**Name:Sujit S Jayappa**

**TrackCode:DS**

**Task1:Create a bar chart to visualize the distribution of a categorical or continous variable**

1. from google.colab import drive
2. drive.mount('/content/drive')

3

Mounted at /content/drive

1 %cd /content/drive/MyDrive/datasets\_for\_coding/

2

/content/drive/MyDrive/datasets\_for\_coding

1. import pandas as pd
2. import seaborn as sns
3. import numpy as np
4. df = pd.read\_csv('Summer-Olympic-medals-1976-to-2008.csv',sep=',', encoding='latin-1')
5. df.head()

**City**

**Year**

**Sport**

**Discipline**

**Event**

**Athlete**

**Gender**

**Country\_Code**

**Country**

**Event\_g**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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 |

10m

1 df.shape

 (15433, 11)

1 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

1 df.dropna(inplace=True)

1. %matplotlib inline
2. from matplotlib import pyplot as plt
3. plt.figure(figsize=(10, 5))
4. sns.countplot(df['Year'])

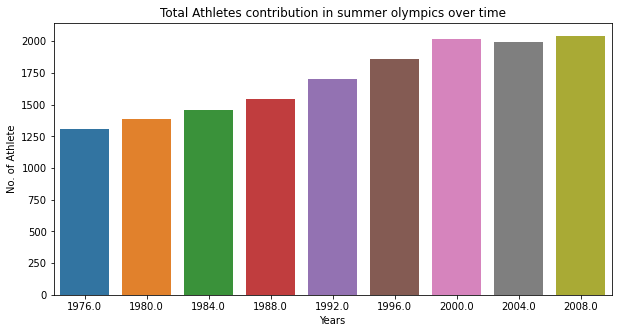
|  |  |  |  |
| --- | --- | --- | --- |
| KÖHLER,  Christa | Women | GDR | East  Germany |
| KOSENKOV,  Aleksandr | Men | URS | Soviet Union |
| BOGGS,  Philip  George  CAGNOTTO, | Men | USA | United  States |
| Giorgio  Franco  WILSON, | Men | ITA | Italy |

United

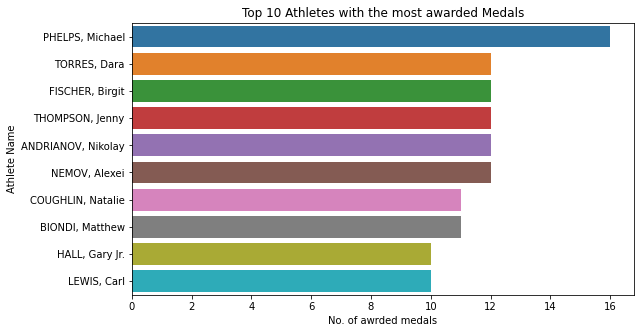
1. plt.title('Total Athletes contribution in summer olympics over time')
2. plt.xlabel('Years')
3. plt.ylabel('No. of Athlete')
4. /usr/local/lib/python3.7/dist-packages/seaborn/\_decorators.py:43: FutureWarning:

Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument

Text(0, 0.5, 'No. of Athlete')



1. athlete\_order = df['Athlete'].value\_counts().head(10).index
2. plt.figure(figsize=(9, 5))
3. sns.countplot(data=df, y='Athlete', order=athlete\_order)
4. plt.title('Top 10 Athletes with the most awarded Medals') 5 plt.xlabel('No. of awrded medals')



6

plt.ylabel

(

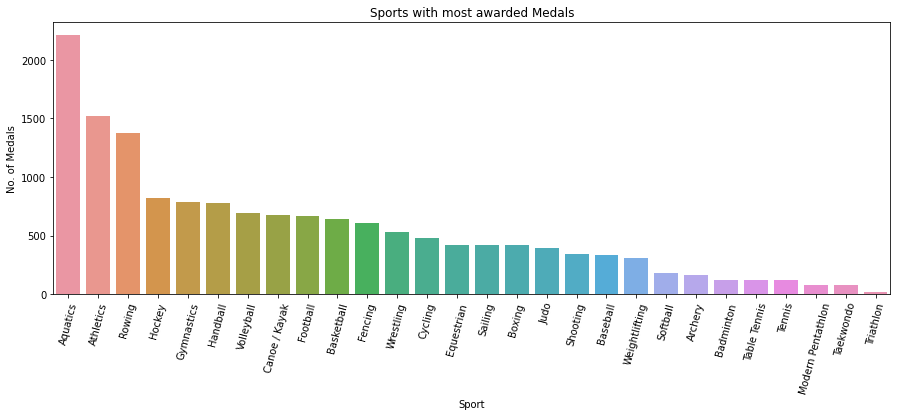
'Athlete Name'

)

;

1. plt.figure(figsize=(15, 5))
2. highest\_sport = df['Sport'].value\_counts().index
3. sns.countplot(data=df, x='Sport', order=highest\_sport)
4. plt.xticks(rotation=75)
5. plt.title('Sports with most awarded Medals')
6. plt.xlabel('Sport')
7. plt.ylabel('No. of Medals')

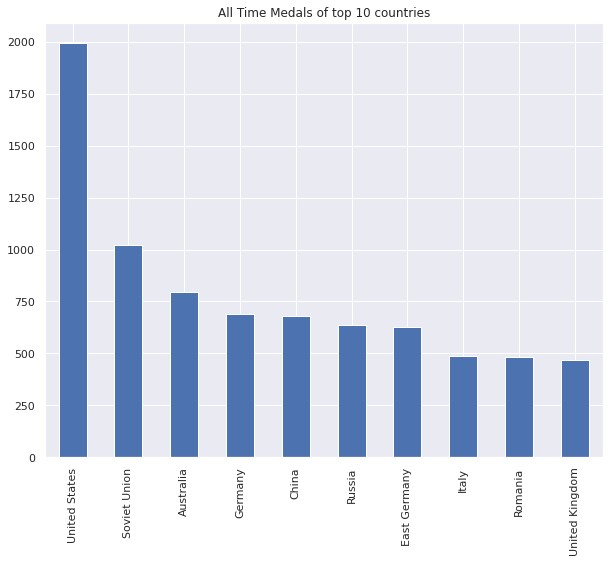
Text(0, 0.5, 'No. of Medals')



1 Start coding or generate with AI.

1. top\_10 = df['Country'].value\_counts()[:10]
2. top\_10.plot(kind='bar',figsize=(10,8))

Text(0.5, 1.0, 'All Time Medals of top 10 countries')



3

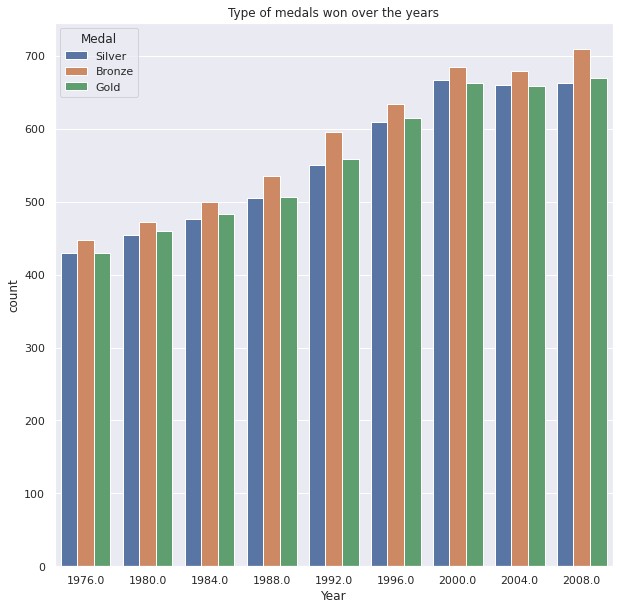
plt.title(

'All Time Medals of top 10 countries'

)

1. sns.countplot(x='Year',hue='Medal',data=df)
2. sns.set(rc={'figure.figsize':(10,10)})
3. plt.title("Type of medals won over the years")

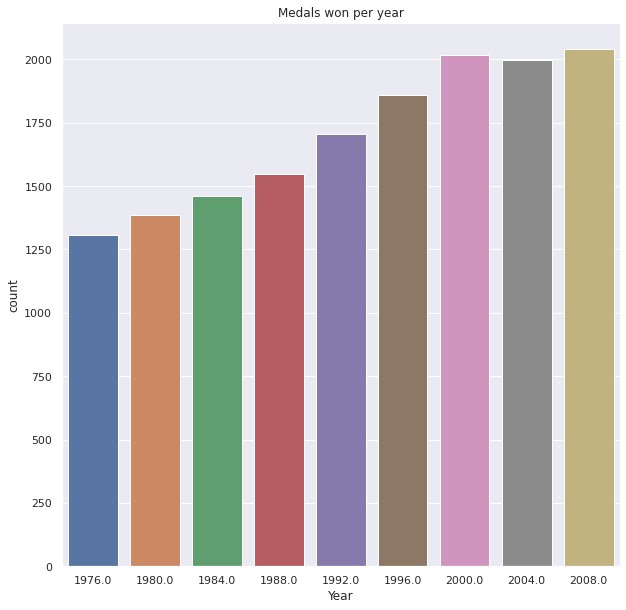
Text(0.5, 1.0, 'Type of medals won over the years')



1. sns.countplot(x='Year',data=df)
2. sns.set(rc={'figure.figsize':(10,10)}).plot(kind='bar',figsize=(10,8))

3

Text(0.5, 1.0, 'Medals won per year')



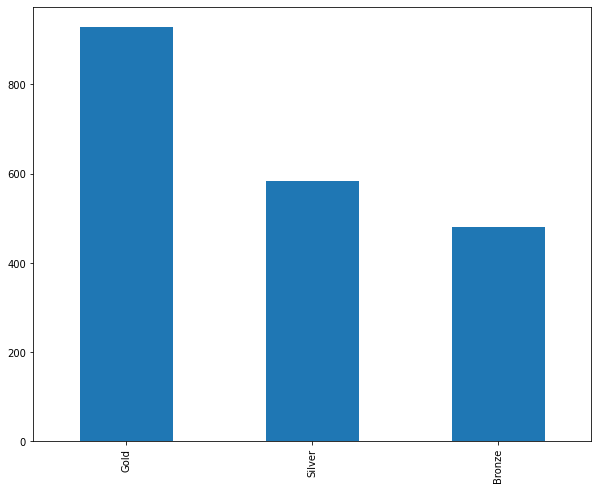
4

plt.title(

"Medals won per year"

)

1. indpie = df[df['Country']=='United States']['Medal'].value\_counts()
2. indpie.plot(kind='bar',figsize=(10,8)) <matplotlib.axes.\_subplots.AxesSubplot at 0x7faacbf0c3d0>

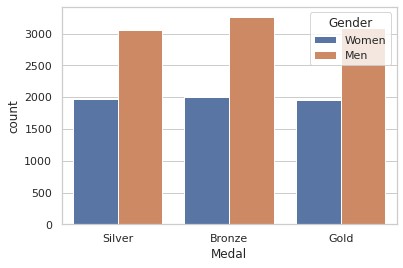


1 sns.countplot(x="Medal", hue="Gender", data=df)

<

matplotlib.axes.\_subplots.AxesSubplot at 0x7f49829be

490>

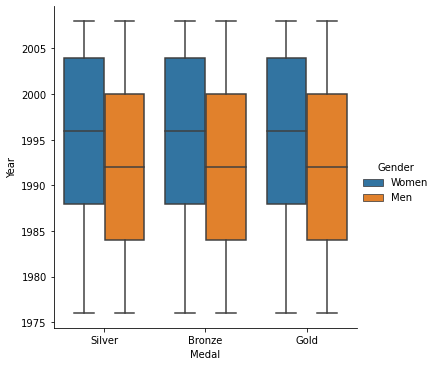


1 sns.catplot(x="Medal", y="Year", hue="Gender",kind="box", data=df)

<

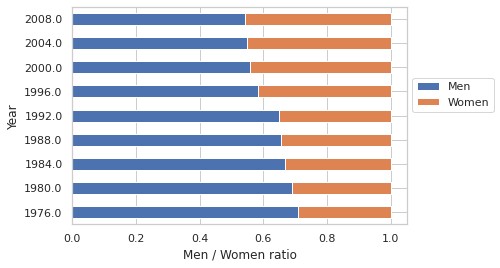
seaborn.axisgrid.FacetGrid at 0x7faac

0798550>



1. gender\_group = df.groupby(['Year', 'Gender']).size().unstack()
2. gender\_group.apply(lambda x:x/x.sum(), axis=1).plot(kind='barh', stacked=True, legend=False)
3. plt.legend(['Men', 'Women'], bbox\_to\_anchor=(1.0, 0.7))
4. plt.xlabel('Men / Women ratio')

Text(0.5, 0, 'Men / Women ratio')



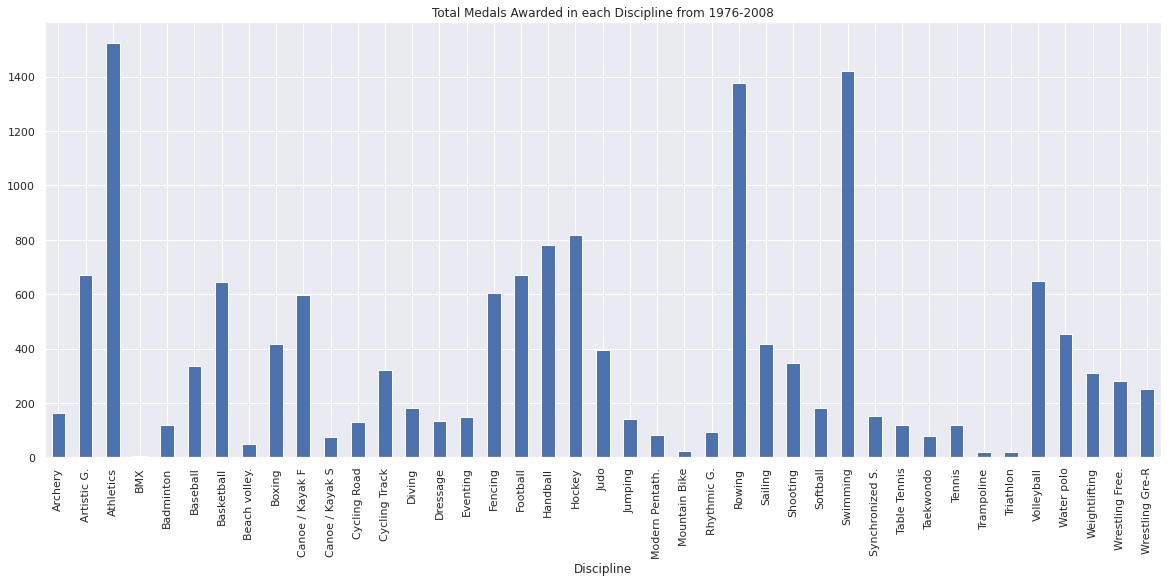
1. p = df.groupby('Discipline').agg('count')
2. p.plot(y='Medal',kind='bar',legend=False,title='Total Medals Awarded in each Discipline from 1976-2008',figsize=(20,8))

3

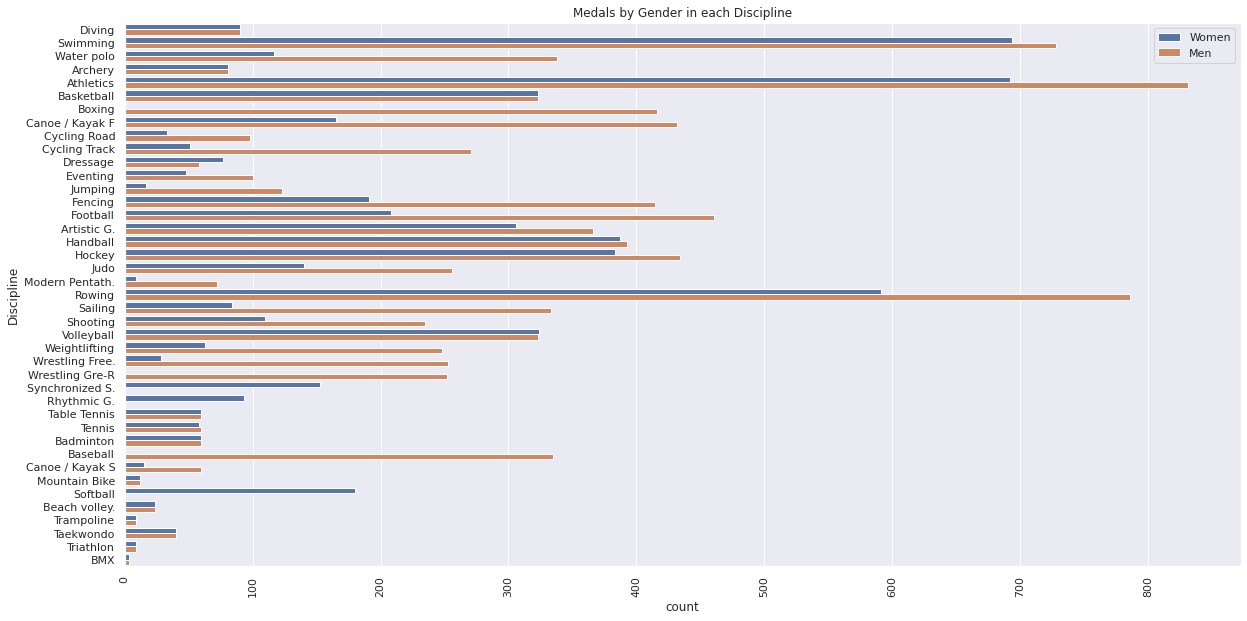
<

matplotlib.axes.\_subplots.AxesSubplot at 0x7f497ed

8d350>



1. sns.countplot(y='Discipline',hue='Gender',data=df)
2. sns.set(rc={'figure.figsize':(10,10)})
3. plt.xticks(rotation=90)
4. plt.title('Medals by Gender in each Discipline') 5 plt.legend(loc=1) # 1 is code for 'upper right' <matplotlib.legend.Legend at 0x7f49665a4190>



1. import numpy as np
2. import matplotlib.pyplot as plt

3

4

1. # Dataset generation
2. data\_dict = {'CSE':33, 'ECE':28, 'EEE':30}
3. courses = list(data\_dict.keys())
4. values = list(data\_dict.values())

9

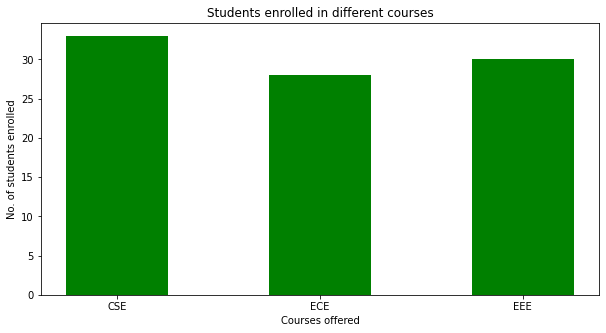
10 fig = plt.figure(figsize = (10, 5))

11

1. # Bar plot
2. plt.bar(courses, values, color ='green',
3. width = 0.5)

15

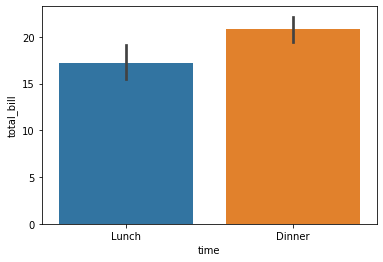
1. plt.xlabel("Courses offered")
2. plt.ylabel("No. of students enrolled")
3. plt.title("Students enrolled in different courses")



19

plt.show()

1. import seaborn as sns
2. import matplotlib.pyplot as plt
3. df = sns.load\_dataset('tips') 4 sns.barplot(x = 'time',
4. y = 'total\_bill',
5. data = df)
6. plt.show()



1. import plotly.express as px
2. data\_canada = px.data.gapminder().query("country == 'Canada'")
3. fig = px.bar(data\_canada, x='year', y='pop')
4. fig.show()

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

35M

pop

30M

25M

20M

15M

10M 5M

0

1950 1960 1970 1980 1990 2000

year

1. import pandas as pd
2. plotdata = pd.DataFrame({ 3 "2018":[57,67,77,83],
3. "2019":[68,73,80,79],
4. "2020":[73,78,80,85]},
5. index=["Django", "Gafur", "Tommy", "Ronnie"])
6. plotdata.plot(kind="bar",figsize=(15, 8))
7. plt.title("FIFA ratings")
8. plt.xlabel("Footballer")
9. plt.ylabel("Ratings")

11

Text(0, 0.5, 'Ratings')

