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


sns.set(style="whitegrid")
```

```
from google.colab import files
files.upload(r"C:\Users\DEVASISH\Downloads\kaggle.json")
```

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.

```
Saving kaggle.json to C:\Users\DEVASISH\Downloads\kaggle.json/kaggle.json
{'C:\\Users\\DEVASISH\\Downloads\\kaggle.json/kaggle.json':
```

```
import kagglehub

# Download latest version
path = kagglehub.dataset_download("piterfm/olympic-games-medals-19862018")

print("Path to dataset files:", path)
```

```
Downloading from https://www.kaggle.com/api/v1/datasets/download/piterfm/olympic-games-medals-19862018?dataset_version_number=9...
100%|██████████| 13.9M/13.9M [00:00<00:00, 99.9MB/s]Extracting files...

Path to dataset files: /root/.cache/kagglehub/datasets/piterfm/olympic-games-medals-19862018/versions/9
```

```
import os

dataset_path = "/root/.cache/kagglehub/datasets/piterfm/olympic-games-medals-19862018/versions/9"
os.listdir(dataset_path)
```

```
['olympic_results.pkl',
 'olympic_athletes.csv',
 'olympic_hosts.csv',
 'olympic_medals.csv',
 'olympic_results.csv']
```

```
import pandas as pd

file_path = dataset_path + "/olympic_medals.csv"
df = pd.read_csv(file_path)
df.head()
```

	discipline_title	slug_game	event_title	event_gender	medal_type	participant_type	participant_title	athlet
0	Curling	beijing-2022	Mixed Doubles	Mixed	GOLD	GameTeam	Italy	https://olympics.com/en/athletes/st
1	Curling	beijing-2022	Mixed Doubles	Mixed	GOLD	GameTeam	Italy	https://olympics.com/en/athletes/ m
2	Curling	beijing-2022	Mixed Doubles	Mixed	SILVER	GameTeam	Norway	https://olympics.com/en/athletes/ sl
3	Curling	beijing-2022	Mixed Doubles	Mixed	SILVER	GameTeam	Norway	https://olympics.com/en/athletes/m ne
4	Curling	beijing-2022	Mixed Doubles	Mixed	BRONZE	GameTeam	Sweden	https://olympics.com/en/athletes/a

```
df.info()
df.describe()
df['medal_type'].value_counts()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21697 entries, 0 to 21696
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype 
---  ------                --------------  ----- 
 0   discipline_title      21697 non-null  object
 1   slug_game             21697 non-null  object
 2   event_title           21697 non-null  object
 3   event_gender          21697 non-null  object
 4   medal_type            21697 non-null  object
 5   participant_type      21697 non-null  object
 6   participant_title     6584 non-null   object
 7   athlete_url           17027 non-null  object
 8   athlete_full_name     18073 non-null  object
 9   country_name          21697 non-null  object
 10  country_code          20195 non-null  object
 11  country_3_letter_code  21697 non-null  object
dtypes: object(12)
memory usage: 2.0+ MB
```

	count
medal_type	
BRONZE	7529
GOLD	7109
SILVER	7059

dtype: int64

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

	0
discipline_title	0
slug_game	0
event_title	0
event_gender	0
medal_type	0
participant_type	0
participant_title	15113
athlete_url	4670
athlete_full_name	3624
country_name	0
country_code	1502
country_3_letter_code	0

dtype: int64

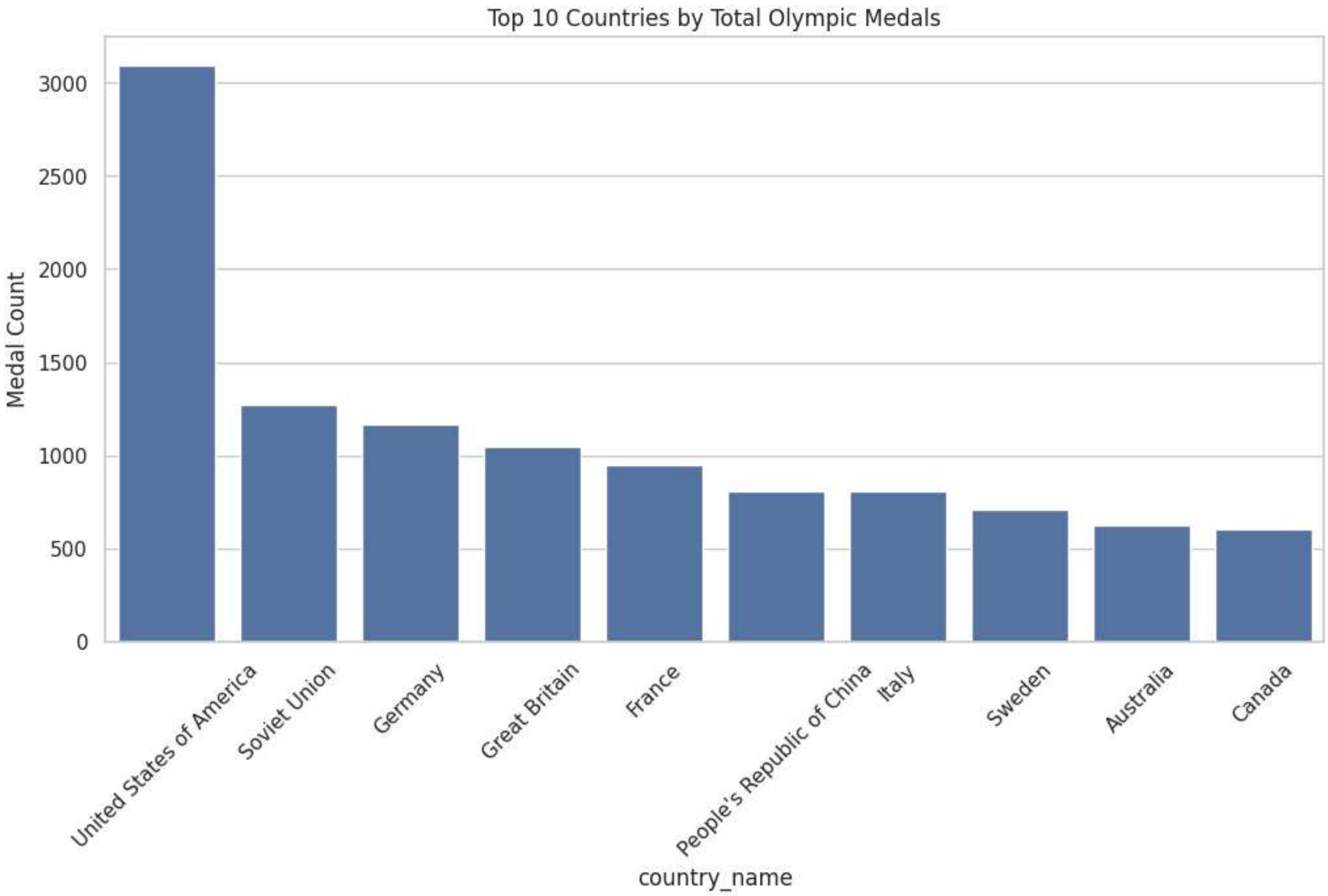
```
df_medals = df[df['medal_type'].notna()]

medal_count = df_medals.groupby(['slug_game', 'country_name', 'medal_type']).size().unstack(fill_value=0)
display(medal_count.head())
```

	medal_type	BRONZE	GOLD	SILVER
slug_game	**country_name**			
albertville-1992	**Austria**	8	6	7
	Canada	3	2	3
	Czechoslovakia	3	0	0
	Democratic People's Republic of Korea	1	0	0
	Finland	3	3	1

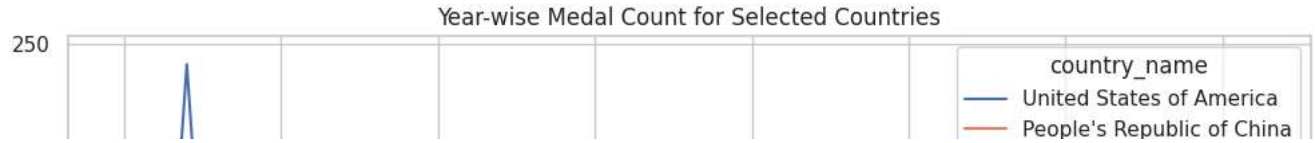
```
total_medals = df_medals.groupby('country_name')['medal_type'].count().sort_values(ascending=False)
plt.figure(figsize=(12,6))
```

```
sns.barplot(x=total_medals.head(10).index, y=total_medals.head(10).values)
plt.xticks(rotation=45)
plt.title("Top 10 Countries by Total Olympic Medals")
plt.ylabel("Medal Count")
plt.show()
```



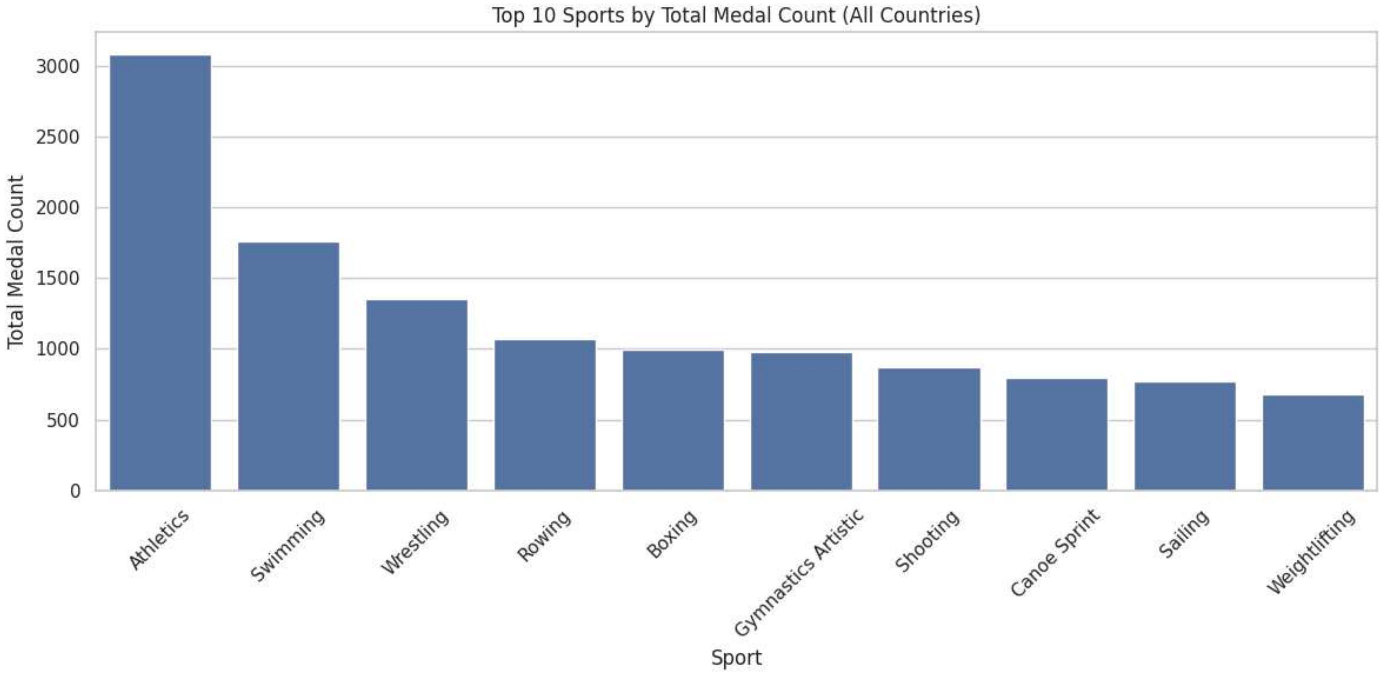
```
df_medals['Year'] = df_medals['slug_game'].str.extract(r'(\d{4})')
yearly_medals = df_medals.groupby(['Year', 'country_name']).size().unstack(fill_value=0)
```

```
yearly_medals[['United States of America', 'People\'s Republic of China', 'Germany', 'Great Britain']].plot(figsize=(12,6))
plt.title("Year-wise Medal Count for Selected Countries")
plt.ylabel("Medal Count")
plt.show()
```



```
sport_total_medals = df_medals.groupby('discipline_title').size().sort_values(ascending=False)
top_10_sports = sport_total_medals.head(10)

plt.figure(figsize=(12, 6))
sns.barplot(x=top_10_sports.index, y=top_10_sports.values)
plt.xticks(rotation=45)
plt.title("Top 10 Sports by Total Medal Count (All Countries)")
plt.ylabel("Total Medal Count")
plt.xlabel("Sport")
plt.tight_layout()
plt.show()
```


Top 10 Sports by Total Medal Count (All Countries)
Total Medal Count
Athletics
Swimming
Wrestling
Rowing
Boxing
Gymnastics Artistic
Shooting
Canoe Sprint
Sailing
Weightlifting
Sport