Part 1 - Basic Questions to analyse Student’s data

8 question have been attempted below using python programming to explore and gain insights on Student’s data

Python code to load and read data:

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read\_excel('Data analyst Data.xlsx')

df.head()

1. How many unique students are included in the dataset

Python Code:

# Considering email id as primary key

df['Email ID'].nunique()

Output:

2157

Explanation:

nunique() function counts the number of unique entries, therefore in all there are 2157 unique students in the dataset.

1. What is the Average CGPA of Student

Python code:

'''Since many students have entered different CGPA for different event

so taking avg for each student then taking complete avg of entire data'''

# Grouping data by primary key(email id) and calculate average CGPA

grouped\_data = df.groupby('Email ID')['CGPA'].mean()

# Convert the result back to a DataFrame

average\_cgpa\_data = grouped\_data.reset\_index()

# Print the resulting DataFrame

print('AVG CGPA of all students: \n')

print(average\_cgpa\_data)

print()

print('Overall AVG CGPA: \n')

print(average\_cgpa\_data['CGPA'].mean())

Output:

AVG CGPA of all students:

Email ID CGPA

0 [10pawan@xyz.com](mailto:10pawan@xyz.com) 7.30

1 [Shubhada@xyz.com](mailto:Shubhada@xyz.com) 7.50

2 [aaditya@xyz.com](mailto:aaditya@xyz.com) 7.30

3 [aaftab@xyz.com](mailto:aaftab@xyz.com) 7.30

4 [aakanksha@xyz.com](mailto:aakanksha@xyz.com) 8.10

... ... ...

2152 [zeel@xyz.com](mailto:zeel@xyz.com) 7.95

2153 [zeenal@xyz.com](mailto:zeenal@xyz.com) 7.60

2154 [zobia@xyz.com](mailto:zobia@xyz.com) 6.50

2155 [zoya@xyz.com](mailto:zoya@xyz.com) 7.90

2156 [zulema@xyz.com](mailto:zulema@xyz.com) 8.10

[2157 rows x 2 columns]

Overall AVG CGPA:

8.039655011464745

Explanation:

Multiple entries of same students should have the same CGPA but its not the case here so we are grouping by email id and then taking average of cgpa for a particular student this will overcome the issue of multiple entries and then we are taking overall average for all 2157 students.

1. What s the Distribution of students across different Graduation years?

Python Code:

# Select the maximum graduation year for each unique email ID

unique\_years = df.groupby('Email ID')['Year of Graduation'].max()

# Calculate the distribution of students across different graduation years

year\_distribution = unique\_years.value\_counts().sort\_index()

# Convert the result to a DataFrame

year\_distribution\_data = pd.DataFrame({'Year of Graduation': year\_distribution.index, 'Student Count': year\_distribution.values})

# Print the resulting DataFrame

print(year\_distribution\_data)

Output:

Year of Graduation Student Count

0 2023 392

1 2024 574

2 2025 727

3 2026 464

Python Visualization Code:

# Plotting the distribution using a bar plot

plt.figure(figsize=(8, 6))

bars = plt.bar(year\_distribution\_data['Year of Graduation'], year\_distribution\_data['Student Count'])

plt.xlabel('Year of Graduation')

plt.ylabel('Student Count')

plt.title('Distribution of Students Across Different Graduation Years')

plt.xticks([int(year) for year in year\_distribution\_data['Year of Graduation']])

for bar in bars:

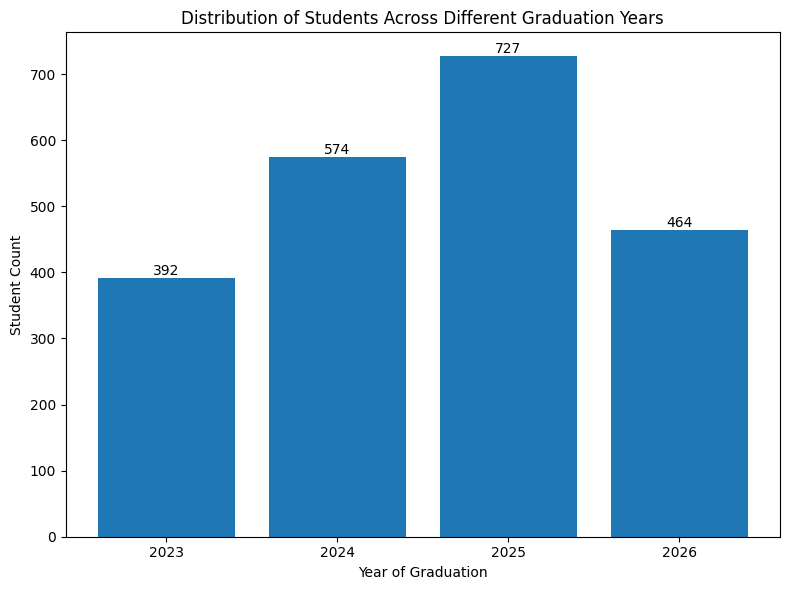
    yval = bar.get\_height()

    plt.text(bar.get\_x() + bar.get\_width()/2, yval, round(yval, 2), ha='center', va='bottom')

plt.tight\_layout()

plt.show()

Visualization:



1. What s the Distribution of students experience with python programming?

Python Code:

grouped\_py = df.groupby('Email ID')['Experience with python (Months)'].apply(lambda x: x.mode().iloc[0])

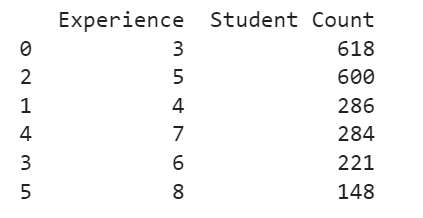
month\_distribution = grouped\_py.value\_counts().sort\_index()

month\_distribution\_data = pd.DataFrame({'Experience': month\_distribution.index, 'Student Count': month\_distribution.values})

month\_distribution\_data = month\_distribution\_data.sort\_values(by='Student Count', ascending=False)

print(month\_distribution\_data)

Output:



Python Visualization Code:

plt.figure(figsize=(8,6))

explode = [0.1 if i==0 else 0 for i in range(len(month\_distribution\_data))]

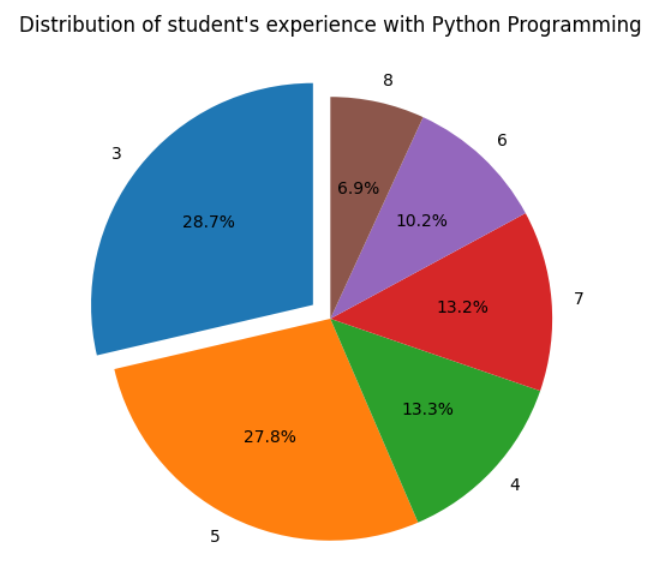
plt.pie(month\_distribution\_data['Student Count'], labels=month\_distribution\_data['Experience'], autopct = '%1.1f%%', startangle =90, explode = explode)

# plt.legend(title='Experience', loc='upper right', labels=month\_distribution\_data['Experience'])

plt.title("Distribution of student's experience with Python Programming")

plt.show()

Visualization:



1. What s the avg family income of student?

Python Code:

# Calculate average family income for each student

average\_family\_income = df.groupby('Email ID')['Family Income'].mean()

# Calculate overall average family income

overall\_average\_income = average\_family\_income.mean()

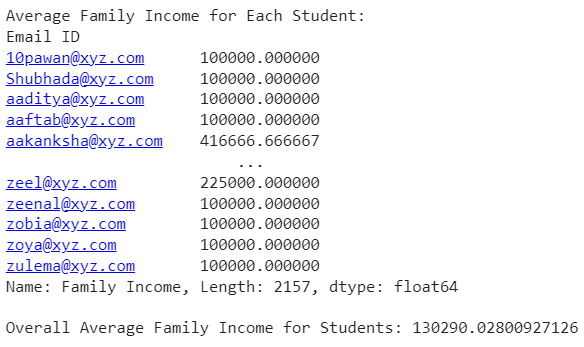
# Print the resulting averages

print("Average Family Income for Each Student:")

print(average\_family\_income)

print("\nOverall Average Family Income for Students:", overall\_average\_income)

Output:



1. How does the GPA varies among diferent colleges (Show top 5 results only)

Python Code:

avg\_gpa\_by\_col = df.groupby('College Name')['CGPA'].mean()

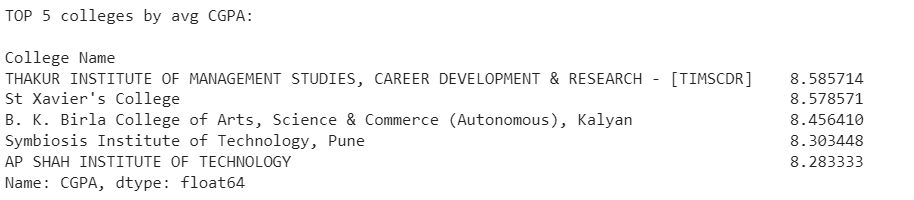
sorted\_colleges = avg\_gpa\_by\_col.sort\_values(ascending = False)

top\_5 = sorted\_colleges.head(5)

print("TOP 5 colleges by avg CGPA:\n")

print(top\_5)

Output:



Python Visualization Code:

plt.figure(figsize=(20,12))

top\_5.plot(kind='bar')

plt.xlabel("College Name")

plt.ylabel("Avg CGPA")

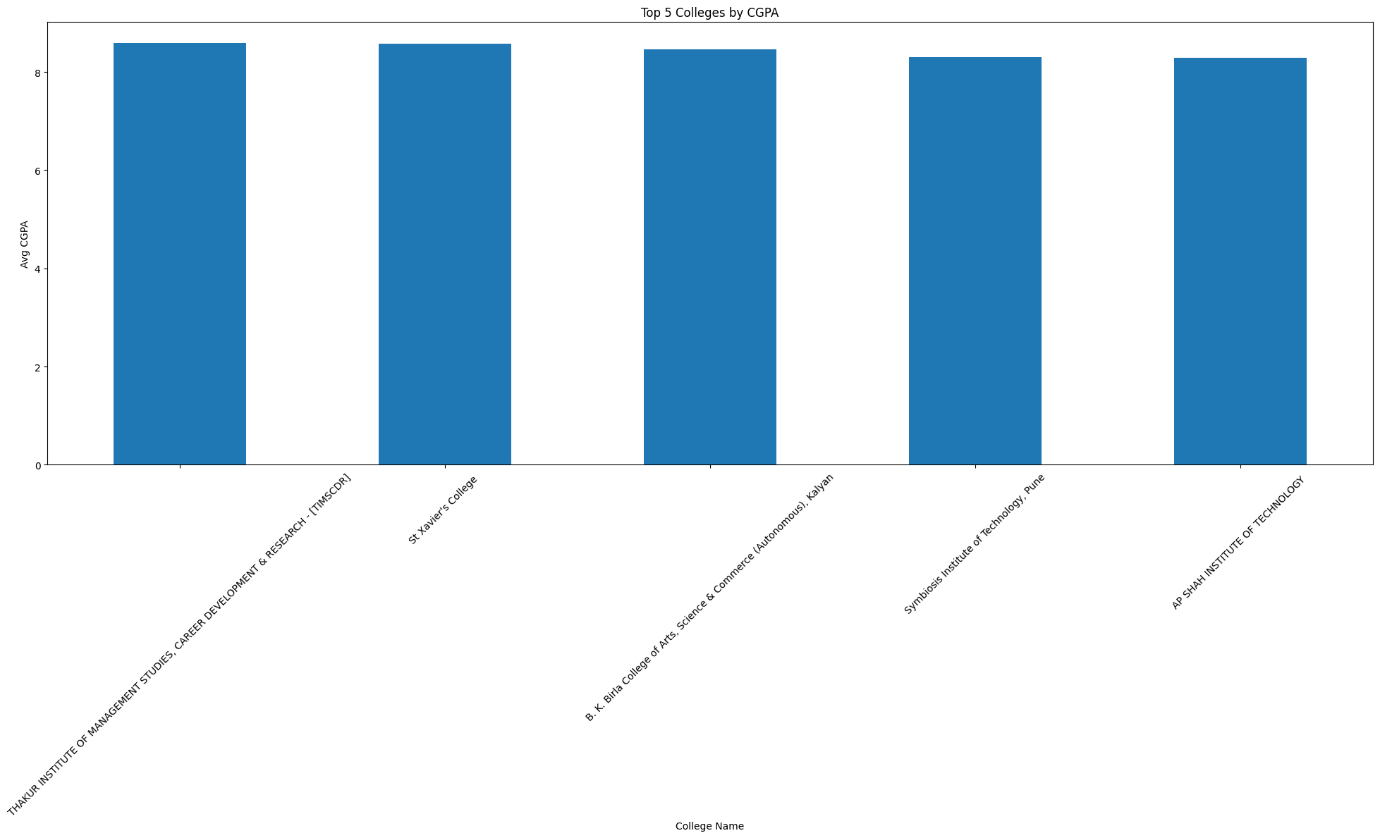
plt.title("Top 5 Colleges by CGPA")

plt.xticks(rotation=45)

plt.tight\_layout()

plt.show()

Visualization:



1. Are the any outlier in quantity(no of cources completed?

Python code:

df['Quantity'].value\_counts()

# '''If the 'quantity' attribute indicates the number of completed courses,

# and all recorded values are 1,then there is no outlier but this suggests

# that all students in the dataset have completed only one course.

# As a result, this attribute may not offer meaningful variation for analysis,

# since there is no diversity in the number of courses completed among the students.'''

Output:



Python Visualization Code:

plt.figure(figsize=(4, 4))

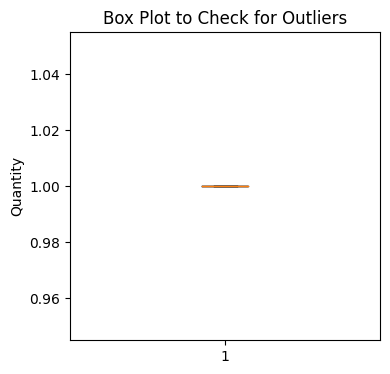
plt.boxplot(df['Quantity'])

plt.title('Box Plot to Check for Outliers')

plt.ylabel('Quantity')

plt.show()

Visualization:



1. Average Cgpa for students from each city?

Python code:

avg\_gpa\_by\_city = df.groupby('City')['CGPA'].mean()

avg\_gpa\_by\_city

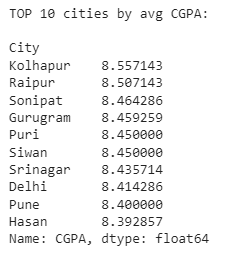
sorted\_city = avg\_gpa\_by\_city.sort\_values(ascending=False)

top\_10 = sorted\_city.head(10)

print("TOP 10 cities by avg CGPA:\n")

print(top\_10)

Output:



Python Visualization Code:

top\_10\_reversed = top\_10[::-1]

plt.figure(figsize=(20, 12))

top\_10\_reversed.plot(kind='barh', color='skyblue')

plt.xlabel("City Names")

plt.ylabel("Avg CGPA")

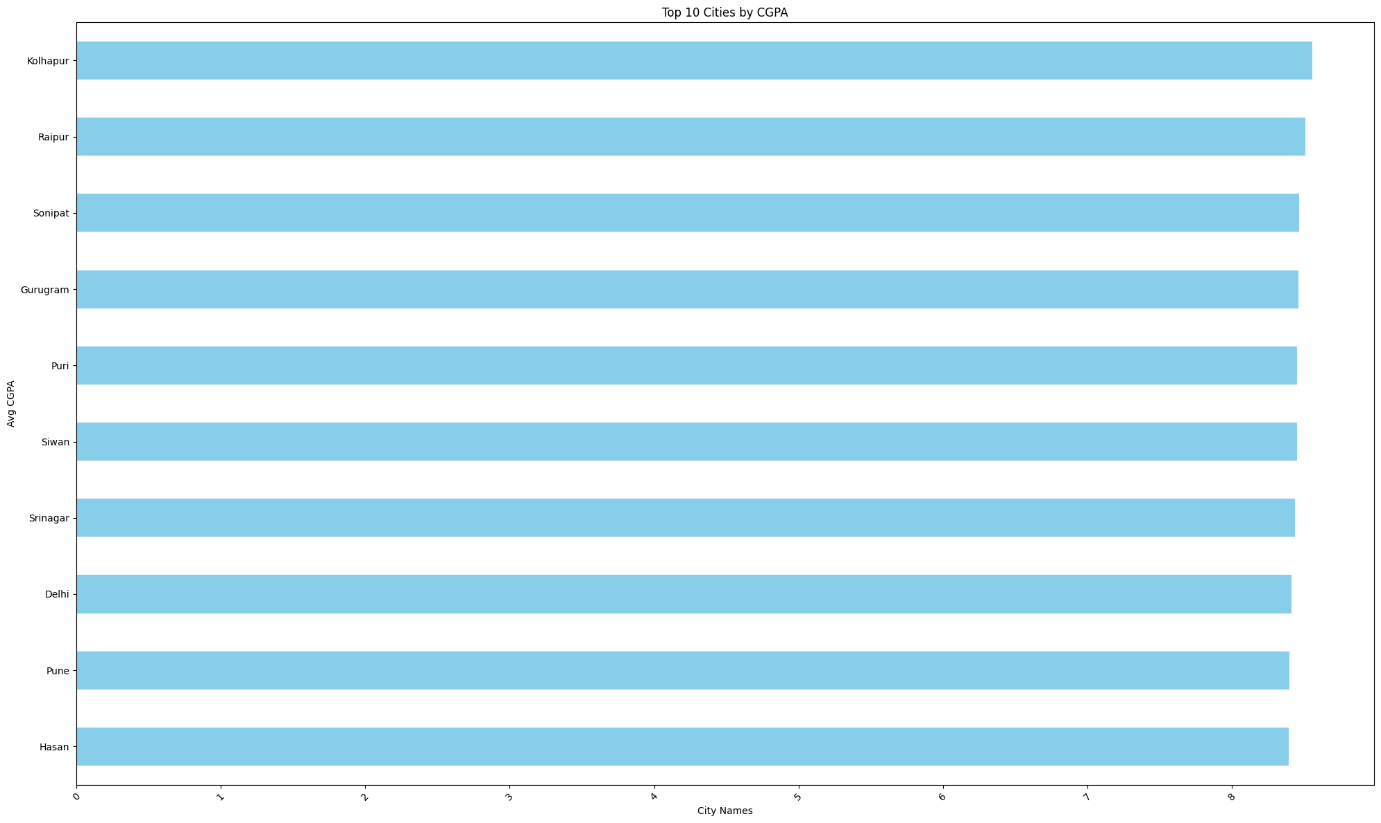
plt.title("Top 10 Cities by CGPA")

plt.xticks(rotation=45)

plt.tight\_layout()

plt.show()

Visualization:



Github Link: https://github.com/shaikharif07/IAC\_Student\_insight/upload/main