

**OLYMPIC SPORTS PARTICIPATION AND PERFORMANCE**

**DATA ANALYTICS**

IMPORTING LIBRARIES

import numpy as np

import pandas as pd

import seaborn as sns

from matplotlib import pyplot as plt

understanding database

Link of dataset: https://drive.google.com/drive/folders/19dCkREUadXgET8rbdKyhJ4nYgLcK-j\_a

DATA CLEANING AND FORMATING

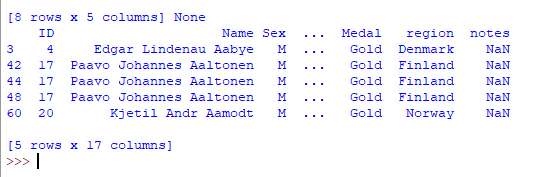
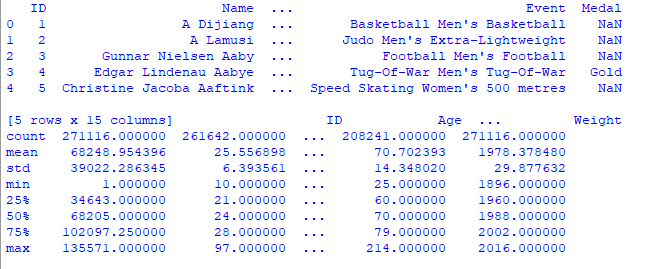
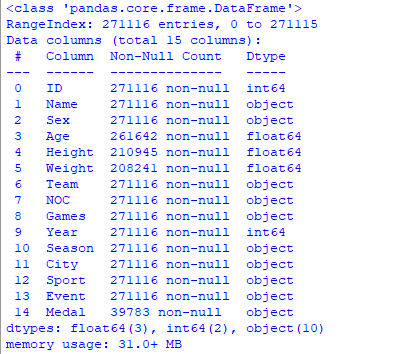
data **=** pd.read\_csv('athlete\_events.csv')  # read file

# data.head() display first 5 entry

# data.describe  about model

# data.info give info about data print(data.head(), data.describe(),data.info())

**Output:**



MERGING TWO DATAFRAMES

# regions and country noc data csv file

regions **=** pd.read\_csv('datasets\_31029\_40943\_noc\_regions.csv')

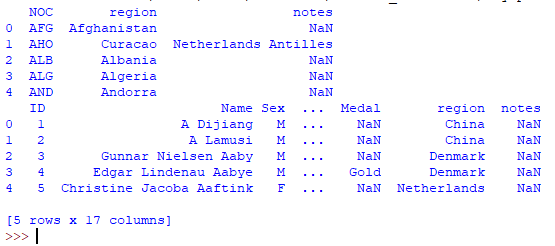
print(regions.head())

# merging to data and regions frame

merged **=** pd.merge(data, regions, on**=**'NOC', how**=**'left')

print(merged.head())

OUTPUT:



DATA ANALYSIS OF OLYMPICS

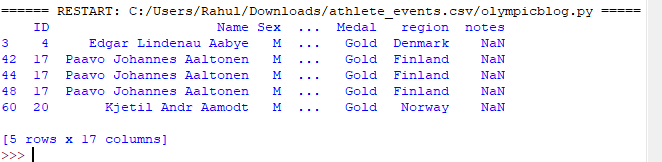
Data analysis of gold medalist

#creating goldmedal dataframes

GoldMedals **=** merged[(merged.Medal **==** 'Gold')]

print(goldMedals.head())

**Output :**



GOLD MEDALIST IN RESPECTIVE OF AGE

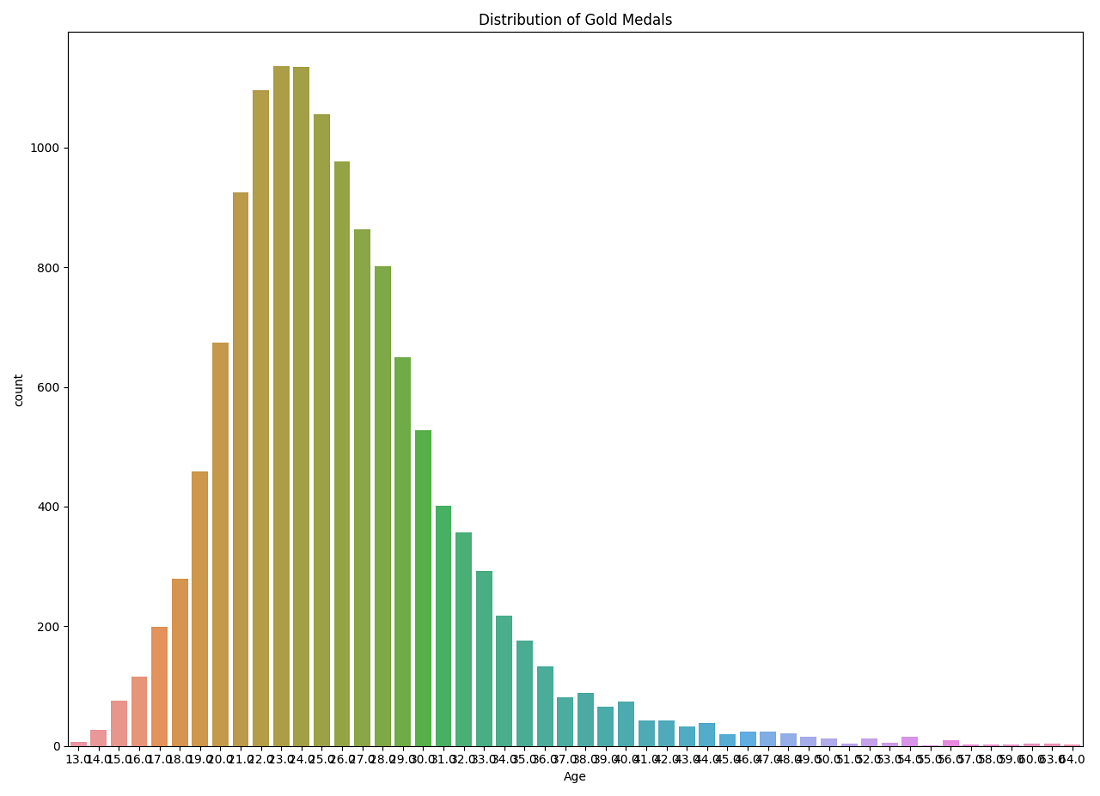
plt.figure(figsize**=**(20, 10))

plt.title('Distribution of Gold Medals')

sns.countplot(goldMedals['Age'])

plt.show

OUTPUT:



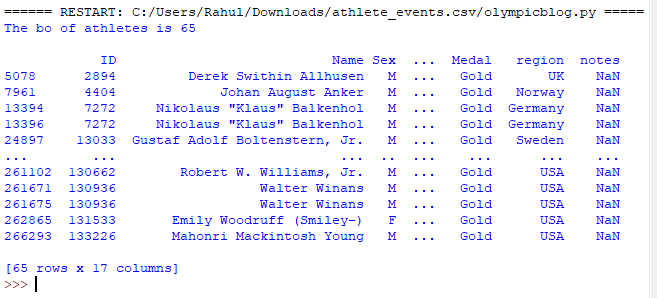
PRINT THE NUMBER OF ATELETES WHO ARE GOLD MEDALIST AND WHOSE AGE IS GREATER THEN 50 WITH THERE INFO :

goldMedals **=** merged[(merged.Medal **==** 'Gold')]

print('The no of athletes is’, goldMedals['ID'][goldMedals['Age'] > 50].count(), '\n')

print(goldMedals[goldMedals['Age'] > 50])

**Output :**



Create a new Dataframe called Master Disciplines in which we will insert this new set of people and then create a visualization with it

masterDisciplines **=** goldMedals['Sport'][goldMedals['Age'] > 50]

plt.figure(figsize**=**(20, 10))

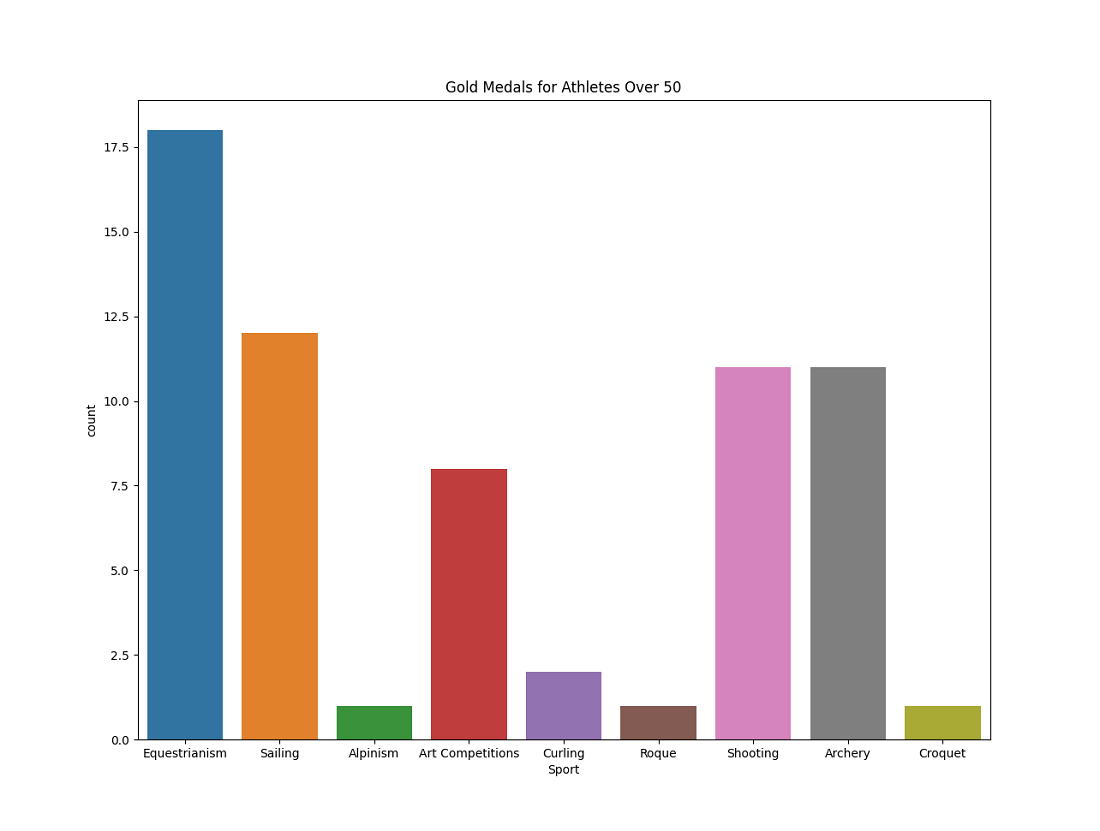
plt.tight\_layout()

sns.countplot(masterDisciplines)

plt.title('Gold Medals for Athletes Over 50')

plt.show()

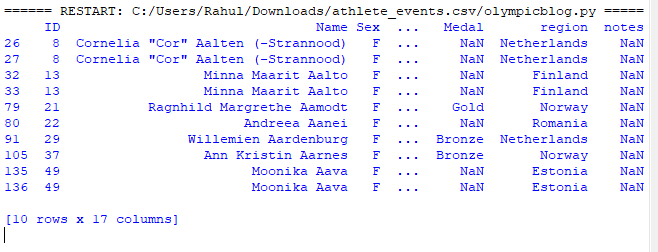
**Output :**



**Women who can play in summer**

|  |
| --- |
| WOMENInOlympics **=** merged[(merged.Sex **==** 'F') &                           (merged.Season **==** 'Summer')]  print(womenInOlympics.head(10))    sns.set(style**=**"darkgrid")  plt.figure(figsize**=**(20, 10))  sns.countplot(x**=**'Year', data**=**womenInOlympics)  plt.title('Women medals per edition of the Games')  plt.show() |

**Output :**



### Top  5 countries who won the most medals

**print**(goldMedals.region.value\_counts().reset\_index(name**=**'Medal').head())

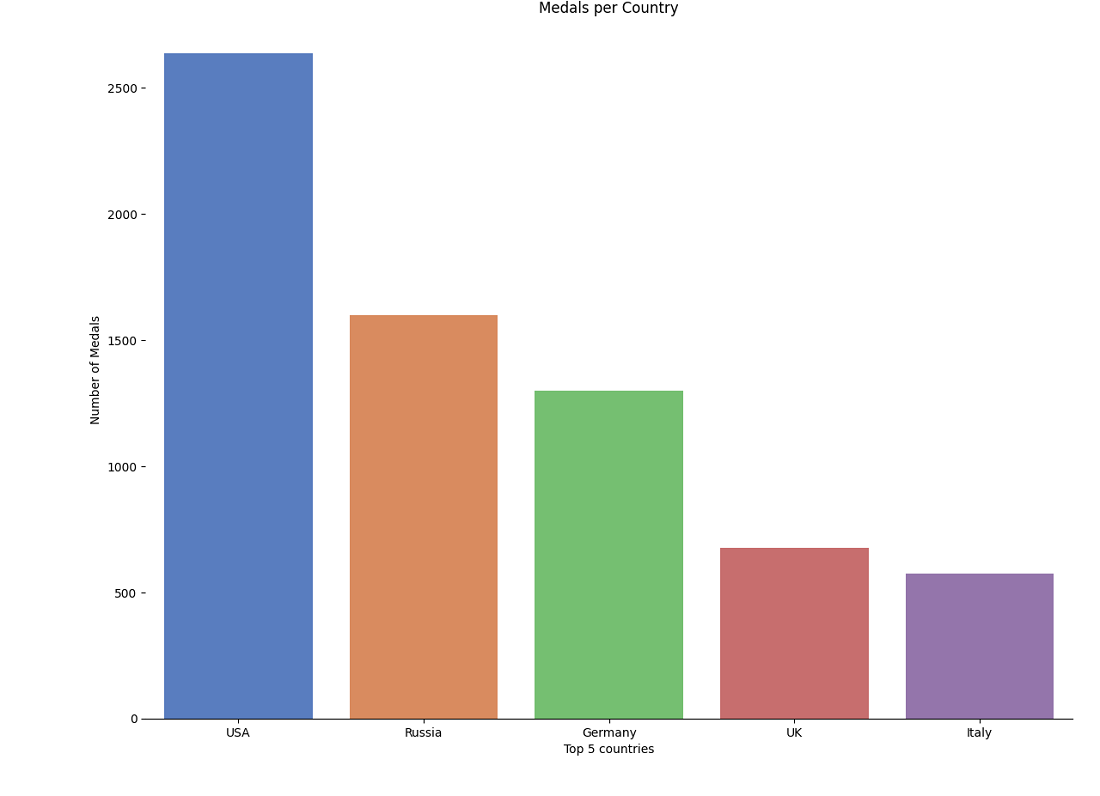
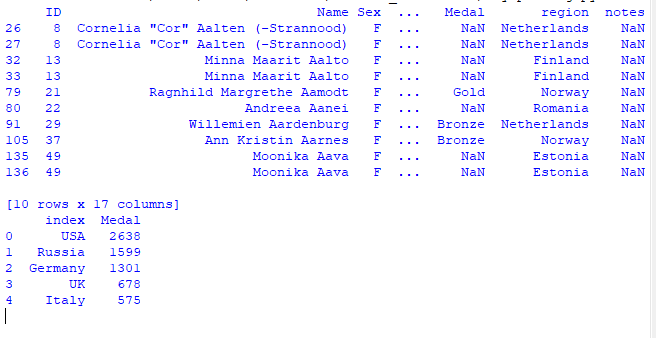
totalGoldMedals **=** goldMedals.region.value\_counts()

.reset\_index(name**=**'Medal').head(5)

g **=** sns.catplot(x**=**"index", y**=**"Medal", data**=**totalGoldMedals,height**=**6, kind**=**"bar", palette**=**"muted")g.despine(left**=**True)g.set\_xlabels("Top 5 countries")g.set\_ylabels("Number of Medals")plt.title('Medals per Country')

plt.show()

**Output:**



### Players weight Analysis

MenOverTime **=** merged[(merged.Sex **==** 'M') &

                     (merged.Season **==** 'Summer')]

wlMenOverTime **=** MenOverTime.loc[MenOverTime['Sport'] **==** 'Weightlifting']

plt.figure(figsize**=**(20, 10))

sns.pointplot('Year', 'Weight', data**=**wlMenOverTime, palette**=**'Set2')

plt.title('Weight over year for Male Lifters')

plt.show()

**Output :**

