

# EDS PRACTICAL NO 5

Atharv Lokhande(239)

Vaibhav Jadhav(225)

1.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
data.head()
df = pd.DataFrame(data)

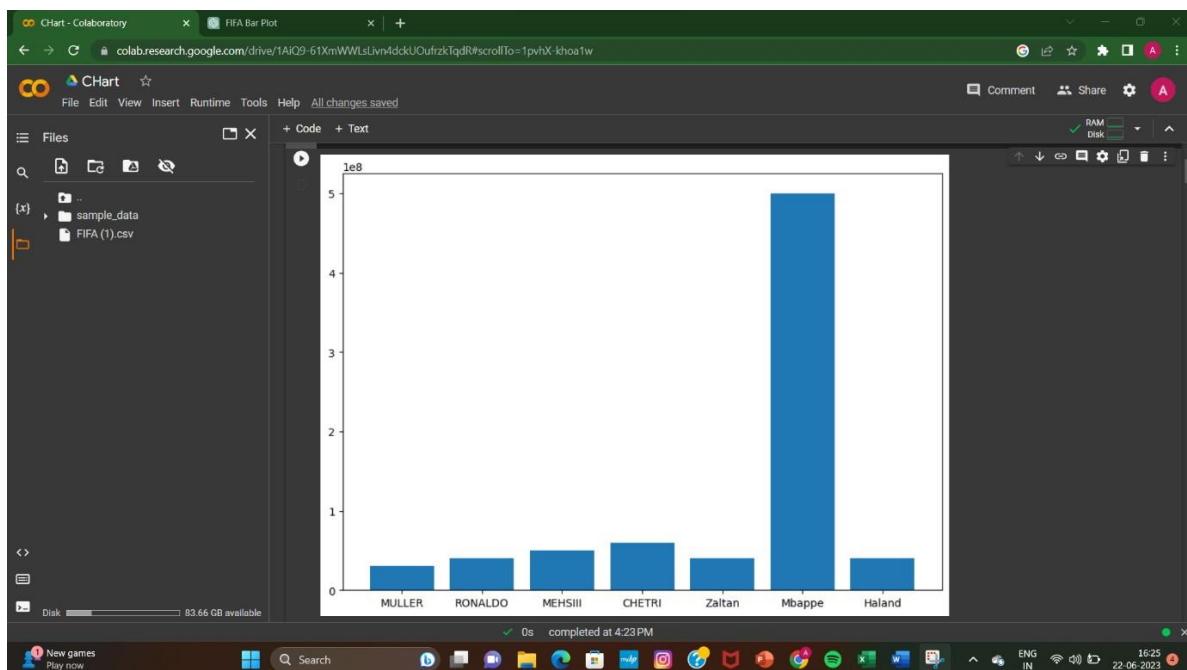
name = df['PN'].head(12)
price = df['SAL'].head(12)

# Figure Size
fig = plt.figure(figsize=(10, 7))

# Horizontal Bar Plot
plt.bar(name[0:10], price[0:10])

# Show Plot
plt.show()
```

OUTPUT-



2.

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

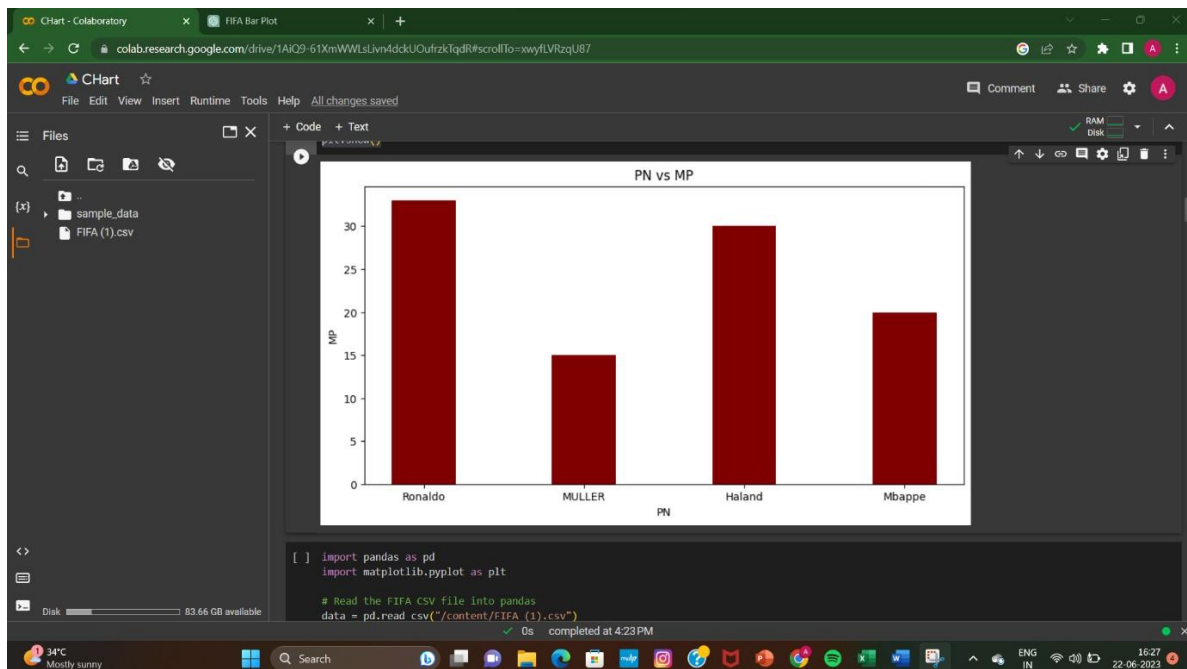
# creating the dataset
data = {'Ronaldo':33, 'MULLER':15, 'Haland':30,
        'Mbappe':20}
courses = list(data.keys())
values = list(data.values())

fig = plt.figure(figsize = (10, 5))

# creating the bar plot
plt.bar(courses, values, color='maroon',
        width = 0.4)

plt.xlabel("PN")
plt.ylabel("MP")
plt.title("PN vs MP")
plt.show()
```

OUTPUT-



3.

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

# Read the FIFA CSV file into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Extract the desired columns
team_name = df['TN'].head(12)
salary = df['SAL'].head(12)

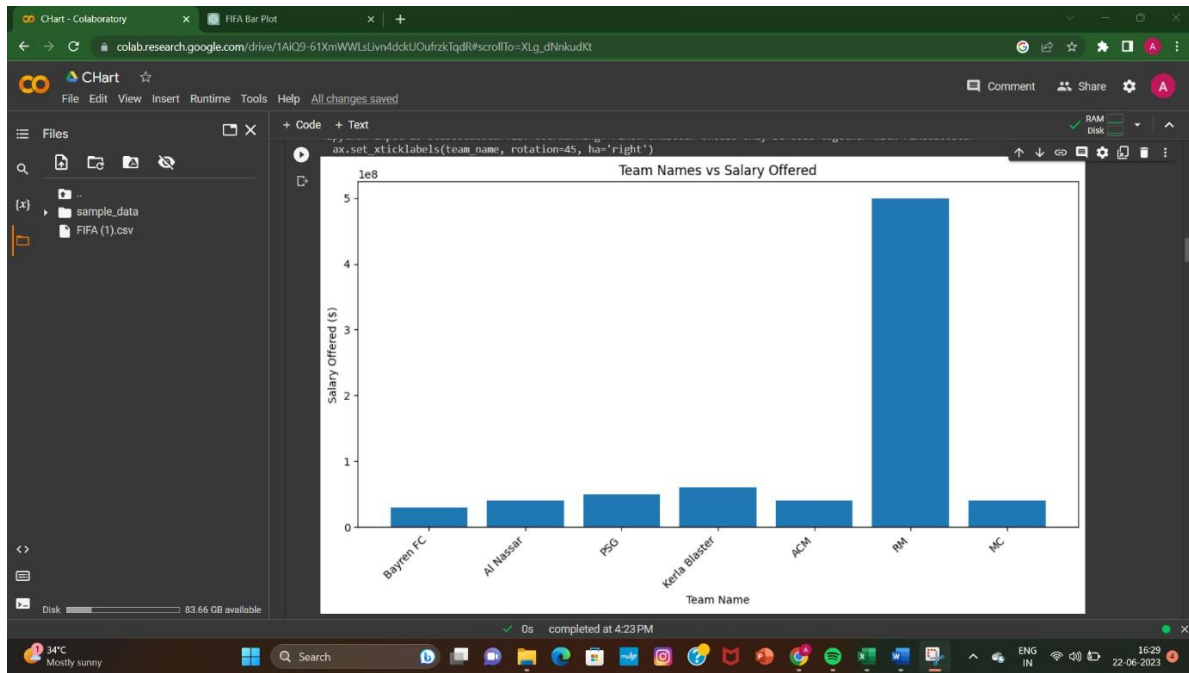
# Figure size
fig, ax = plt.subplots(figsize=(10, 6))

# Bar plot
ax.bar(team_name, salary)

# Customize the plot
ax.set_xlabel('Team Name')
ax.set_ylabel('Salary Offered ($)')
ax.set_title('Team Names vs Salary Offered')
ax.set_xticklabels(team_name, rotation=45, ha='right')
plt.tight_layout()

# Show the plot
plt.show()
```

OUTPUT-



4.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Group the data by team and calculate the total goals scored
team_goals = df.groupby('TN')['GS'].sum()

# Get the team names and goal scores
team_names = team_goals.index.tolist()
goal_scores = team_goals.values.tolist()

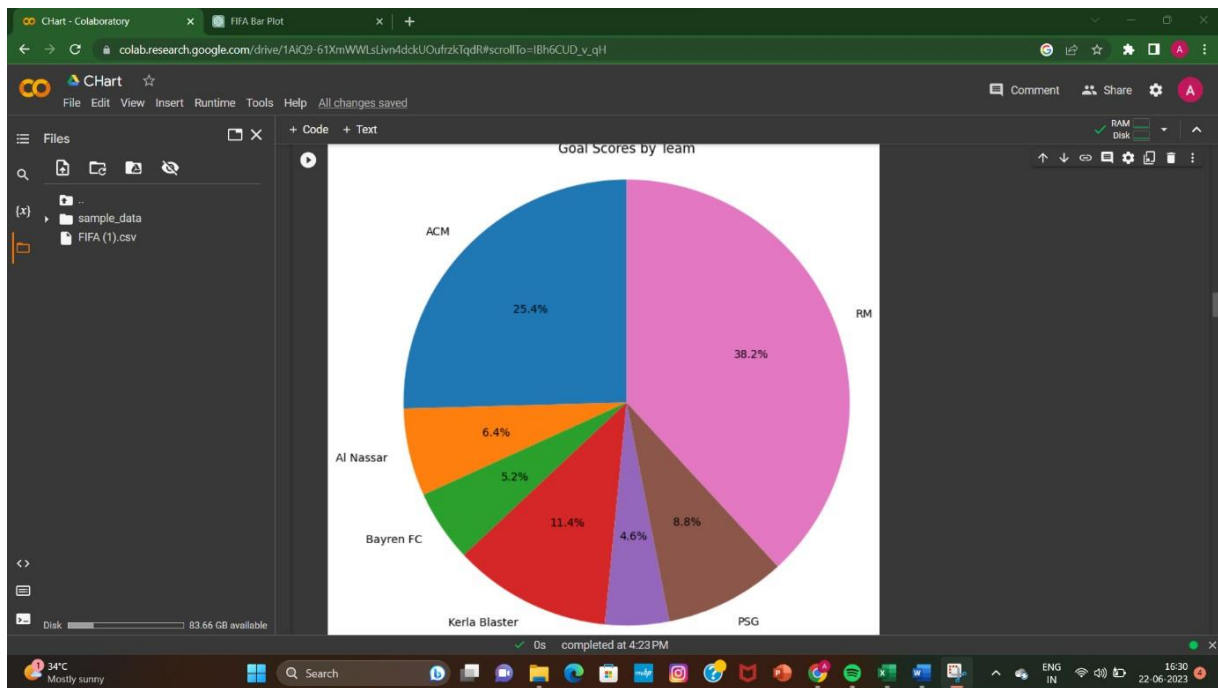
# Create the pie chart
fig, ax = plt.subplots(figsize=(8, 8))
ax.pie(goal_scores, labels=team_names, autopct='%1.1f%%',
startangle=90)

# Add a title
ax.set_title('Goal Scores by Team')

# Equal aspect ratio ensures that pie is drawn as a circle
ax.axis('equal')

# Show the pie chart
plt.show()
```

OUTPUT-



5.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Group the data by League name and calculate the total goal scores
league_goals = df.groupby('LN')['GS'].sum()

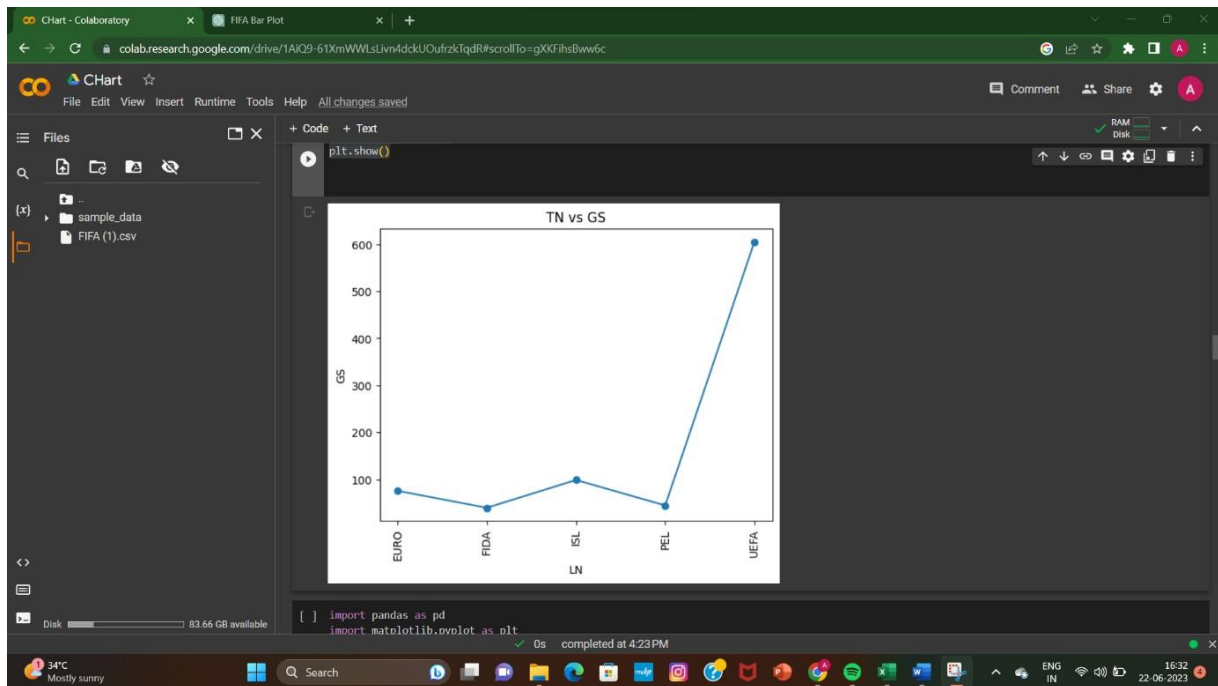
# Create a line chart
plt.plot(league_goals.index, league_goals.values, marker='o')

# Set the chart title and axis labels
plt.title('TN vs GS')
plt.xlabel('LN')
plt.ylabel('GS')

# Rotate the x-axis labels for better visibility
plt.xticks(rotation=90)

# Display the chart
plt.show()
```

## OUTPUT-



6.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Extract the desired columns
age = df['PN']
overall = df['GS']

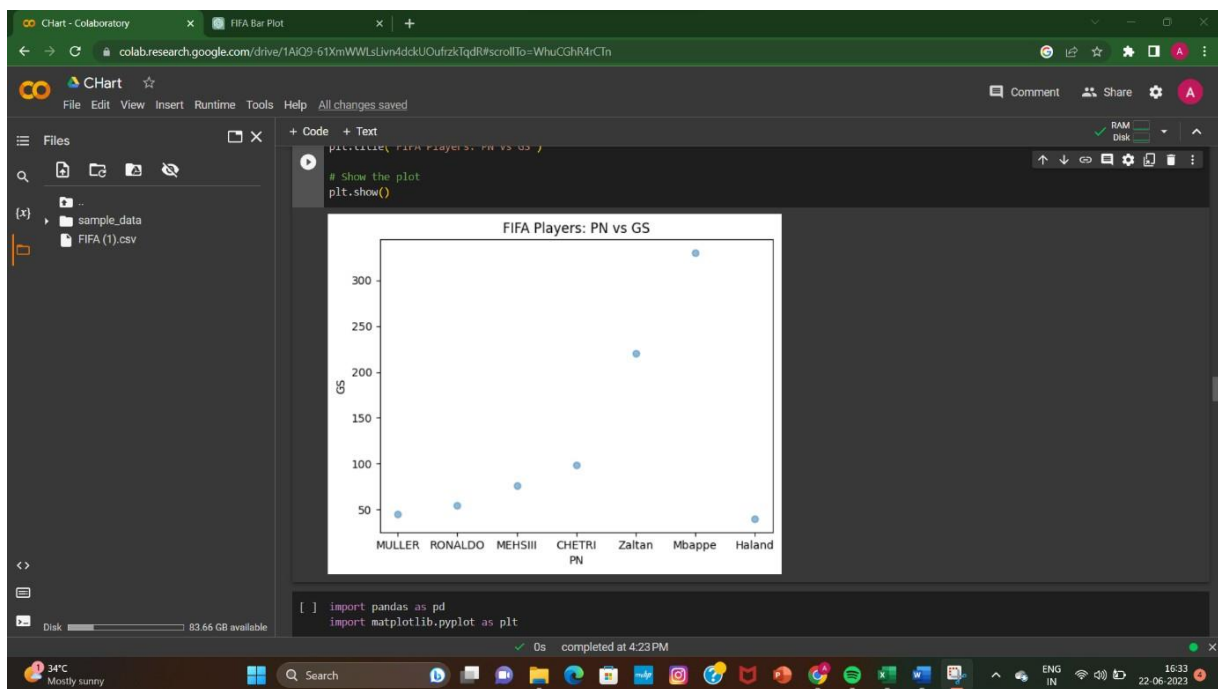
# Create scatter plot
plt.scatter(age, overall, alpha=0.5)

# Set axis labels
plt.xlabel('PN')
plt.ylabel('GS')

# Set plot title
plt.title('FIFA Players: PN vs GS')

# Show the plot
plt.show()
```

## OUTPUT-



7.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Group the data by league and calculate the total matches played
matches_per_league = df.groupby('LN')['MP'].sum()

# Sort the leagues based on the total matches played
sorted_leagues = matches_per_league.sort_values(ascending=False)

# Plot the data
fig, ax = plt.subplots(figsize=(10, 6))
sorted_leagues.plot(kind='bar', ax=ax)

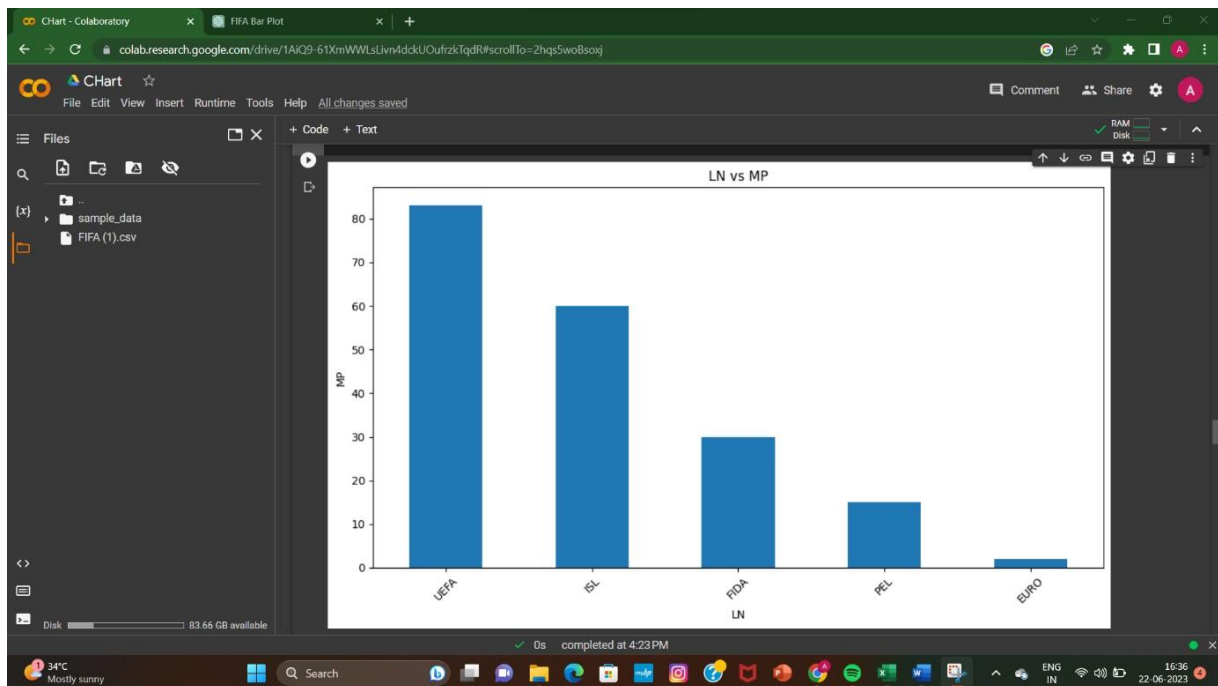
# Set labels and title
ax.set_xlabel('LN')
ax.set_ylabel('MP')
ax.set_title('LN vs MP')

# Rotate x-axis labels for better readability
plt.xticks(rotation=45)
```



```
# Show the plot
plt.tight_layout()
plt.show()
```

## OUTPUT-



8.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Count the number of matches played per team
team_matches = df['TN'].value_counts()

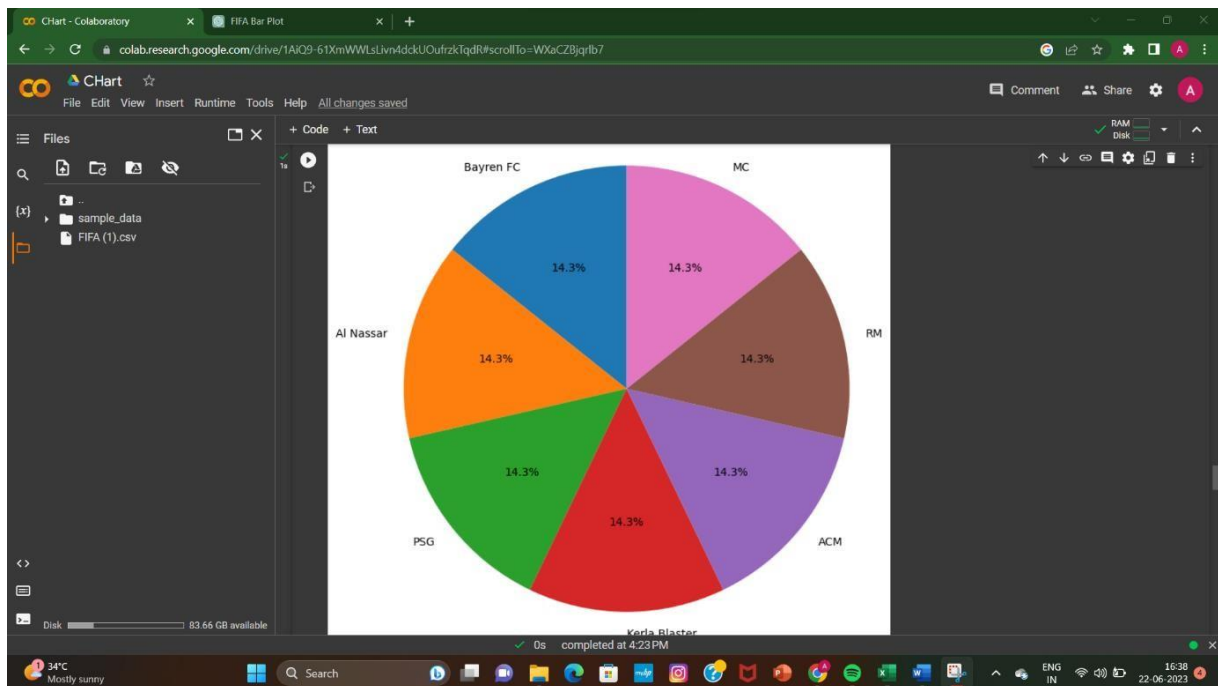
# Extract team names and match counts
team_names = team_matches.index.tolist()
matches_played = team_matches.tolist()

# Create a pie chart
plt.figure(figsize=(8, 8))
plt.pie(matches_played, labels=team_names, autopct='%1.1f%%',
startangle=90)
```

```
plt.title('TN vs MP')
plt.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle

# Show the pie chart
plt.show()
```

## OUTPUT-



9.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Extract the desired columns
team_name = df['TN']
matches_played = df['MP']
salary = df['SAL']

# Sort the data by matches played in descending order
sorted_indices = matches_played.argsort()[::-1] # Get the indices that
sort the matches played in descending order
team name = team name[sorted_indices]
```

```

matches_played = matches_played[sorted_indices]
salary = salary[sorted_indices]

# Get the top 10 teams
top_10_teams = team_name.head(10)
top_10_matches_played = matches_played.head(10)
top_10_salary = salary.head(10)

# Create the bar graph
fig, ax = plt.subplots(figsize=(12, 6))

# Set the x and y values
x = range(len(top_10_teams))
y1 = top_10_matches_played
y2 = top_10_salary

# Plot the number of matches played
ax.bar(x, y1, width=0.4, label='MP')

# Plot the salary offered
ax.bar(x, y2, width=0.4, label='SAL', alpha=0.7)

# Add labels, title, and legend
ax.set_xlabel('TN')
ax.set_ylabel('Count')
ax.set_title('TN vs. MP and SAL( 7 Teams)')
ax.set_xticks(x)
ax.set_xticklabels(top_10_teams, rotation=90)
ax.legend()

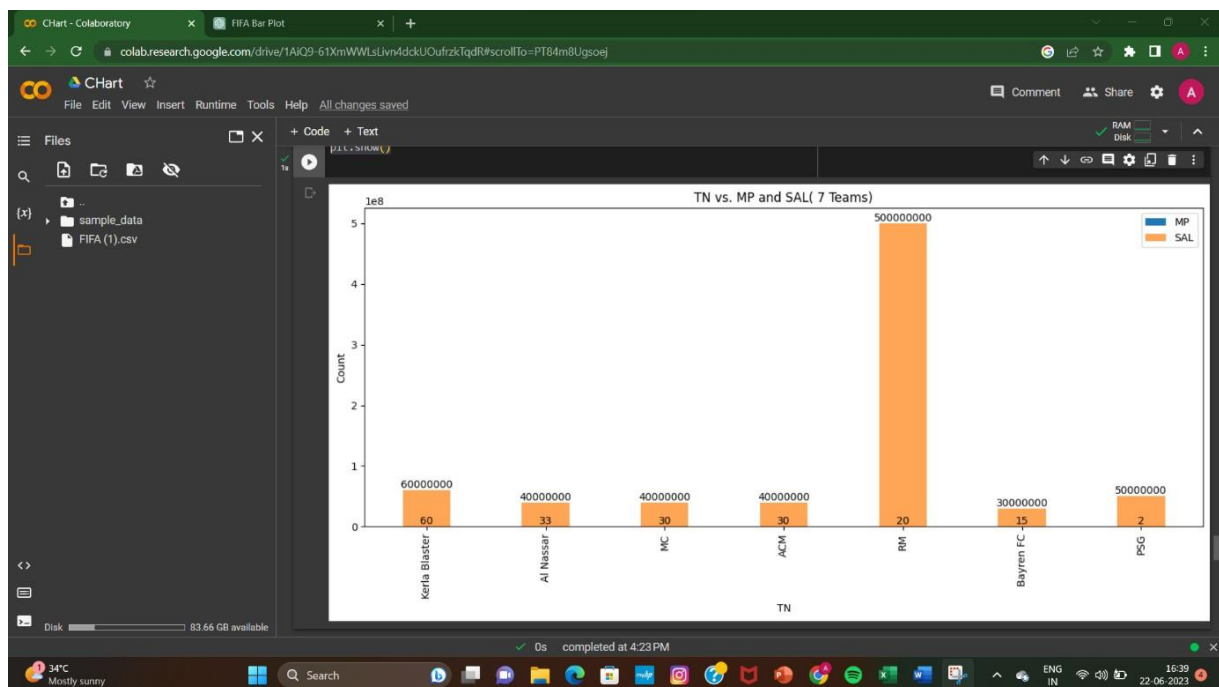
# Add data labels
for i, v1, v2 in zip(x, y1, y2):
    ax.text(i, v1 + 50, str(v1), ha='center', va='bottom')
    ax.text(i, v2 + 1000, str(v2), ha='center', va='bottom')

# Adjust the layout
plt.tight_layout()

# Show the plot
plt.show()

```

## OUTPUT-



10.

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

# Read CSV into pandas
data = pd.read_csv("/content/FIFA (1).csv")
df = pd.DataFrame(data)

# Extract the desired columns
name = df['PN'].head(12)
matches_played = df['MP'].head(12)
goals_scored = df['GS'].head(12)

# Set the figure size
fig, ax = plt.subplots(figsize=(12, 8))

# Plot the bars for matches played
ax.bar(name, matches_played, label='MP', color='blue', alpha=0.6)

# Plot the bars for goals scored
ax.bar(name, goals_scored, label='GS', color='orange', alpha=0.6)

# Set the title and labels
ax.set_title('FIFA Players: MP vs. GS', fontsize=16)
ax.set_xlabel('PN', fontsize=12)
```

```

ax.set_ylabel('Count', fontsize=12)

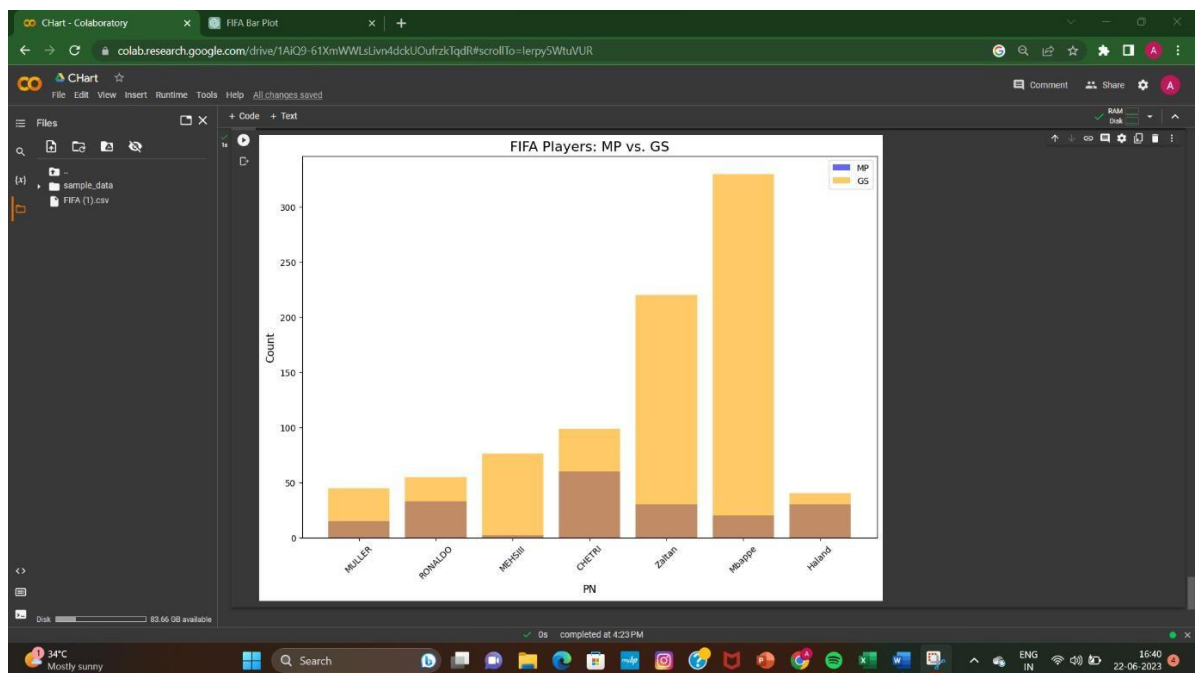
# Add a legend
ax.legend()

# Rotate x-axis labels for better visibility
plt.xticks(rotation=45)

# Show the plot
plt.show()

```

OUTPUT-



```
11.
import pandas as pd
import matplotlib.pyplot as plt

# Read CSV into pandas
data = pd.read_csv("/content/Fifa.csv")

# Extract the columns from the dataset
TN = data['TN']
GS = data['GS']
MP = data['MP']

# Group the data by team name
grouped_data = data.groupby('TN').sum()

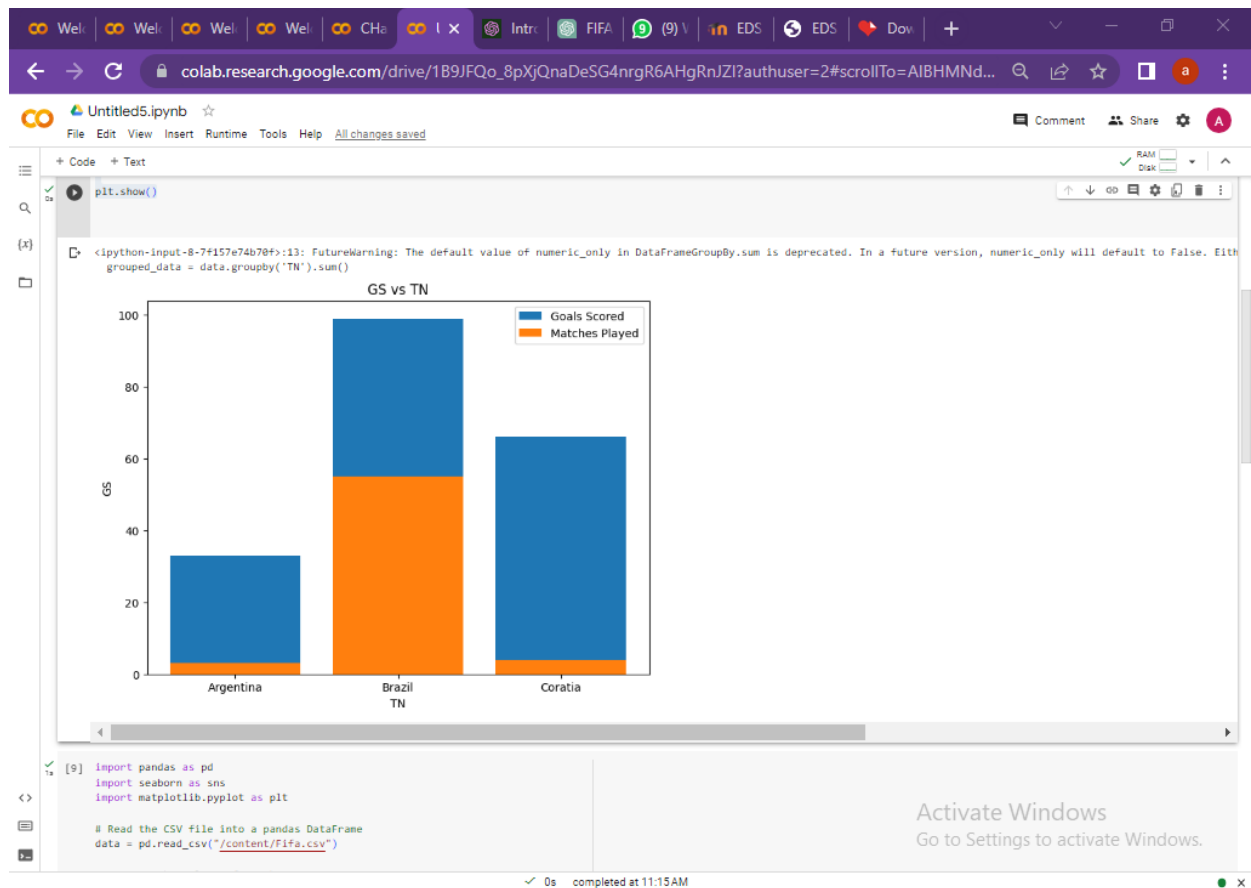
# Plotting
fig, ax = plt.subplots(figsize=(8, 6))

# Bar plot for goals scored
ax.bar(grouped_data.index, grouped_data['GS'], label='Goals Scored')

# Bar plot for matches played
ax.bar(grouped_data.index, grouped_data['MP'], label='Matches Played')

ax.set_xlabel('TN')
ax.set_ylabel('GS')
ax.set_title('GS vs TN')
ax.legend()

plt.show()
```



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

# Read the CSV file into a pandas DataFrame
data = pd.read_csv("/content/Fifa.csv")

# Extract the columns from the DataFrame
TN = data['TN']
GS = data['GS'].astype(int)
MP = data['MP'].astype(int)

# Group the data by team name and calculate the sum of goals scored and
matches played
grouped_data = data.groupby('TN').sum().reset_index()

# Create a panel graph
fig, axes = plt.subplots(2, 1, figsize=(8, 10))

# Bar plot for goals scored
sns.barplot(x='TN', y='GS', data=grouped_data, ax=axes[0])
axes[0].set_xlabel('TN')
axes[0].set_ylabel('GS')
axes[0].set_title('GS vs TN')

# Bar plot for matches played
sns.barplot(x='TN', y='MP', data=grouped_data, ax=axes[1])
axes[1].set_xlabel('TN')
axes[1].set_ylabel('MP')
axes[1].set_title('MP vs TN')

# Adjust the spacing between subplots
plt.tight_layout()

# Show the plot
plt.show()
```



