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

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
import pandas as pd
df = pd.read_csv("TASK3 dataset.csv")
df.head()
```

	Unnamed: 0	Student ID	Well versed with the subject	Explains concepts in an understandable way	Use of presentations	Degree of difficulty of assignments	Solves doubts willingly	Structuring of the course	Provides support for students going above and beyond	Course recommendation based on relevance
0	0	340	5	2	7	6	9	2	1	8
1	1	253	6	5	8	6	2	1	2	9
2	2	680	7	7	6	5	4	2	3	1

```
df = df.drop(columns=["Unnamed: 0"])
df.head()
```

	Student ID	Well versed with the subject	Explains concepts in an understandable way	Use of presentations	Degree of difficulty of assignments	Solves doubts willingly	Structuring of the course	Provides support for students going above and beyond	Course recommendation based on relevance
0	340	5	2	7	6	9	2	1	8
1	253	6	5	8	6	2	1	2	9
2	680	7	7	6	5	4	2	3	1
3	806	9	6	7	1	5	9	4	6
4	632	8	10	8	4	6	6	9	9

```
df.isnull().sum()
```

	0
Student ID	0
Well versed with the subject	0
Explains concepts in an understandable way	0
Use of presentations	0
Degree of difficulty of assignments	0
Solves doubts willingly	0
Structuring of the course	0
Provides support for students going above and beyond	0
Course recommendation based on relevance	0

dtype: int64

```
df = df.dropna()
```

```
df.describe()
```

	Student ID	Well versed with the subject	Explains concepts in an understandable way	Use of presentations	Degree of difficulty of assignments	Solves doubts willingly	Structuring of the course	Provides support for students going above and beyond	Course recommendation based on relevance
count	1001.000000	1001.000000	1001.000000	1001.000000	1001.000000	1001.000000	1001.000000	1001.000000	1001.000000
mean	500.000000	7.497502	6.081918	5.942058	5.430569	5.474525	5.636364	5.662338	5.598402
std	289.108111	1.692998	2.597168	1.415853	2.869046	2.874648	2.920212	2.891690	2.886617
min	0.000000	5.000000	2.000000	4.000000	1.000000	1.000000	1.000000	1.000000	1.000000
25%	250.000000	6.000000	4.000000	5.000000	3.000000	3.000000	3.000000	3.000000	3.000000
50%	500.000000	8.000000	6.000000	6.000000	5.000000	6.000000	6.000000	6.000000	6.000000
75%	750.000000	9.000000	8.000000	7.000000	8.000000	8.000000	8.000000	8.000000	8.000000

```
df['Overall_Satisfaction'] = df.iloc[:,2:].mean(axis=1)
df['Overall_Satisfaction'].head()
```

	Overall_Satisfaction
0	5.000000
1	4.714286
2	4.000000
3	5.428571
4	7.428571

dtype: float64

```
df['Overall_Satisfaction'].mean()
```

```
np.float64(5.689453403739118)
```

```
df.mean().sort_values(ascending=False)
```

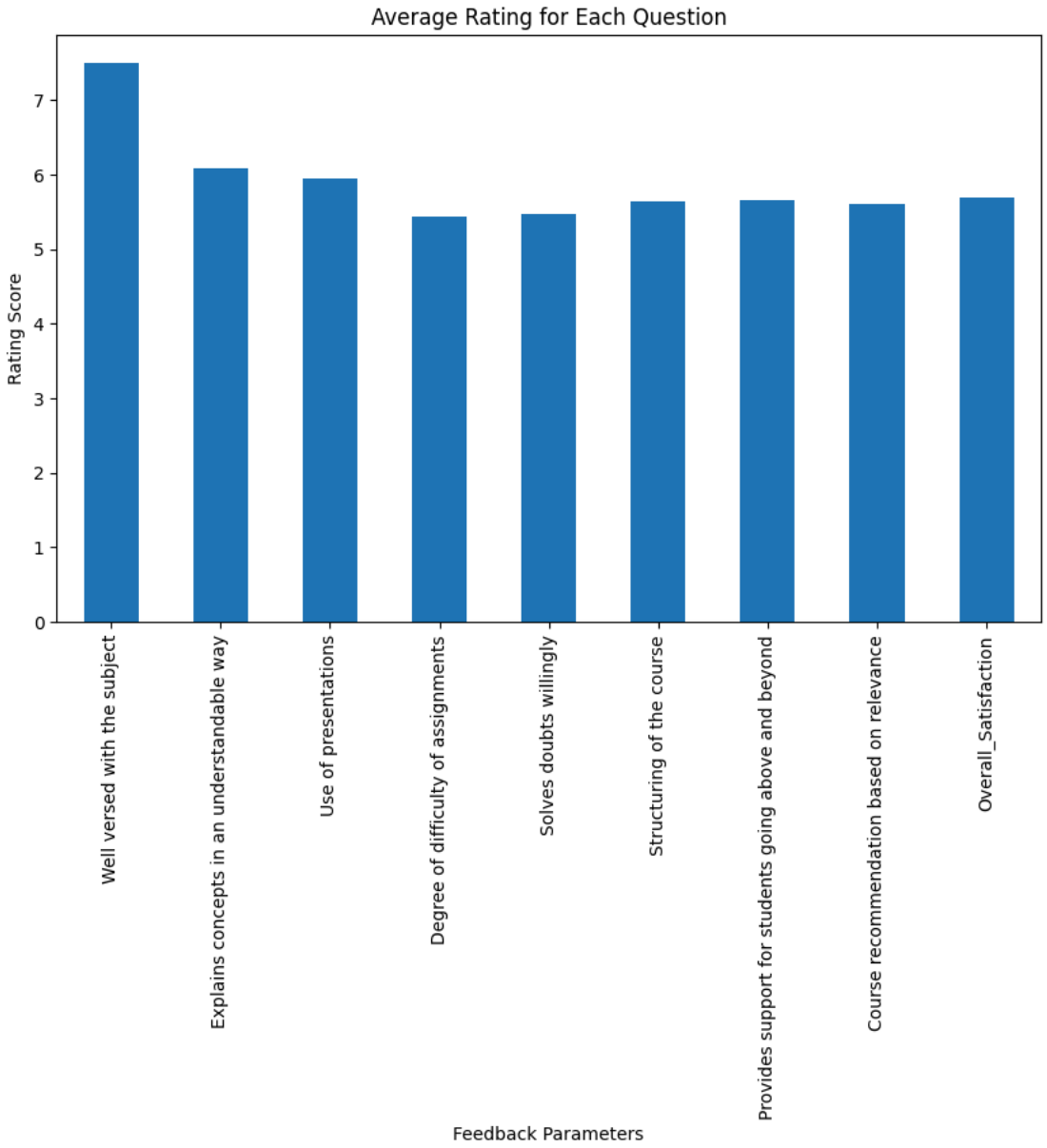
	0
Student ID	500.000000
Well versed with the subject	7.497502
Explains concepts in an understandable way	6.081918
Use of presentations	5.942058
Overall_Satisfaction	5.689453
Provides support for students going above and beyond	5.662338
Structuring of the course	5.636364
Course recommendation based on relevance	5.598402
Solves doubts willingly	5.474525
Degree of difficulty of assignments	5.430569

dtype: float64

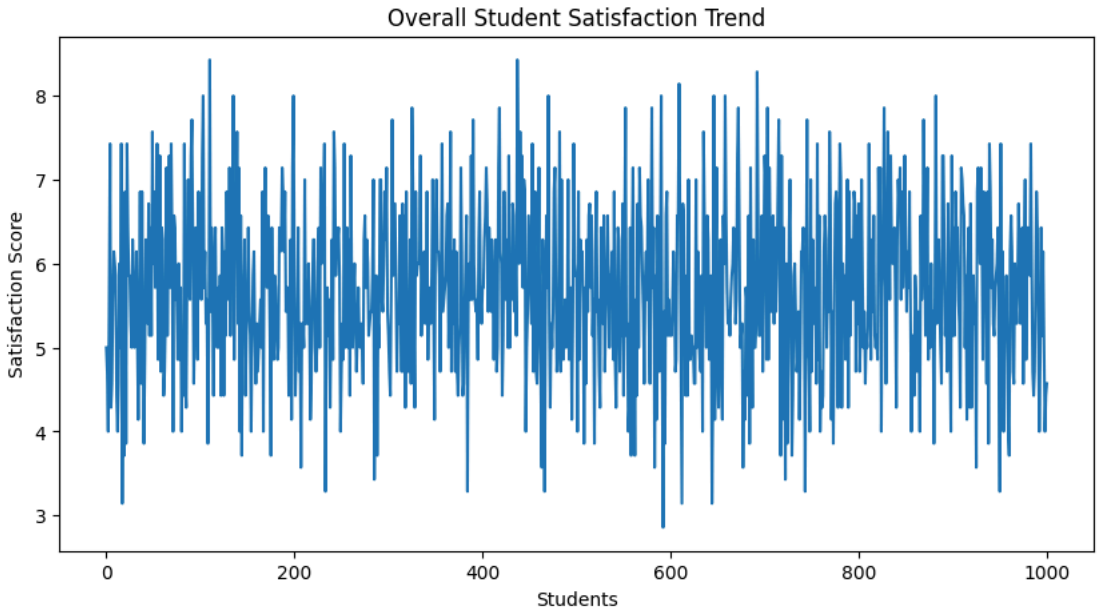
```
import matplotlib.pyplot as plt

df.iloc[:,1:].mean().plot(kind='bar', figsize=(10,6))
plt.title("Average Rating for Each Question")
plt.ylabel("Rating Score")
plt.xlabel("Feedback Parameters")
```

```
plt.show()
```




```
df['Overall_Satisfaction'].plot(kind='line', figsize=(10,5))
plt.title("Overall Student Satisfaction Trend")
plt.ylabel("Satisfaction Score")
plt.xlabel("Students")
plt.show()
```




```
df.columns
```

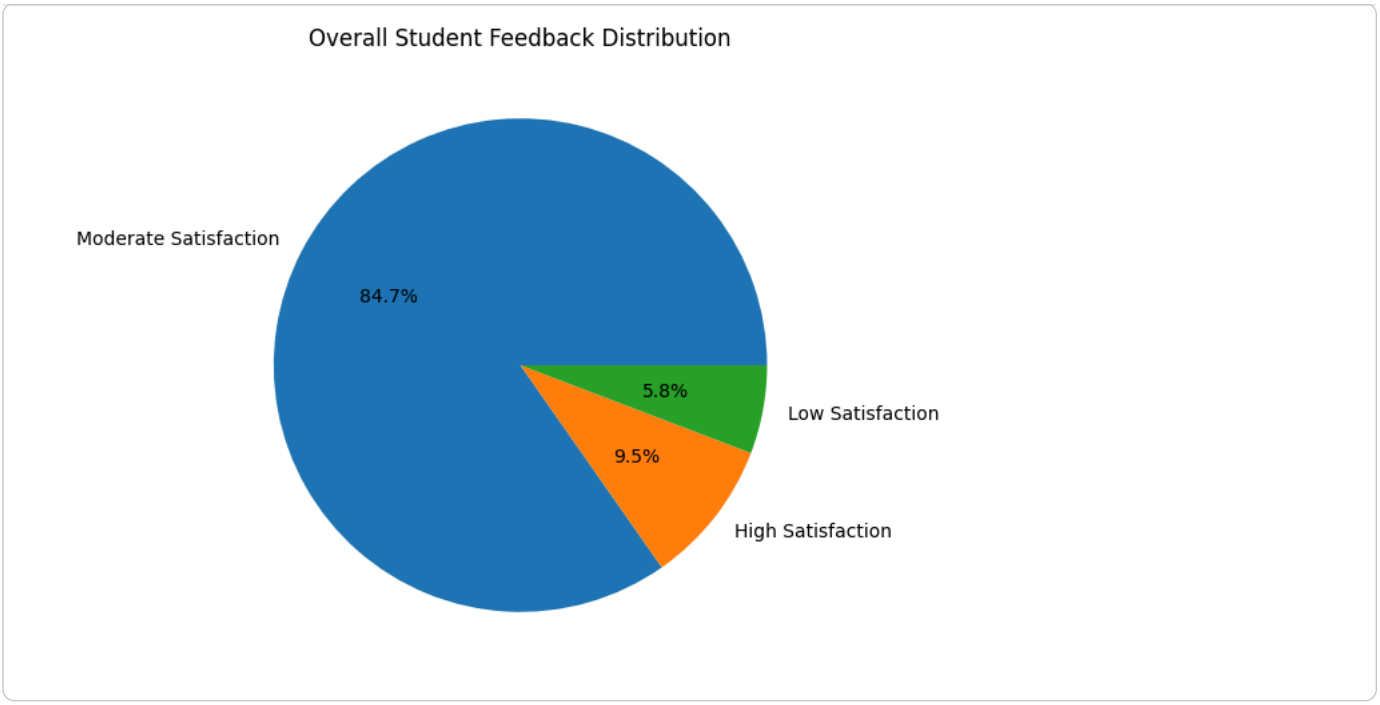
```
Index(['Student ID', 'Well versed with the subject',
       'Explains concepts in an understandable way', 'Use of presentations',
       'Degree of difficulty of assignments', 'Solves doubts willingly',
       'Structuring of the course',
       'Provides support for students going above and beyond',
       'Course recommendation based on relevance', 'Overall_Satisfaction'],
      dtype='object')
```

```
# Create Overall Feedback Score (Average)
df['Overall_Feedback'] = df.iloc[:, 2:].mean(axis=1)

# Convert score into categories
df['Feedback_Category'] = pd.cut(
    df['Overall_Feedback'],
    bins=[0,4,7,10],
    labels=['Low Satisfaction','Moderate Satisfaction','High Satisfaction']
)

# Plot Pie Chart
df['Feedback_Category'].value_counts().plot(
    kind='pie',
    autopct='%1.1f%%',
    figsize=(6,6)
)

plt.title("Overall Student Feedback Distribution")
plt.ylabel("")
plt.show()
```



Insights & Findings from Student Feedback Analysis

1 Average Rating for Each Question

- Teachers are well-versed with the subject (highest rating ~7.5)
- Concept explanation and presentation usage are around 6, meaning students understand but improvement is possible.
- Assignment difficulty and doubt solving scored around 5.4–5.6 indicating students face challenges.
- Course structure and student support are slightly below expectation.
- Overall Satisfaction is around 5.7, meaning students are generally satisfied but not highly satisfied.

2 Overall Student Satisfaction Trend

- Satisfaction varies significantly from student to student.
- Most scores range between 5 and 8.
- Few students have very low satisfaction (<4), showing dissatisfaction cases.
- Student experience is inconsistent and needs improvement.

3 Overall Satisfaction Distribution

- Majority of students fall under Moderate Satisfaction.
- A smaller percentage have High Satisfaction.
- A very small portion show Low Satisfaction.

Conclusion & Recommendations

Conclusion:

- Students are mostly moderately satisfied with the course.
- Teaching knowledge is strong but engagement and clarity need improvement.
- Assignments feel difficult for many students.
- Student support beyond classes should be increased.

Recommendations:

- Provide clearer explanations and interactive learning.
- Simplify assignments or provide proper guidance.
- Increase doubt-solving sessions.
- Improve course structure and relevance.

Final Summary – Student Feedback Internship Project

This project analyzes student feedback data to understand how students feel about teaching quality, course structure, assignments, and overall learning experience.

Using Python (Pandas & Matplotlib), I performed: ✓ Data loading & cleaning
✓ Understanding key feedback parameters
✓ Visualization using charts
✓ Insights generation
✓ Recommendations for improvement

The analysis helps the institution understand student satisfaction levels and areas where improvement is required to enhance the learning experience.

Key Insights

- Most students fall under **moderate satisfaction**
- Faculty subject knowledge is strong
- Need improvement in:
    - explaining concepts clearly