1st project

Import all lib

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

Read student csv file

```
df=pd.read_csv("student.csv")
df
```

```
    id         name  class  mark  gender
0    1     John Deo   Four    75  female
1    2     Max Ruin  Three    85    male
2    3       Arnold  Three    55    male
3    4   Krish Star   Four    60  female
4    5    John Mike   Four    60  female
5    6    Alex John   Four    55    male
6    7  My John Rob  Fifth    78    male
7    8       Asruid   Five    85    male
8    9      Tes Qry    Six    78    male
9   10     Big John   Four    55  female
10  11       Ronald    Six    89  female
11  12        Recky    Six    94  female
12  13          Kty  Seven    88  female
13  14         Bigy  Seven    88  female
14  15     Tade Row   Four    88    male
15  16        Gimmy   Four    88    male
16  17        Tumyu    Six    54    male
17  18        Honny   Five    75    male
18  19        Tinny   Nine    18    male
19  20       Jackly   Nine    65  female
20  21  Babby John   Four    69  female
21  22       Reggid  Seven    55  female
22  23        Herod  Eight    79    male
23  24    Tiddy Now  Seven    78    male
24  25     Giff Tow  Seven    88    male
25  26       Crelea  Seven    79    male
26  27     Big Nose  Three    81  female
27  28    Rojj Base  Seven    86  female
28  29  Tess Played  Seven    55    male
29  30    Reppy Red    Six    79  female
30  31  Marry Toeey   Four    88    male
31  32    Binn Rott  Seven    90  female
32  33    Kenn Rein    Six    96  female
```

```
33  34      Gain Toe  Seven    69    male
34  35    Rows Noump    Six    88  female
```

Explore the data

```
# Display the first few rows of the dataset
print(df.head())
```

```
   id        name  class  mark  gender
0   1    John Deo   Four    75  female
1   2    Max Ruin  Three    85    male
2   3      Arnold  Three    55    male
3   4  Krish Star   Four    60  female
4   5   John Mike   Four    60  female
```

```
# Check the data types of columns
print(df.dtypes)
```

```
id         int64
name      object
class     object
mark       int64
gender    object
dtype: object
```

```
# Check for missing values
print(df.isnull().sum())
```

```
id        0
name      0
class     0
mark      0
gender    0
dtype: int64
```

```
# Summary statistics
print(df.describe())
```

```
              id       mark
count  35.000000  35.000000
mean   18.000000  74.657143
std    10.246951  16.401117
min     1.000000  18.000000
25%     9.500000  62.500000
50%    18.000000  79.000000
75%    26.500000  88.000000
max    35.000000  96.000000
```

Perform data anlysis

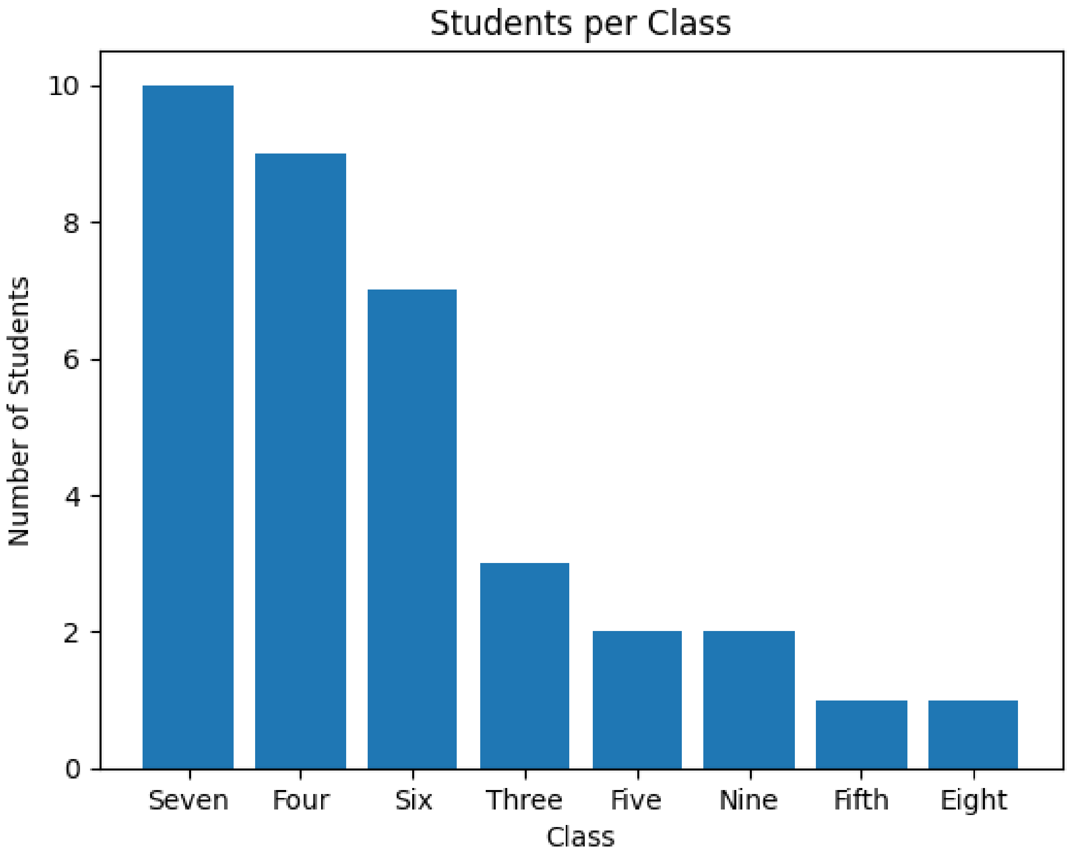
```
# Calculate the average mark
average_mark = df['mark'].mean()
print("Average Mark:", average_mark)
```

```
Average Mark: 74.65714285714286
```

```
# Count of students in each class
class_counts = df['class'].value_counts()
print("Students per Class:")
print(class_counts)
```

```
Students per Class:
class
Seven    10
Four      9
Six       7
Three     3
Five      2
Nine      2
Fifth     1
Eight     1
Name: count, dtype: int64
```

```
# Create a bar plot for Students per Class
plt.bar(class_counts.index, class_counts.values)
plt.xlabel("Class")
plt.ylabel("Number of Students")
plt.title("Students per Class")
plt.show()
```



Gender base analysis:

```
# Calculate the average mark for each gender
average_mark_gender = df.groupby('gender')['mark'].mean()
print("Average Mark per Gender:")
print(average_mark_gender)
```

```
Average Mark per Gender:
gender
female    77.529412
male      71.944444
Name: mark, dtype: float64
```

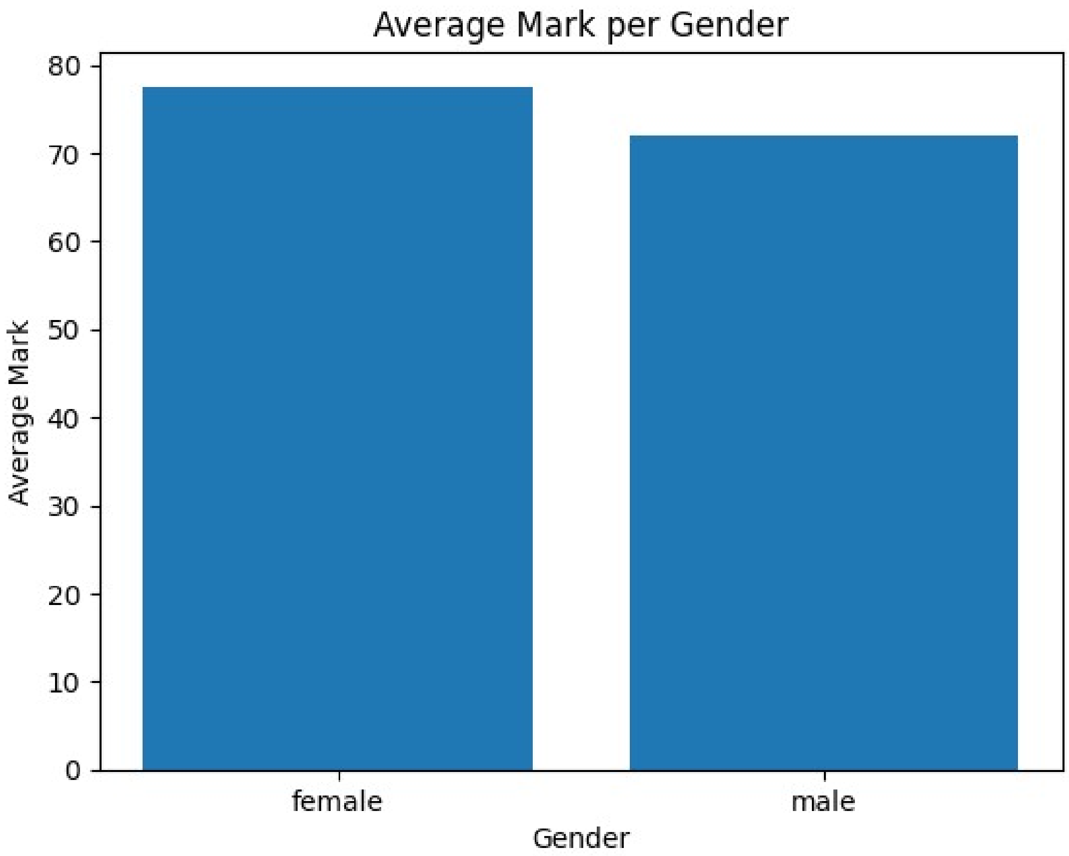
```
# Count of students for each gender
gender_counts = df['gender'].value_counts()
print("Number of Students per Gender:")
print(gender_counts)
```

```
Number of Students per Gender:
gender
male      18
```

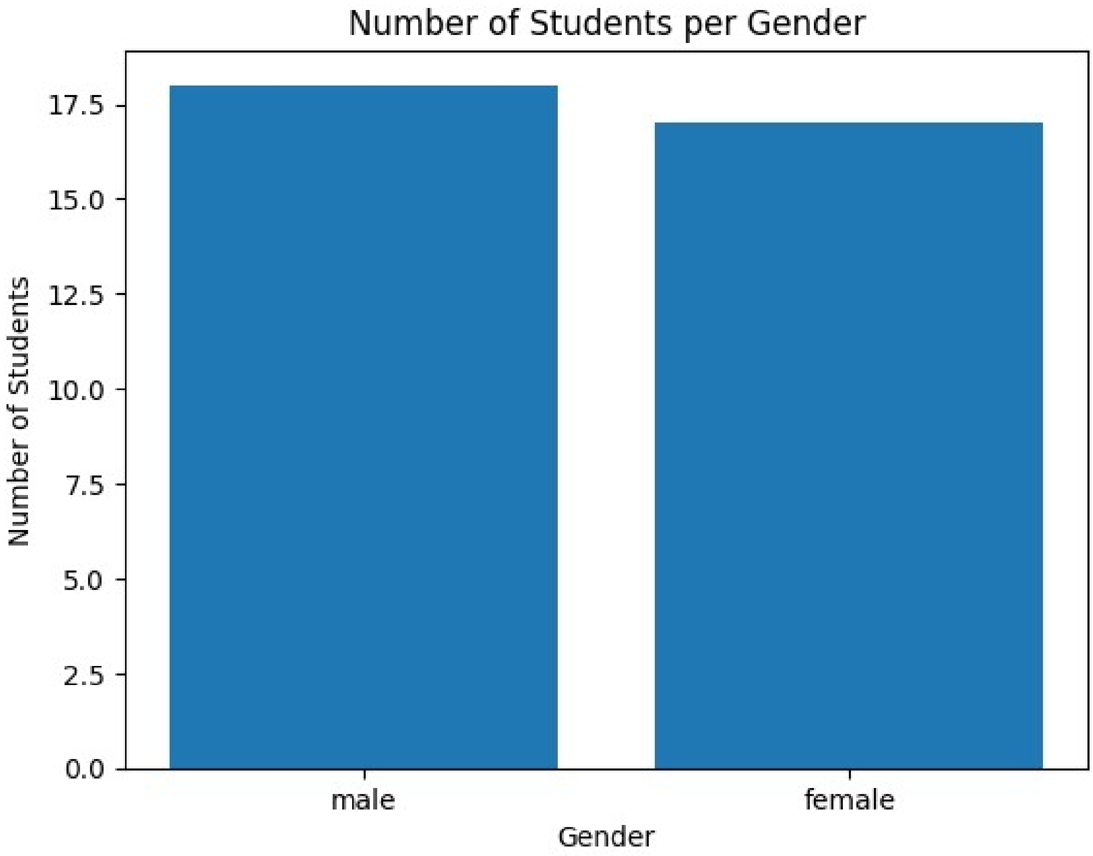
```
female    17
Name: count, dtype: int64
```

Data Visualization for Gender Analysis :

```
# Create a bar plot for Average Mark per Gender
plt.bar(average_mark_gender.index, average_mark_gender.values)
plt.xlabel("Gender")
plt.ylabel("Average Mark")
plt.title("Average Mark per Gender")
plt.show()
```



```
# Create a bar plot for Number of Students per Gender
plt.bar(gender_counts.index, gender_counts.values)
plt.xlabel("Gender")
plt.ylabel("Number of Students")
plt.title("Number of Students per Gender")
plt.show()
```



the maximum and minimum marks obtained by students and identify the student with the highest and lowest marks:

```python
# Maximum and minimum marks obtained by students
max_mark = df['mark'].max()
min_mark = df['mark'].min()
print("Maximum Mark:", max_mark)
print("Minimum Mark:", min_mark)
```

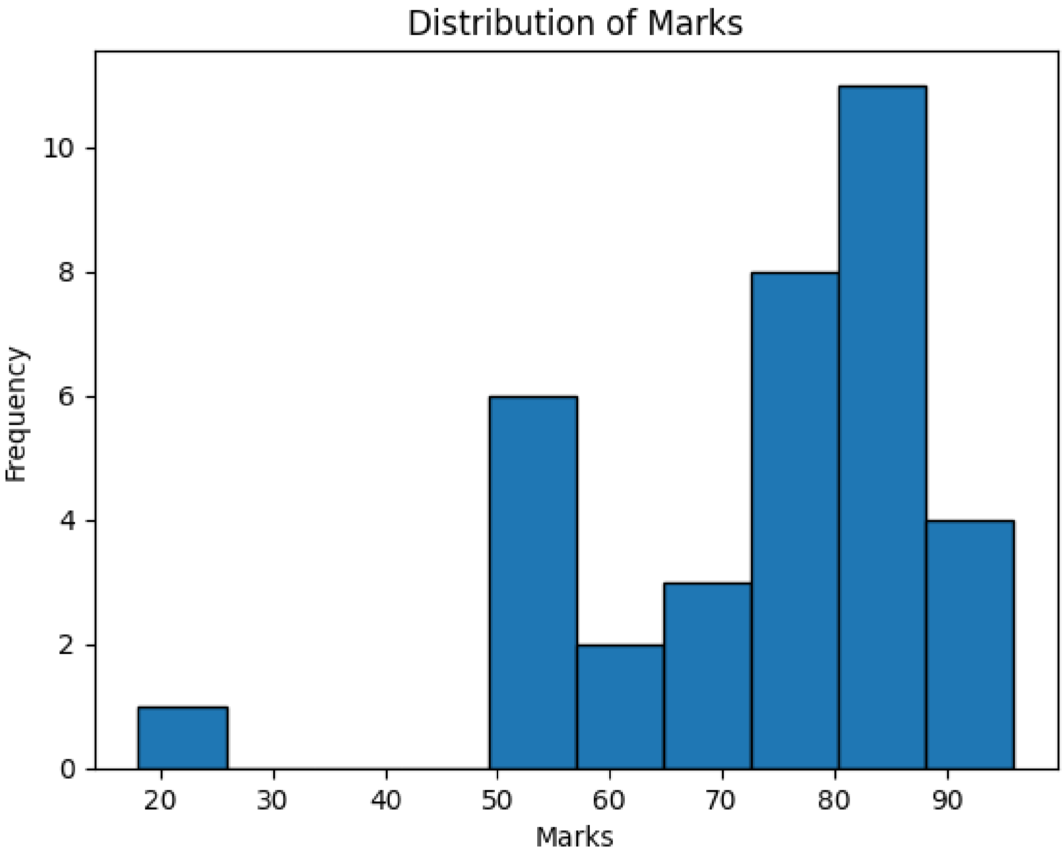
```
Maximum Mark: 96
Minimum Mark: 18
```

```python
# Student with the highest and lowest marks
highest_mark_student = df.loc[df['mark'].idxmax()]
lowest_mark_student = df.loc[df['mark'].idxmin()]

print("Student with the Highest Mark:")
print(highest_mark_student)
print("\nStudent with the Lowest Mark:")
print(lowest_mark_student)
```

```
Student with the Highest Mark:
id                33
name       Kenn Rein
class            Six
mark              96
gender        female
Name: 32, dtype: object

Student with the Lowest Mark:
id           19
name      Tinny
class      Nine
mark         18
gender     male
Name: 18, dtype: object
```

Let's visualize the distribution of marks using a histogram:

```
# Create a histogram for marks
plt.hist(df['mark'], bins=10, edgecolor='black')
plt.xlabel("Marks")
plt.ylabel("Frequency")
plt.title("Distribution of Marks")
plt.show()
```



Class-wise Analysis:

```
# Calculate average mark for each class
average_mark_class = df.groupby('class')['mark'].mean()
print("Average Mark per Class:")
print(average_mark_class)

# Highest mark in each class
highest_mark_class = df.groupby('class')['mark'].max()
print("\nHighest Mark per Class:")
print(highest_mark_class)

# Lowest mark in each class
lowest_mark_class = df.groupby('class')['mark'].min()
print("\nLowest Mark per Class:")
print(lowest_mark_class)
```

```
Average Mark per Class:
class
Eight    79.000000
Fifth    78.000000
Five     80.000000
```

```
Four      70.888889
Nine      41.500000
Seven     77.600000
Six       82.571429
Three     73.666667
Name: mark, dtype: float64

Highest Mark per Class:
class
Eight     79
Fifth     78
Five      85
Four      88
Nine      65
Seven     90
Six       96
Three     85
Name: mark, dtype: int64

Lowest Mark per Class:
class
Eight     79
Fifth     78
Five      75
Four      55
Nine      18
Seven     55
Six       54
Three     55
Name: mark, dtype: int64
```

Gender and Class-based Analysis:

```
# Calculate average mark for each gender within each class
average_mark_gender_class = df.groupby(['class', 'gender'])
['mark'].mean()
print("\nAverage Mark per Gender and Class:")
print(average_mark_gender_class)
```

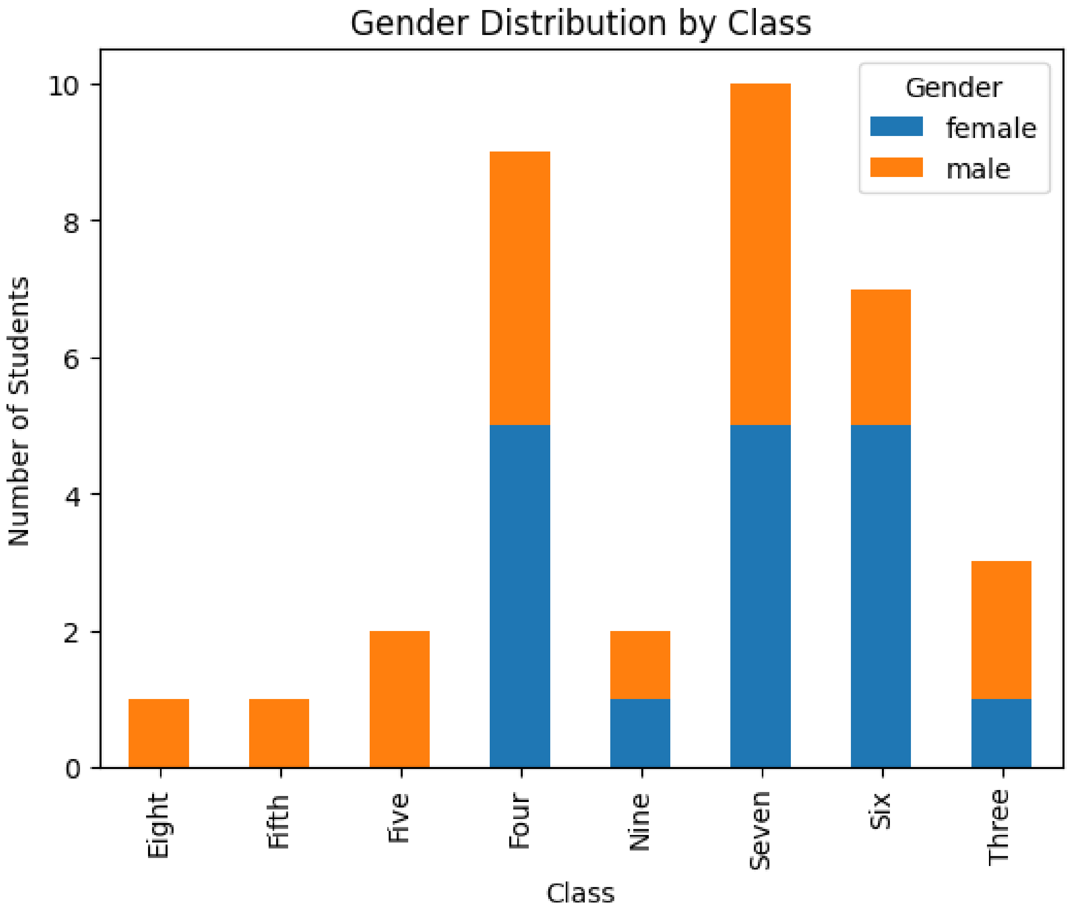
```
Average Mark per Gender and Class:
class  gender
Eight  male      79.00
Fifth  male      78.00
Five   male      80.00
Four   female    63.80
       male      79.75
Nine   female    65.00
```

```
       male      18.00
Seven  female    81.40
       male      73.80
Six    female    89.20
       male      66.00
Three  female    81.00
       male      70.00
Name: mark, dtype: float64
```

Gender Distribution by Class:

```
# Gender distribution within each class
gender_class_counts = df.groupby(['class', 'gender']).size().unstack()
print("\nGender Distribution by Class:")
print(gender_class_counts)

# Create a stacked bar plot for gender distribution within each class
gender_class_counts.plot(kind='bar', stacked=True)
plt.xlabel("Class")
plt.ylabel("Number of Students")
plt.title("Gender Distribution by Class")
plt.legend(title="Gender", loc="upper right")
plt.show()
```

```
Gender Distribution by Class:
gender  female  male
class
Eight      NaN   1.0
Fifth      NaN   1.0
Five       NaN   2.0
Four       5.0   4.0
Nine       1.0   1.0
Seven      5.0   5.0
Six        5.0   2.0
Three      1.0   2.0
```



Top-performing Class:

```
# Class with the highest average mark
top_class = average_mark_class.idxmax()
print("\nClass with the Highest Average Mark:", top_class)


Class with the Highest Average Mark: Six
```

Analysis:

Conclusion

1)The student dataset provides valuable insights into the academic performance of students in different classes. We observed variations in the average marks, highest marks, and lowest marks among the classes. Additionally, we analyzed the performance of students based on gender within each class.

2)The dataset includes top-performing students who achieved exceptional marks, and the gender distribution within each class shows varying student compositions. Class Six has the highest average mark, indicating better overall academic performance among its students.

3)The analysis can aid educators, administrators, and parents in understanding students' performance and identifying areas for improvement in different classes. It demonstrates the power of data analysis using Numpy, Pandas, and Matplotlib to gain meaningful insights from datasets and make data-driven decisions in education and other fields.