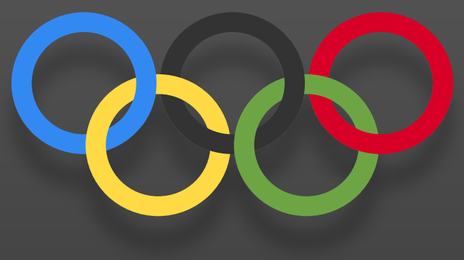
OLYMPICS Data Analysis
Using Python & PowerBI



Introduction

The Olympic Games represent one of the world's largest and most prestigious sporting events, uniting athletes from across the globe. With vast amounts of data generated from performances, scores, and historical trends, data analysis plays a crucial role in understanding and optimizing athletic outcomes. By applying statistical methods and machine learning, analysts can uncover patterns, enhance training, and even predict future medal winners.

Objective

- Analyze the dataset to understand trends in medal distribution.
- Identify the top-performing countries and athletes.
- Study the gender distribution of events and medals.
- Visualize the data using Python.

Dataset overview

Dataset Name: Summer-Olympic-medals-1976-to-2008

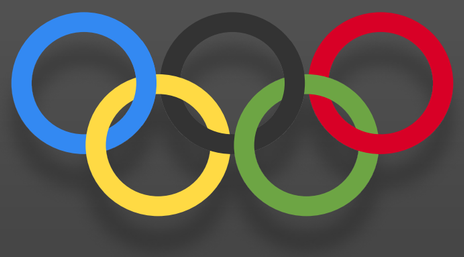
Source: Unified Mentor (Internship Project)

Total Entries: 15433

Total Columns: 11 columns: City, Year, Sport, Discipline, Event, Athlete, Gender, Country_Code, Country, Event_gender, Medal

Purpose: To perform exploratory data analysis & uncovering trends and insights related to athletes, countries, and sports over the years.

"This dataset, provided by Unified Mentor as part of the internship project, contains 15433 records of Olympics since 1976 till 2008 . It includes key details like City, Year, Sport, Discipline, Event, Athlete, Gender, Country_Code, Country, Event_gender, Medal. The analysis focuses on understanding historical data, medals won by countries, athletes, gender distribution etc using Python and Power BI."



Step 1: Data Preparation

- Importing libraries
- loading datase
- checking datasets and creating brief summary of dataset

- **Importing libraries**

```
[ ]  # Importing necessary libraries
     import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
```

- **loading dataset**

```
[ ]  # Loading the dataset (assuming CSV format)
     df = pd.read_csv('/content/Summer-Olympic-medals-1976-to-2008.csv', encoding='latin-1')
```

- **Checking datasets:**

Checking the first few rows of data set

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

```
      City    Year     Sport Discipline           Event  \
0  Montreal  1976.0  Aquatics     Diving  3m springboard
1  Montreal  1976.0  Aquatics     Diving  3m springboard
2  Montreal  1976.0  Aquatics     Diving  3m springboard
3  Montreal  1976.0  Aquatics     Diving  3m springboard
4  Montreal  1976.0  Aquatics     Diving    10m platform

                  Athlete Gender Country_Code       Country Event_gender  \
0          KÖHLER, Christa  Women          GDR   East Germany            W
1       KOSENKOV, Aleksandr    Men          URS   Soviet Union            M
2      BOGGS, Philip George    Men          USA  United States            M
3  CAGNOTTO, Giorgio Franco    Men          ITA          Italy            M
4    WILSON, Deborah Keplar  Women          USA  United States            W

    Medal
0  Silver
1  Bronze
2    Gold
3  Silver
4  Bronze
```

- **Summary of data set:**

total entries: 15433
total column: 11
data types: 1 float, 10 object

```
# Summary(1) of the dataset
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15433 entries, 0 to 15432
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   City          15316 non-null  object 
 1   Year          15316 non-null  float64
 2   Sport         15316 non-null  object 
 3   Discipline    15316 non-null  object 
 4   Event         15316 non-null  object 
 5   Athlete       15316 non-null  object 
 6   Gender        15316 non-null  object 
 7   Country_Code  15316 non-null  object 
 8   Country       15316 non-null  object 
 9   Event_gender  15316 non-null  object 
 10  Medal         15316 non-null  object 
dtypes: float64(1), object(10)
memory usage: 1.3+ MB
None
```

Step 2: Data Cleaning

- Checking for missing values and remove or impute them if necessary.
- Drop rows with missing values if any
- After cleaning, check the dataset again

- **Checking for missing values**

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

```
City              117
Year              117
Sport             117
Discipline        117
Event             117
Athlete           117
Gender            117
Country_Code      117
Country           117
Event_gender      117
Medal             117
dtype: int64
```

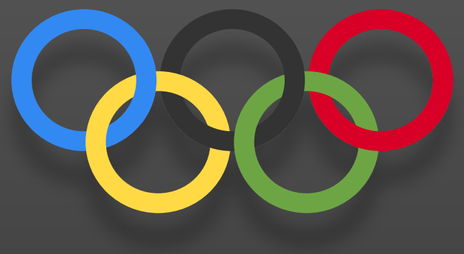
- 117 empty rows were found

- Drop rows with missing values if any
- Checking the dataset again

```
# Drop rows with missing values if any
df_cleaned = df.dropna()
```

```
# After cleaning, check the dataset again
print(df_cleaned.info())
```

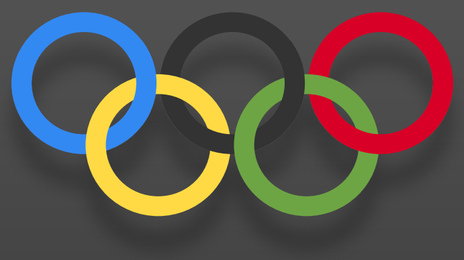
```
<class 'pandas.core.frame.DataFrame'>
Index: 15316 entries, 0 to 15432
Data columns (total 11 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   City          15316 non-null  object 
 1   Year          15316 non-null  float64
 2   Sport         15316 non-null  object 
 3   Discipline    15316 non-null  object 
 4   Event         15316 non-null  object 
 5   Athlete       15316 non-null  object 
 6   Gender        15316 non-null  object 
 7   Country_Code  15316 non-null  object 
 8   Country       15316 non-null  object 
 9   Event_gender  15316 non-null  object 
 10  Medal         15316 non-null  object 
dtypes: float64(1), object(10)
memory usage: 1.4+ MB
None
```



Data Visualization with Python

- "After verifying data quality, we proceeded with exploratory data patterns, trends, and insights from the dataset. Using Matplotlib and Seaborn, we created multiple charts to analyze “Summer-Olympic-medals-1976-to-2008” "Exploratory Data Analysis (EDA) in Python“ "Unveiling Insights with Data Visualization“ analysis (EDA) using Python. Visualization helps in uncovering content distribution, genre popularity, and trends over time."

- "Exploratory Data Analysis (EDA) in Python“
- "Unveiling Insights with Data Visualization



Step 3: Exploratory Data Analysis (EDA)

The step include summarizing statistics of dataset and Visualizing key insights

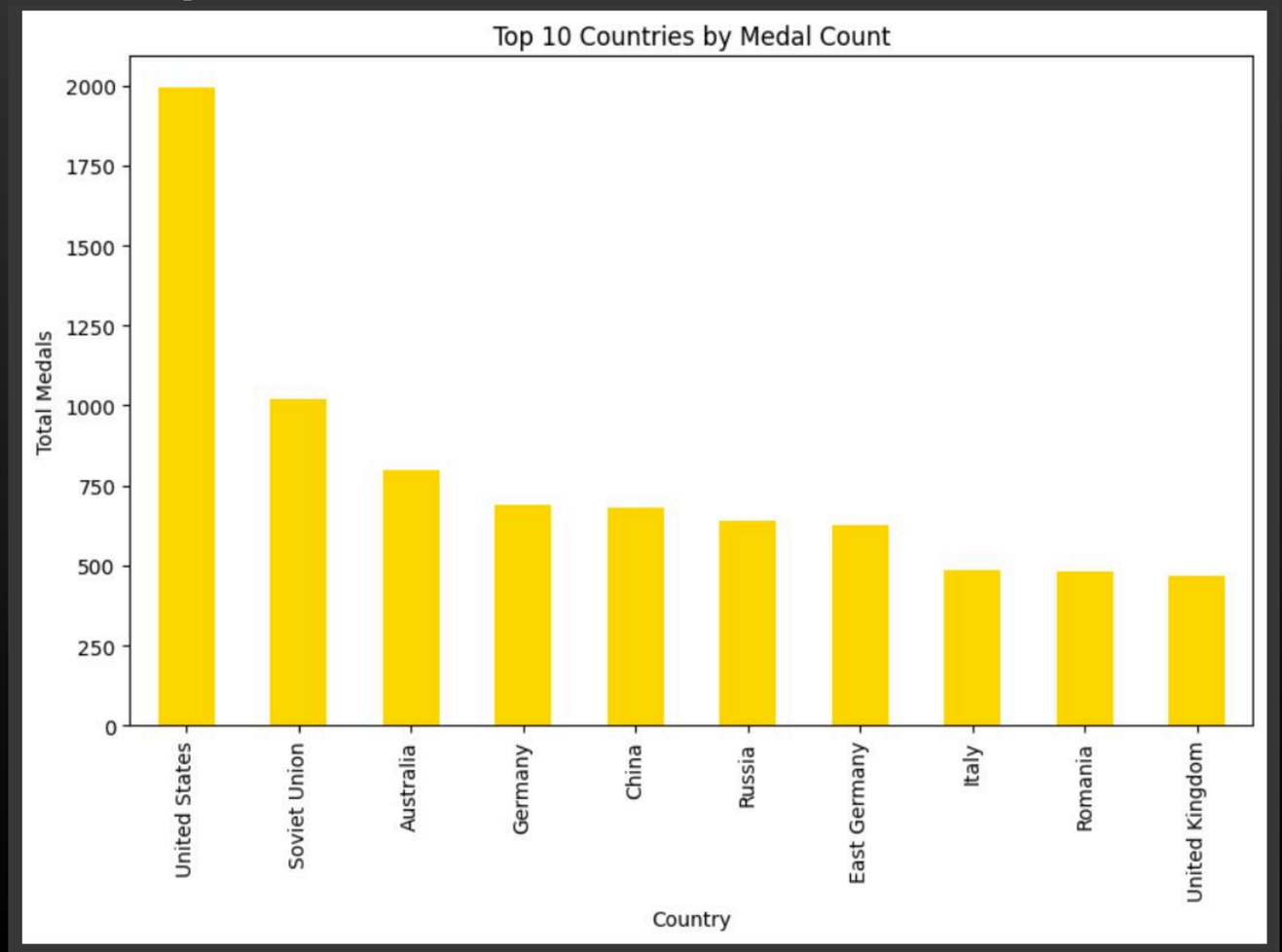
- Plotting the top 10 countries by medals
- Medals Won Over the Years
- Gender Distribution in Events
- Top Athletes with Most Medals

- **Plotting the top 10 countries by medals**

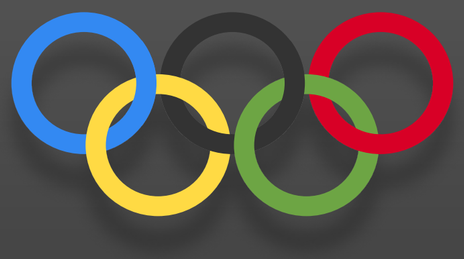
- **Code**

```
# Plotting the top 10 countries by medals
plt.figure(figsize=(10, 6))
medals_by_country.head(10).plot(kind='bar', color='gold')
plt.title("Top 10 Countries by Medal Count")
plt.xlabel("Country")
plt.ylabel("Total Medals")
plt.show()
```

- **Output**



- Bar Graph on right side shows top 10 countries by medal won, X axis shows countries and Y axis shows medals count



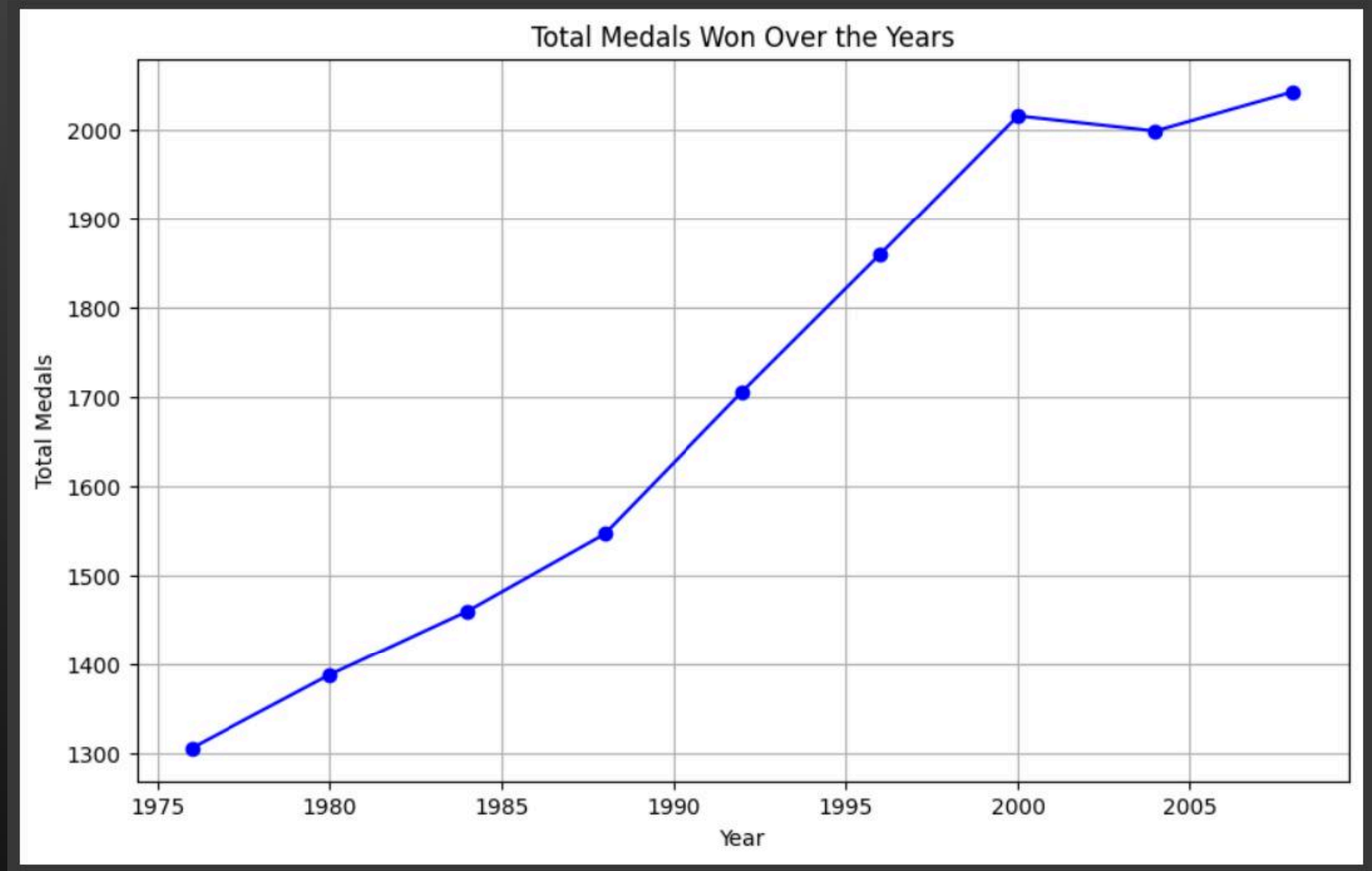
- **Medals Won Over the Years**

- **Code**

```
[ ]  # 3.2 Medals Won Over the Years

     # Grouping by Year and counting the medals won

     medals_over_years = df_cleaned.groupby('Year')['Medal'].count()

     # Plotting the trend of medals won over the years

     plt.figure(figsize=(10, 6))
     plt.plot(medals_over_years.index, medals_over_years.values,

     marker='o', linestyle='-', color='b')
     plt.title("Total Medals Won Over the Years")
     plt.xlabel("Year")
     plt.ylabel("Total Medals")
     plt.grid(True)
     plt.show()
```

- **Medals Won Over the Years**

- **Output**



- Line Chart on right side shows total medals won over years, X axis shows year and Y axis shows medals

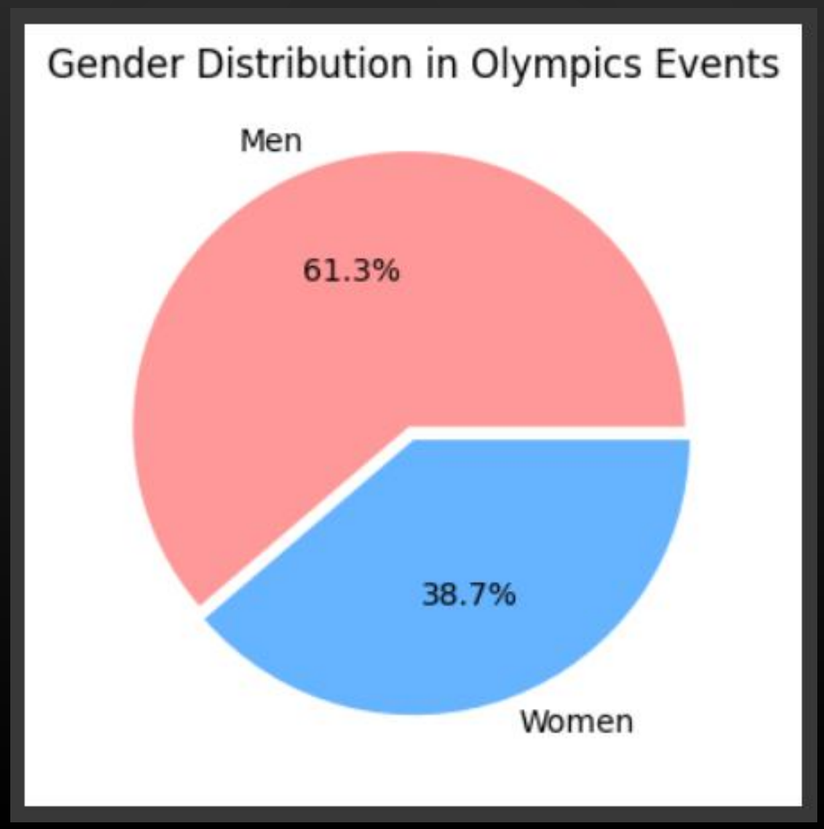
- **Gender Distribution in Events**

- **Code**

```
# 3.3 Gender Distribution in Events

gender_distribution = df_cleaned['Gender'].value_counts()

# Plotting gender distribution

plt.figure(figsize=(6, 4))
gender_distribution.plot(kind='pie', autopct='%1.1f%%',
colors=['#ff9999','#66b3ff'], explode=[0.05, 0])
plt.title("Gender Distribution in Olympics Events")
plt.ylabel('')
plt.show()
```

- **Output**

- Pie Chart on right side shows Percentage of men and woman participated in Olympics till 2008.



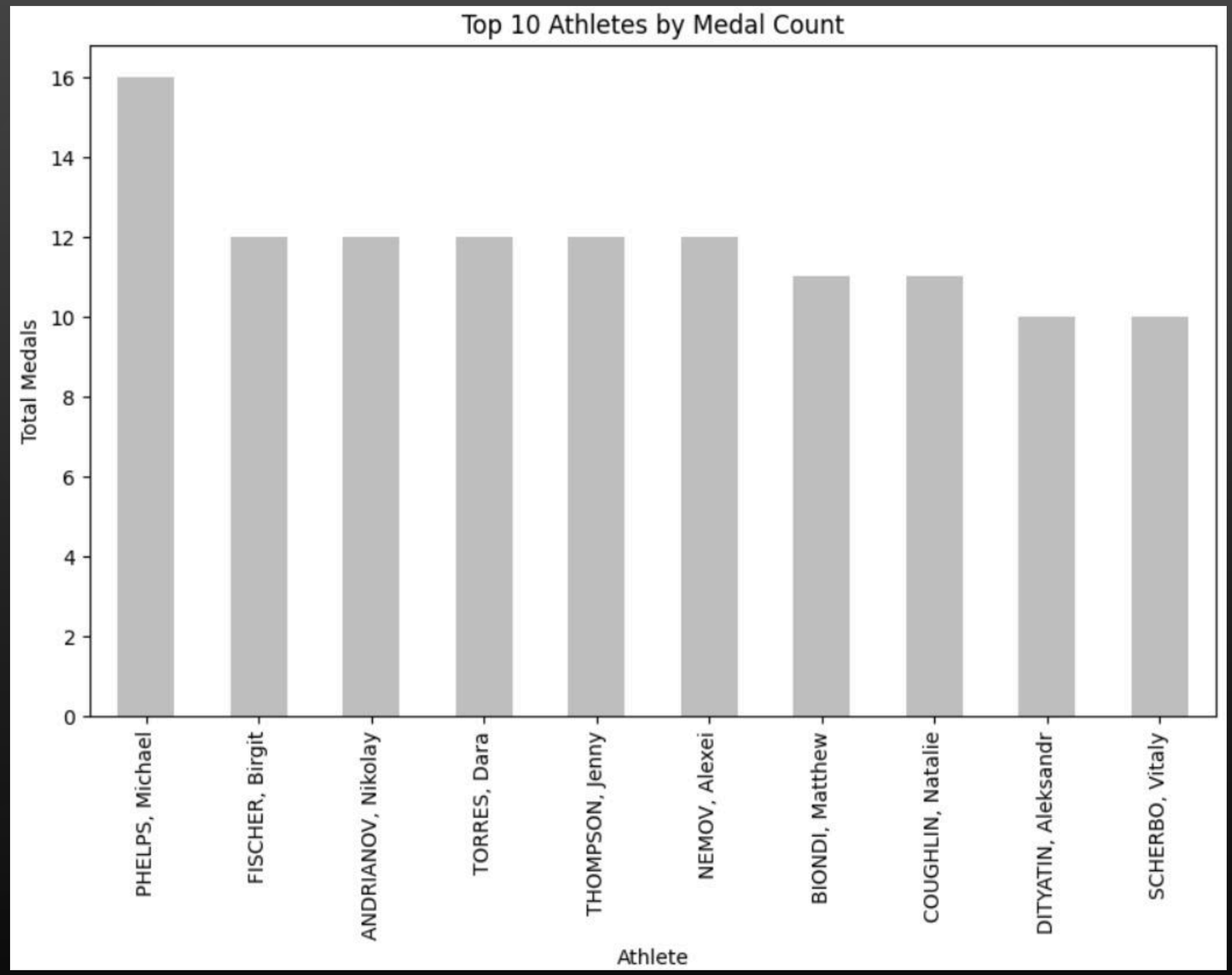
- **Top Athletes with Most Medals**
  - Group by Athlete and count the number of medals
  - **Code**

```
[ ]  # 3.4 Top Athletes with Most Medals

     # Group by Athlete and count the number of medals

     athlete_medal_count = df_cleaned.groupby('Athlete')['Medal'].count().sort_values(ascending=False)
```

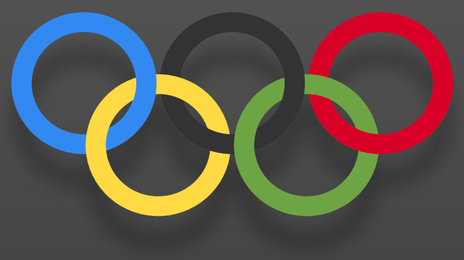
  - Plotting the top 10 athletes with most medals

```
[ ]  # Plotting the top 10 athletes with most medals

     plt.figure(figsize=(10, 6))
     athlete_medal_count.head(10).plot(kind='bar', color='silver')
     plt.title("Top 10 Athletes by Medal Count")
     plt.xlabel("Athlete")
     plt.ylabel("Total Medals")
     plt.show()
```

- **Top Athletes with Most Medals**
  - **Output**



- Bar Graph on right side shows Top 10 athletes by Medal count.



Power BI Dashboard

- After conducting initial data checks in Python, we used Power BI for advanced data transformation and visualization. Power BI enabled us to:

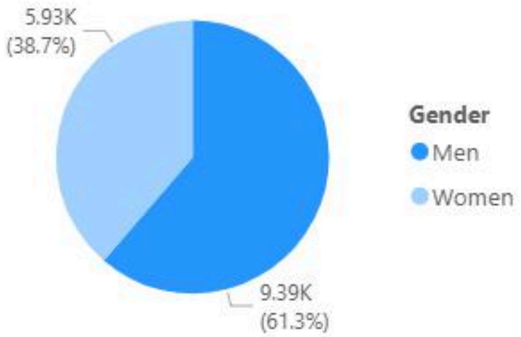
  - Clean and Transform Data – Resolved formatting issues such as removing empty shells
  - Create Interactive Dashboards – To explore Olympic data trends dynamically.
  - Visualize Key Insights – Leveraged bar charts, pie charts, and line charts for clearer understanding.

- Using Power BI, we converted raw Olympic data into an interactive, visually engaging dashboard that reveals deep insights into medal distributions, country-wise performance, and participation trends across different years.

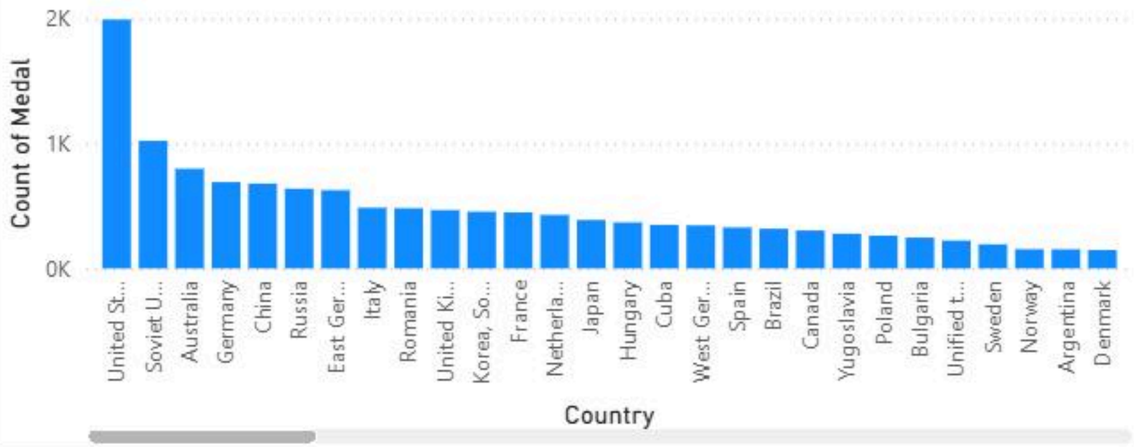
Power BI Dashboard

Summer-Olympic-Data Analysis-1976-to-2008

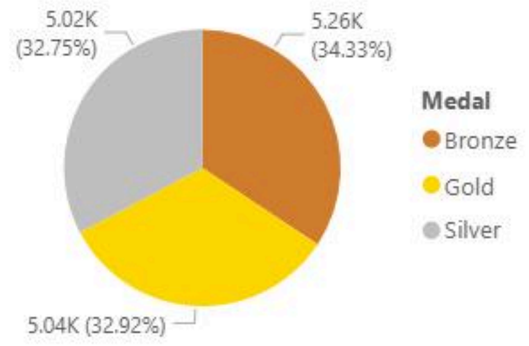
Count of Gender by Gender



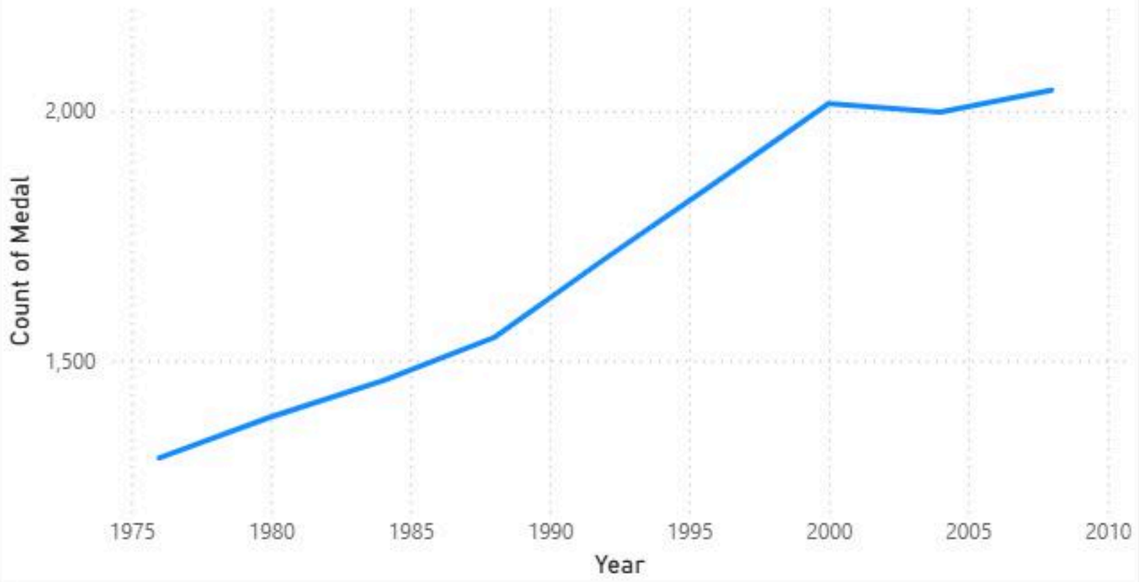
Count of Medal by Country



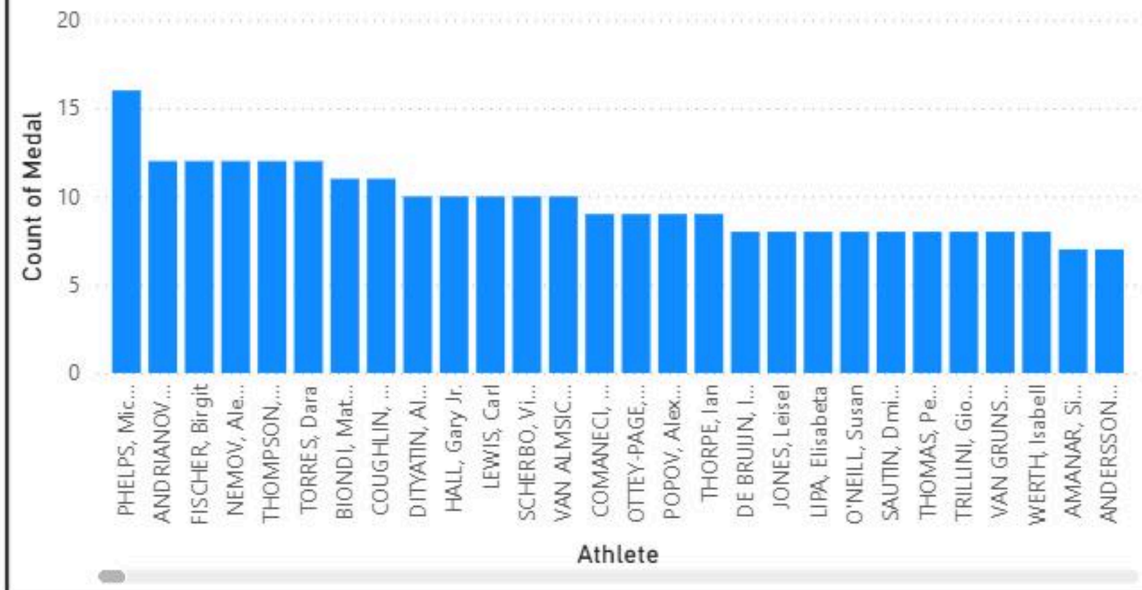
Count of Medals

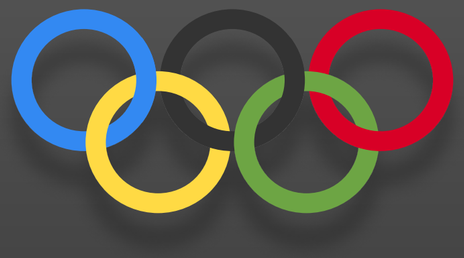


Count of Medal by Year



Count of Medal by Athlete





Conclusion & Key Takeaways

- Data Cleaning: 117 empty entries were found
- Since 1976-2008 United State of America won most medals followed by Russia, Australia, Germany, China and follows.
- Since 1976-2008 total of 5930 (38.7%) Women participated where as 9390 (61.3%) Men participated in Olympics.
- Since 1976-2008 highest medals won is by Phelps Michael from USA, 16 total medals, 14 gold, 2 bronze medals.
- In year 1976 total of 1305 medals were awarded and i year 2008 total of 2042 medals were awarded.

Thank You