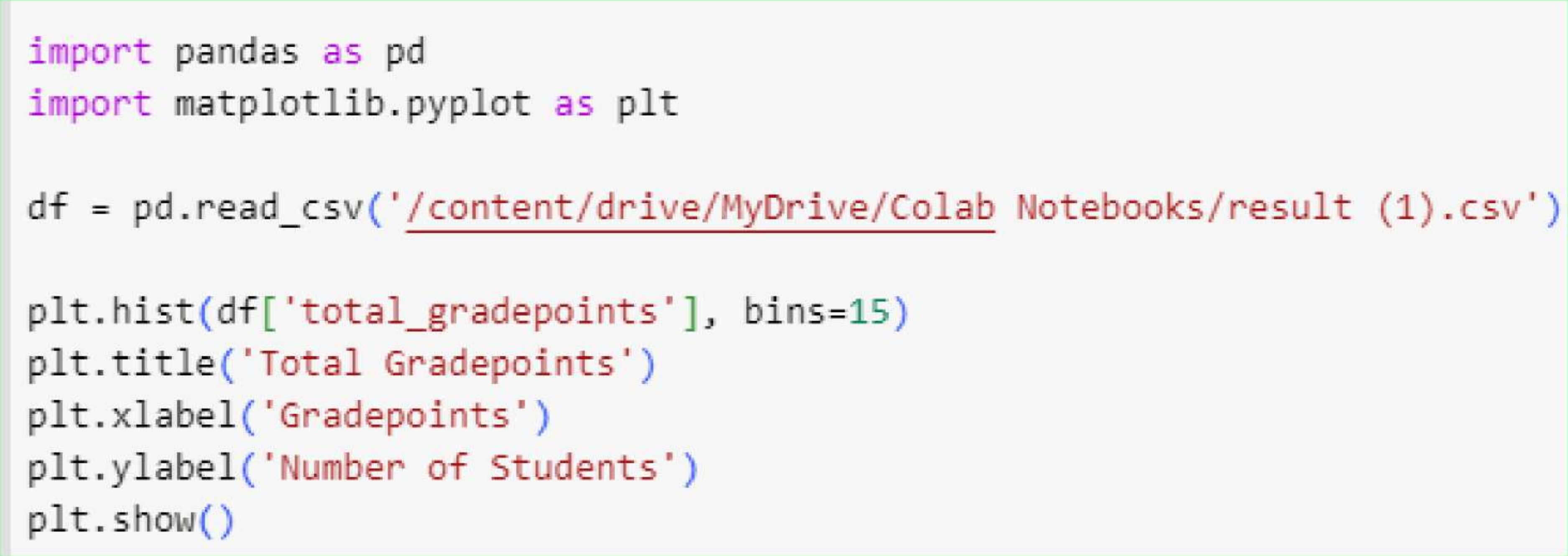




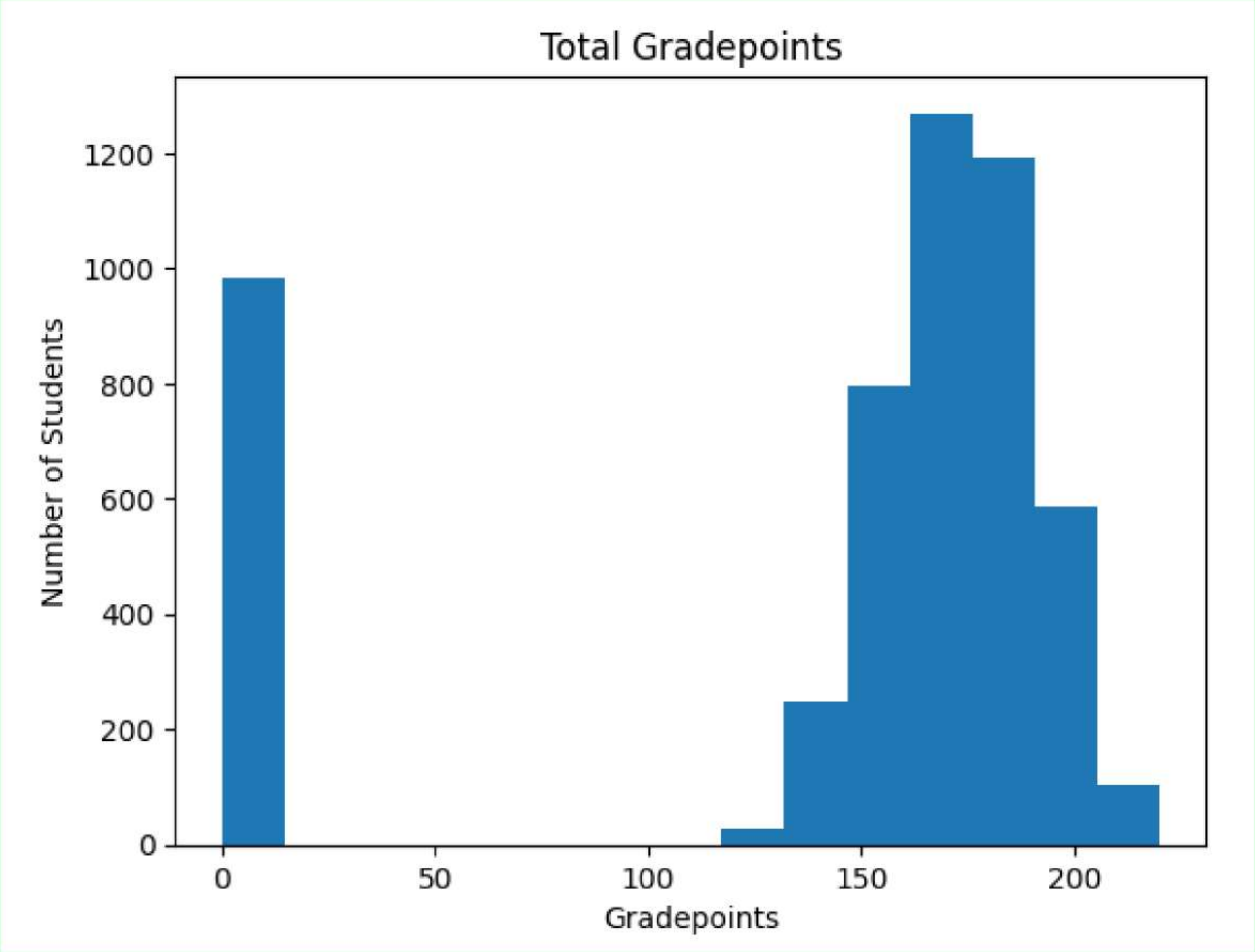


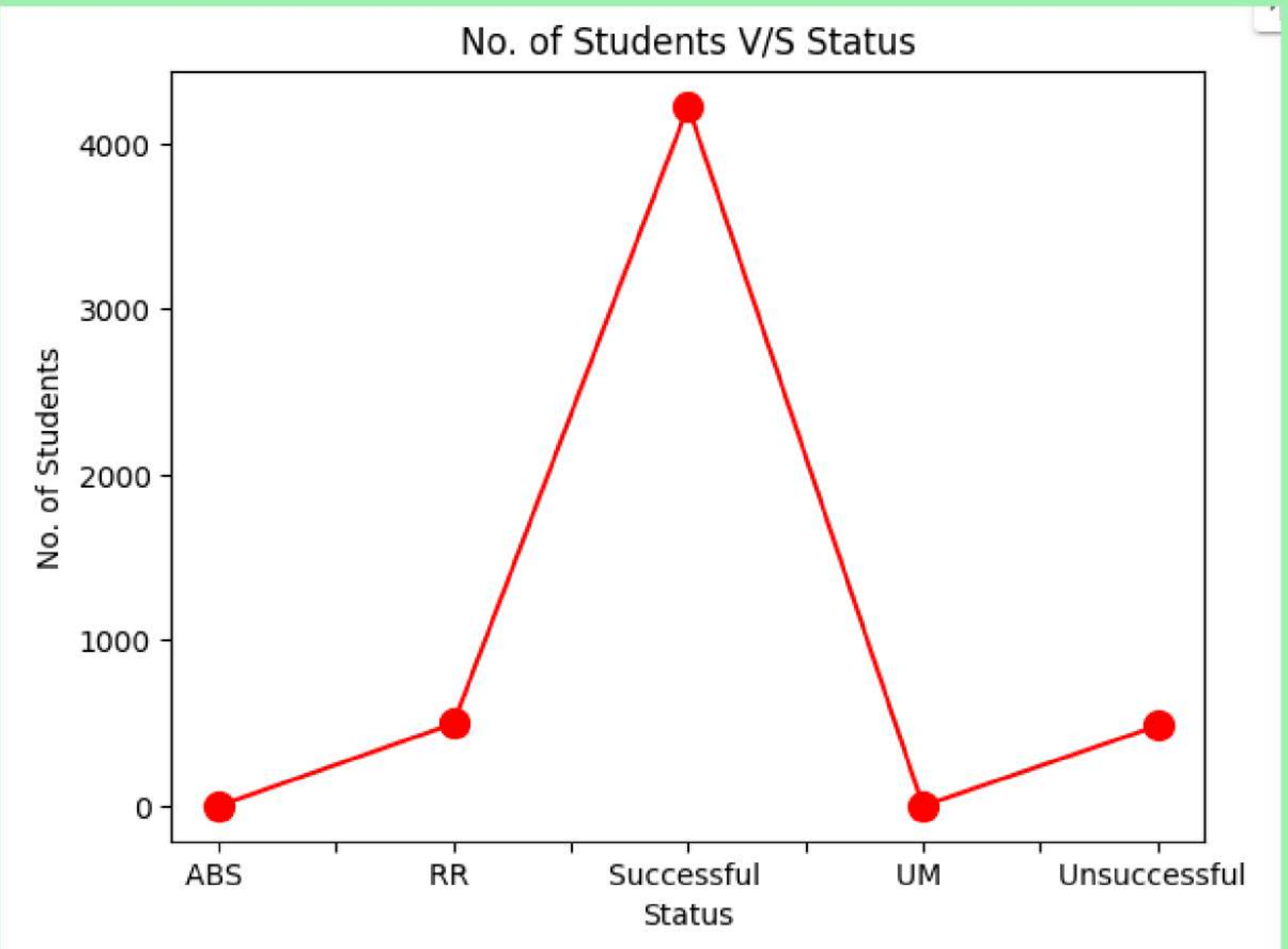
```
import matplotlib.pyplot as plt
import pandas as pd

df = pd.read_csv("/content/drive/MyDrive/Colab Notebooks/result (1).csv")

df1 = df.groupby("gender").count()
print(df1)

df1["status"].plot(kind="pie",autopct = "%1.f%%",title="% of Male and Female students")
```

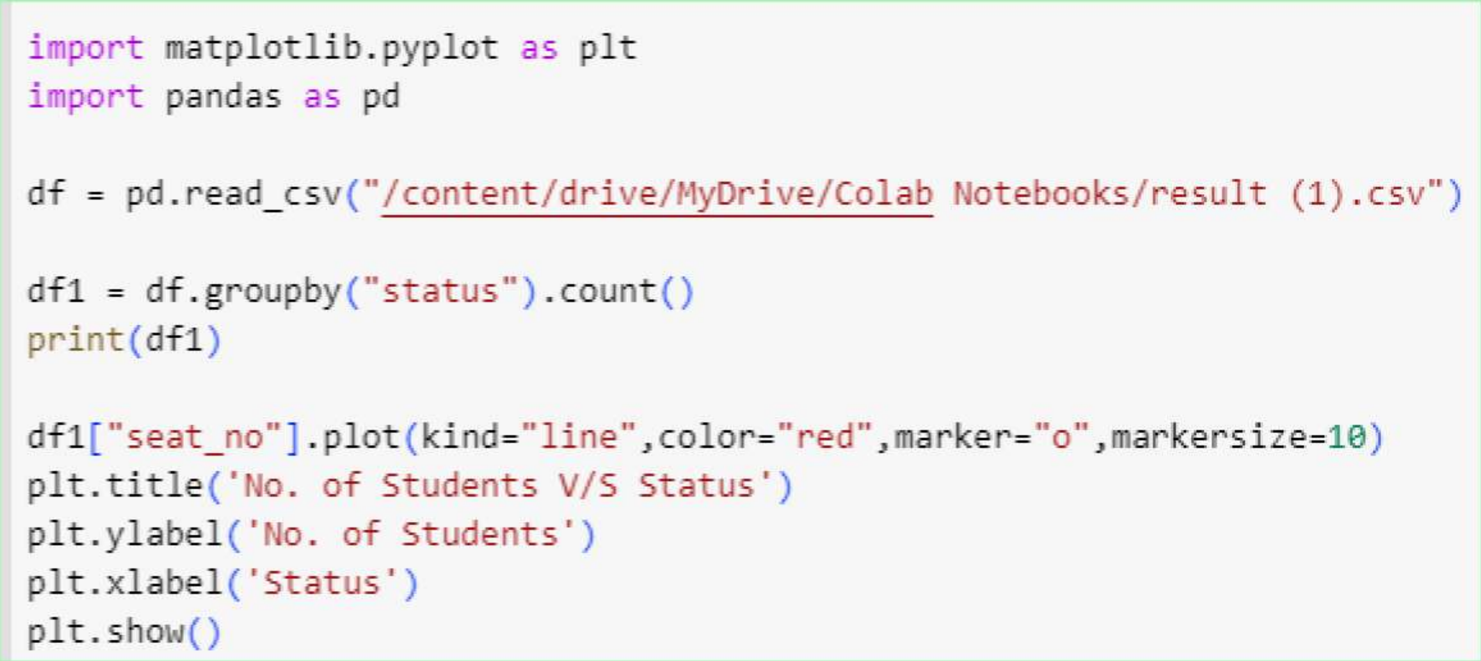
Data Visualization



```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/result (1).csv')

plt.hist(df['total_gradepoints'], bins=15)
plt.title('Total Gradepoints')
plt.xlabel('Gradepoints')
plt.ylabel('Number of Students')
plt.show()
```



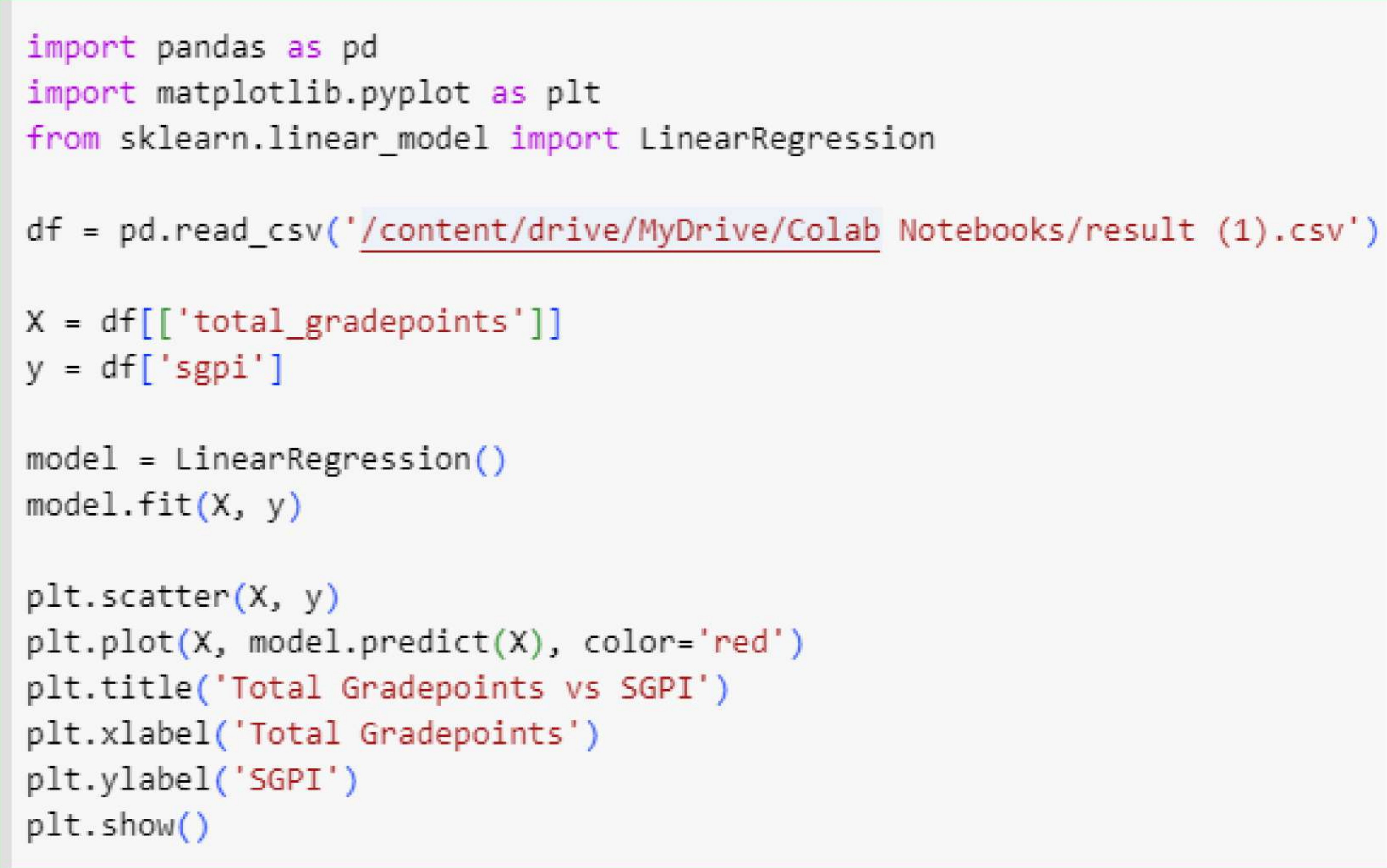




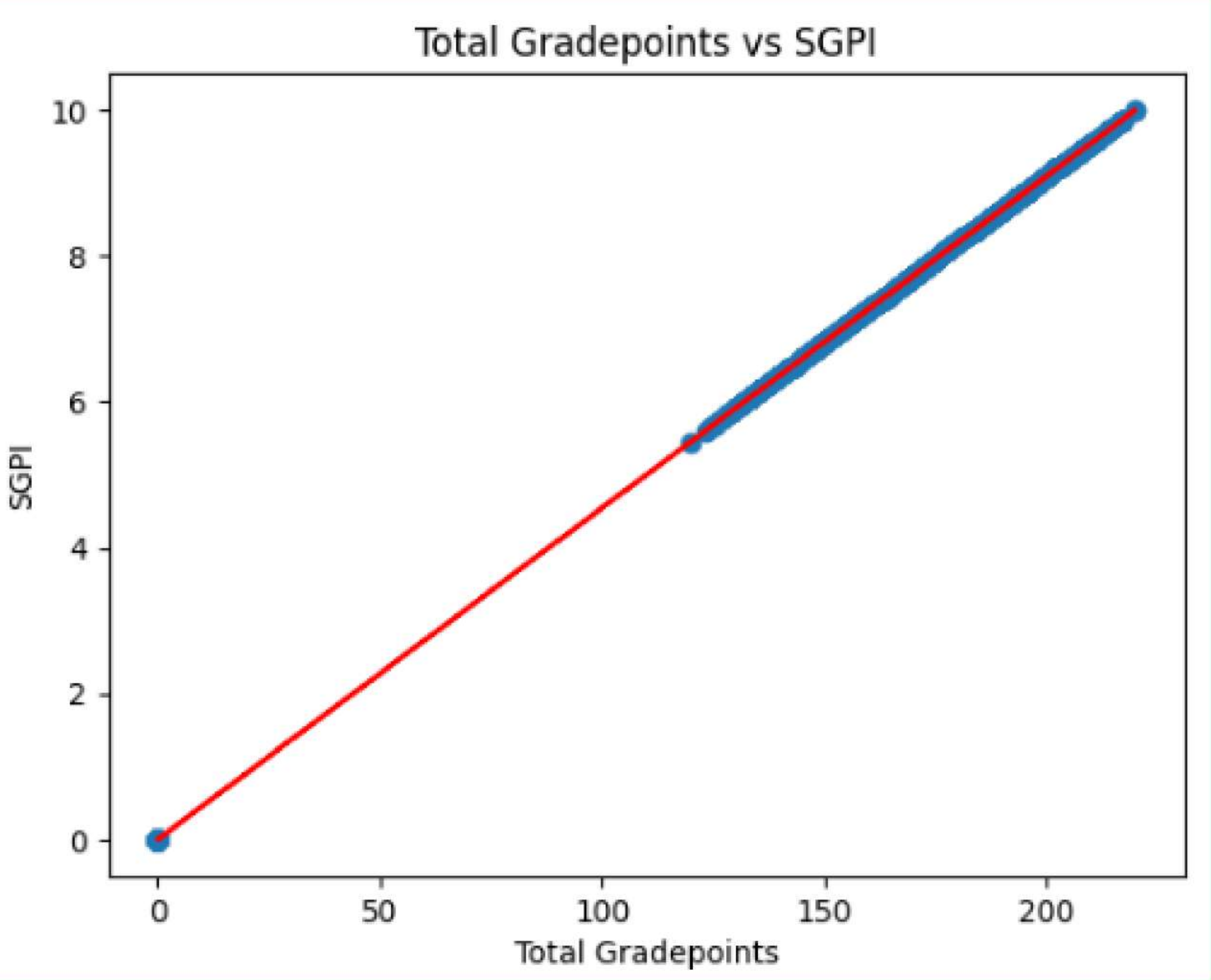
```
import matplotlib.pyplot as plt
import pandas as pd

df = pd.read_csv("/content/drive/MyDrive/Colab Notebooks/result (1).csv")

df1 = df.groupby("status").count()
print(df1)

df1["seat_no"].plot(kind="line",color="red",marker="o",markersize=10)
plt.title('No. of Students V/S Status')
plt.ylabel('No. of Students')
plt.xlabel('Status')
plt.show()
```

Predictive Technique

Linear Regression

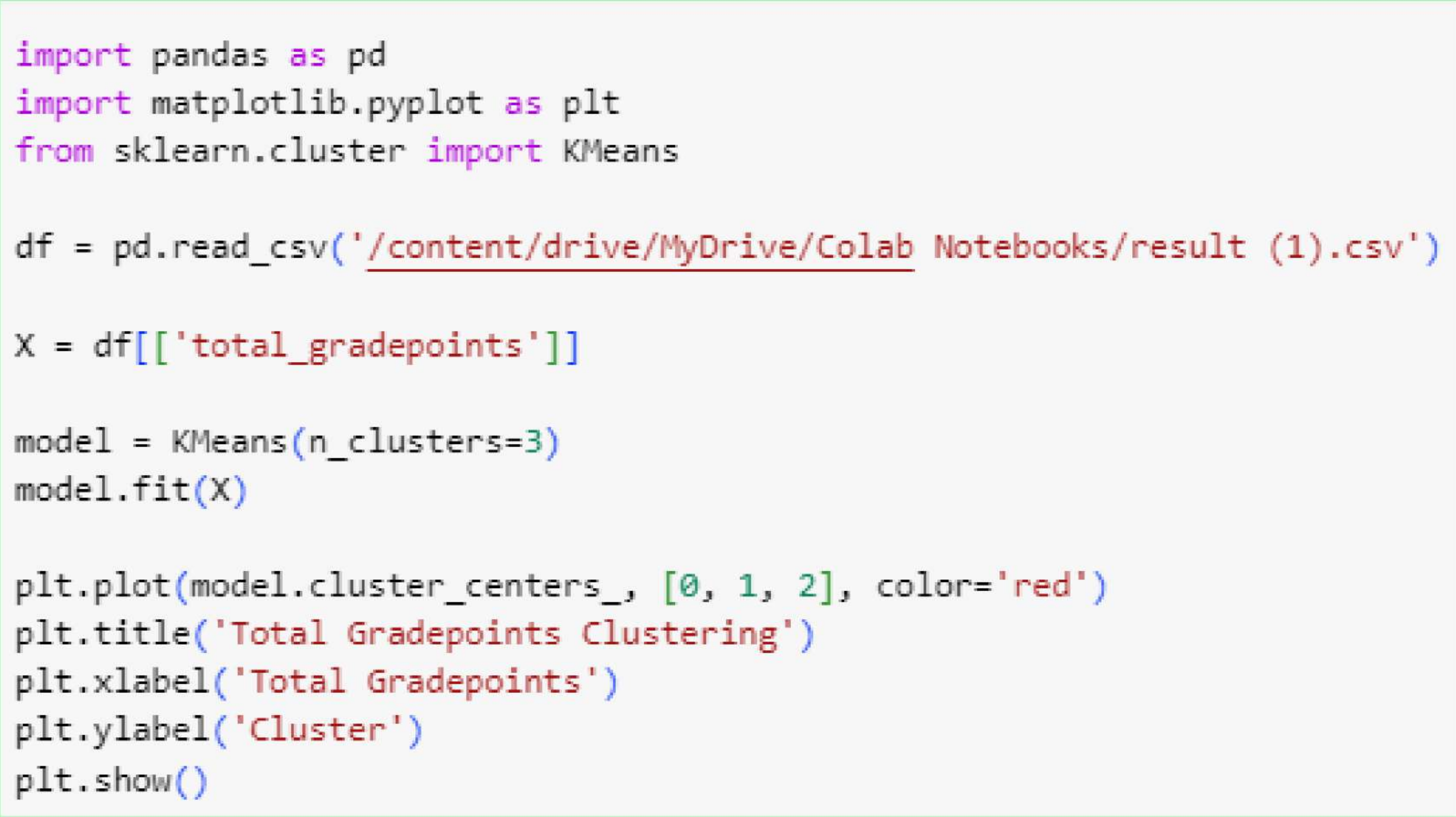


```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/result (1).csv')

X = df[['total_gradepoints']]
y = df['sgpi']

model = LinearRegression()
model.fit(X, y)

plt.scatter(X, y)
plt.plot(X, model.predict(X), color='red')
plt.title('Total Gradepoints vs SGPI')
plt.xlabel('Total Gradepoints')
plt.ylabel('SGPI')
plt.show()
```

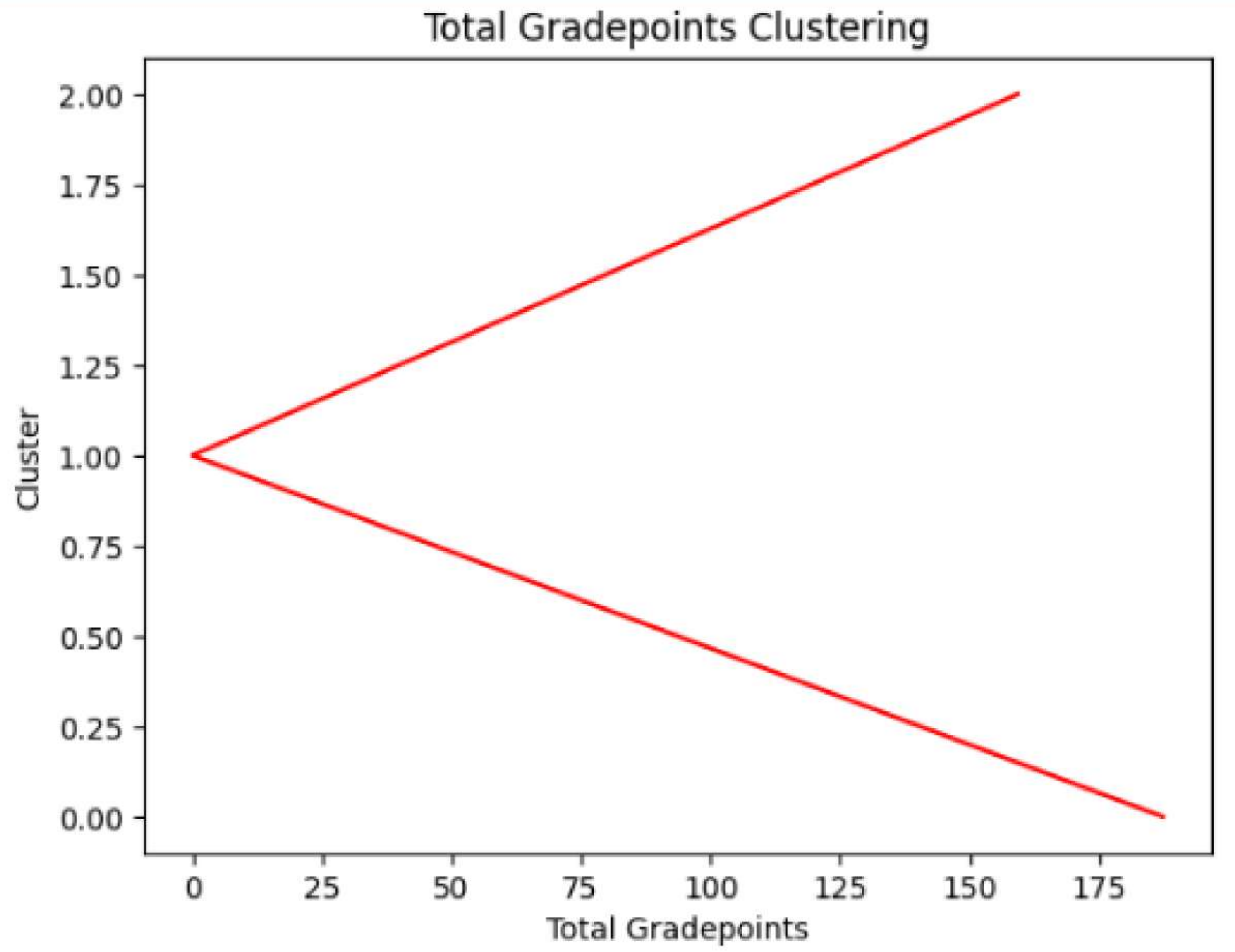


Predictive Technique

K-Means



```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans

df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/result (1).csv')

X = df[['total_gradepoints']]

model = KMeans(n_clusters=3)
model.fit(X)

plt.plot(model.cluster_centers_, [0, 1, 2], color='red')
plt.title('Total Gradepoints Clustering')
plt.xlabel('Total Gradepoints')
plt.ylabel('Cluster')
plt.show()
```



Application

1. **Pandas, NumPy, and Matplotlib** are widely used in data analysis and visualization in various fields such as finance, healthcare, and social media
2. **Nearest Neighbors (KNN)** is used for image recognition and recommender systems.
3. **Linear regression** is used for predicting stock prices and house prices.
4. **K-Means clustering** is used for customer segmentation and image compression.

References

1. K-Means Clustering in Python : A Practical Guide – Real
2. Python K-Means Clustering using Python - Medium
3. The k-Nearest Neighbors (kNN) Algorithm in Python – Real Python
4. Everything you need to Know about Linear Regression! - Analytics Vidhya

Conclusion

- In conclusion, our analysis of the University dataset has provided valuable insights into the Students and the factors influencing their Result.
- We discovered significant correlations between Marks and variables such as seat no., gender,center, and year
- The analysis highlighted the importance of Education, Marks disparities, and gender biases during this Exam.
- Through data cleaning, preprocessing, visualization, and modeling, we were able to extract meaningful information