In [1]:

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

In [2]:

```
athlete = pd.read_csv("C:/Users/Asus/Documents/Projects/Python/A data analysis resume pr
region = pd.read_csv("C:/Users/Asus/Documents/Projects/Python/A data analysis resume pro
```

In [3]:

```
athlete.head()
```

Out[3]:

	Unnamed: 0	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	
0	0	A Dijiang	M	24.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Bas Bas
1	1	A Lamusi	M	23.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Light
2	2	Gunnar Nielsen Aaby	M	24.0	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	F F
3	3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	T War T
4	26	Cornelia "Cor" Aalten (-Strannood)	F	18.0	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	At Wo 100

In [4]:

```
athlete.drop(['Unnamed: 0'], axis=1, inplace = True)
```

In [5]:

```
athlete.head()
```

Out[5]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Me
0	A Dijiang	M	24.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	N
1	A Lamusi	M	23.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	N
2	Gunnar Nielsen Aaby	M	24.0	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	N
3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	G
4	Cornelia "Cor" Aalten (-Strannood)	F	18.0	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 metres	N

In [6]:

```python
region.drop(['Unnamed: 0'], axis=1, inplace= True)
```

In [7]:

```python
region.head()
```

Out[7]:

	NOC	region	notes
0	EOR	Refugee	NaN
1	LBN	Lebanon	NaN
2	SGP	Singapore	NaN
3	ROC	Russia	NaN
4	AFG	Afghanistan	NaN

In [8]:

```python
# Merge Data set
df = athlete.merge(region, how = 'left', on = 'NOC')
df.head()
```

Out[8]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Mec
0	A Dijiang	M	24.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	N
1	A Lamusi	M	23.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	N
2	Gunnar Nielsen Aaby	M	24.0	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	N
3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	Gc
4	Cornelia "Cor" Aalten (-Strannood)	F	18.0	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 metres	N

In [9]:

```python
df.shape
```

Out[9]:

```
(237673, 14)
```

In [10]:

```python
df.rename(columns={'region': 'Region', 'notes': 'Notes'}, inplace=True)
```

In [11]:

```python
df.head()
```

Out[11]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Mec
0	A Dijiang	M	24.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	N
1	A Lamusi	M	23.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	N
2	Gunnar	M	24.0	Denmark	DEN	1920	1920	Summer	Antwerpen	Football	Football	N

	Nielsen Aaby					Summer					Men's Football	
3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	Go
4	Cornelia "Cor" Aalten (-Strannood)	F	18.0	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 metres	N

In [12]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 237673 entries, 0 to 237672
Data columns (total 14 columns):
 #   Column  Non-Null Count   Dtype  
---  ------  --------------   -----  
 0   Name    237673 non-null  object 
 1   Sex     237673 non-null  object 
 2   Age     228484 non-null  float64
 3   Team    237673 non-null  object 
 4   NOC     237673 non-null  object 
 5   Games   237673 non-null  object 
 6   Year    237673 non-null  int64  
 7   Season  237673 non-null  object 
 8   City    237673 non-null  object 
 9   Sport   237673 non-null  object 
 10  Event   237673 non-null  object 
 11  Medal   36537 non-null   object 
 12  Region  237650 non-null  object 
 13  Notes   4525 non-null    object 
dtypes: float64(1), int64(1), object(12)
memory usage: 27.2+ MB
```

In [13]:
```
#statistical Summary
df.describe()
```

Out[13]:

	Age	Year
count	228484.000000	237673.000000
mean	25.746267	1979.096246
std	6.638720	31.783967
min	10.000000	1896.000000
25%	21.000000	1960.000000
50%	25.000000	1988.000000
75%	29.000000	2004.000000
max	97.000000	2020.000000

In [14]:
```
#Check null values
nan_values = df.isna()
nan_columns = nan_values.any()
nan_columns
```

Out[14]:
```
Name       False
Sex        False
Age         True
```

```
Team       False
NOC        False
Games      False
Year       False
Season     False
City       False
Sport      False
Event      False
Medal       True
Region      True
Notes       True
dtype: bool
```

In [15]:

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

Out[15]:

```
Name           0
Sex            0
Age         9189
Team           0
NOC            0
Games          0
Year           0
Season         0
City           0
Sport          0
Event          0
Medal     201136
Region        23
Notes     233148
dtype: int64
```

In [16]:

```
# Print the columns name conttaining null values orr missing valuess in the form of a li
athletes_null_culumns = df.columns[df.isnull().any()].tolist()
athletes_null_culumns
```

Out[16]:

```
['Age', 'Medal', 'Region', 'Notes']
```

In [17]:

```
#India details
df.query('Team == "India"').head()
```

Out[17]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Medal	Region
436	S. Abdul Hamid	M	NaN	India	IND	1928 Summer	1928	Summer	Amsterdam	Athletics	Athletics Men's 110 metres Hurdles	NaN	India
437	S. Abdul Hamid	M	NaN	India	IND	1928 Summer	1928	Summer	Amsterdam	Athletics	Athletics Men's 400 metres Hurdles	NaN	India
790	Shiny Kurisingal Abraham-Wilson	F	19.0	India	IND	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 800 metres	NaN	India
791	Shiny Kurisingal Abraham-Wilson	F	19.0	India	IND	1984 Summer	1984	Summer	Los Angeles	Athletics	Athletics Women's 4 x 400 metres Relay	NaN	India
792	Shiny	F	23.0	India	IND	1988	1988	Summer	Seoul	Athletics	Athletics	NaN	India

Kurisingal Abraham-Wilson	Summer	Women's 800 metres

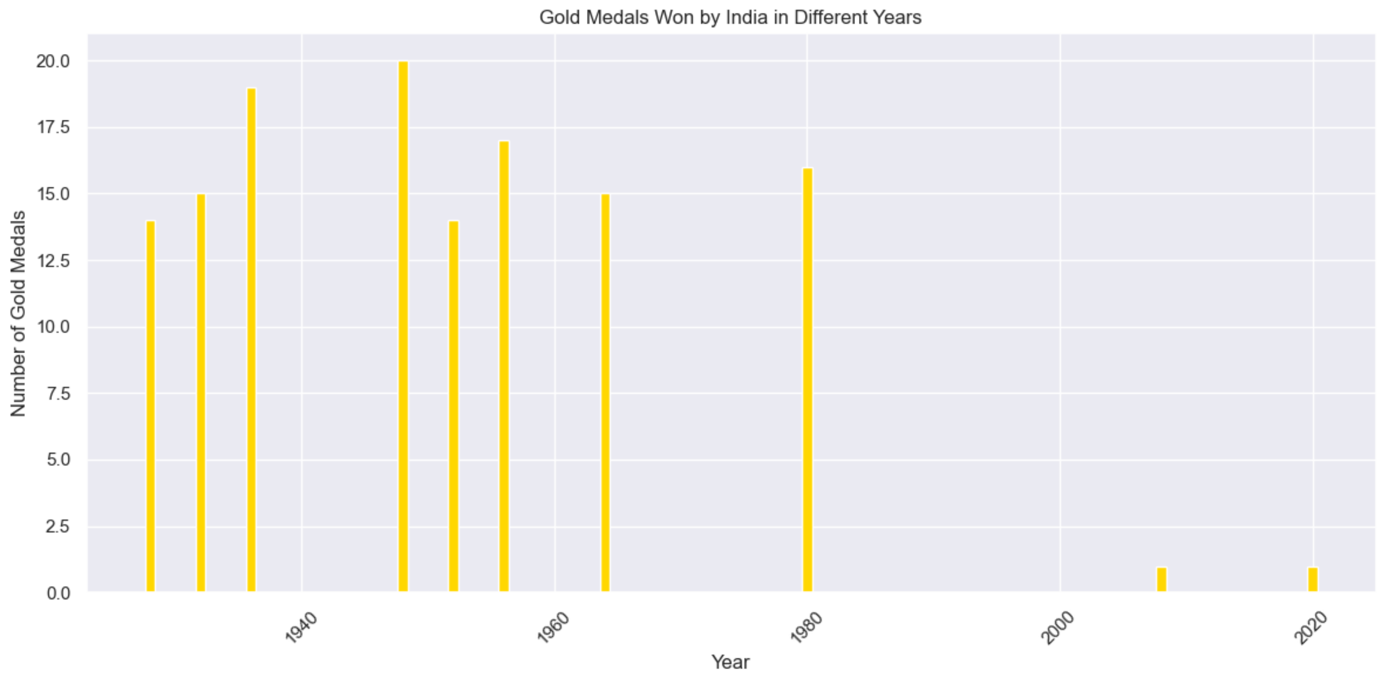
In [18]:

```
india_participation = df.query('Team == "India"')


# Finding years in which India won gold medals
india_gold_years = india_participation[india_participation['Medal'] == 'Gold']['Year'].u
print("Years in which India won gold medals:", india_gold_years)
```

```
Years in which India won gold medals: [1928 1964 1932 1936 1980 2008 1948 1952 1956 202
0]
```

In [43]:

```
gold_medal_counts = india_participation[india_participation['Medal'] == 'Gold']['Year'].

# Create a bar plot
plt.figure(figsize=(12, 6))
plt.bar(gold_medal_counts.index, gold_medal_counts.values, color='gold')
plt.xlabel('Year')
plt.ylabel('Number of Gold Medals')
plt.title('Gold Medals Won by India in Different Years')
plt.xticks(rotation=45)
plt.tight_layout()

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



In [20]:

```
#Japan details
df.query('Team == "Japan"').head()
```

Out[20]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Medal	Region
556	Isao Ko Abe	M	24.0	Japan	JPN	1936 Summer	1936	Summer	Berlin	Athletics	Athletics Men's Hammer Throw	NaN	Japan
557	Kazuo Abe	M	25.0	Japan	JPN	1960 Summer	1960	Summer	Roma	Wrestling	Wrestling Men's Lightweight, Freestyle	NaN	Japan

558	Kinya Abe	M	23.0	Japan	JPN	1992 Summer	1992	Summer	Barcelona	Fencing	Fencing Men's Foil, Individual	NaN	Japan
559	Kiyoshi Abe	M	25.0	Japan	JPN	1972 Summer	1972	Summer	Munich	Wrestling	Wrestling Men's Featherweight, Freestyle	NaN	Japan
560	Naoki Abe	M	23.0	Japan	JPN	1968 Summer	1968	Summer	Mexico City	Athletics	Athletics Men's 4 x 100 metres Relay	NaN	Japan

In [21]:

```
#Top Countries Participating
top_10_countries = df.Team.value_counts().sort_values(ascending = False).head(10)
top_10_countries
```

Out[21]:

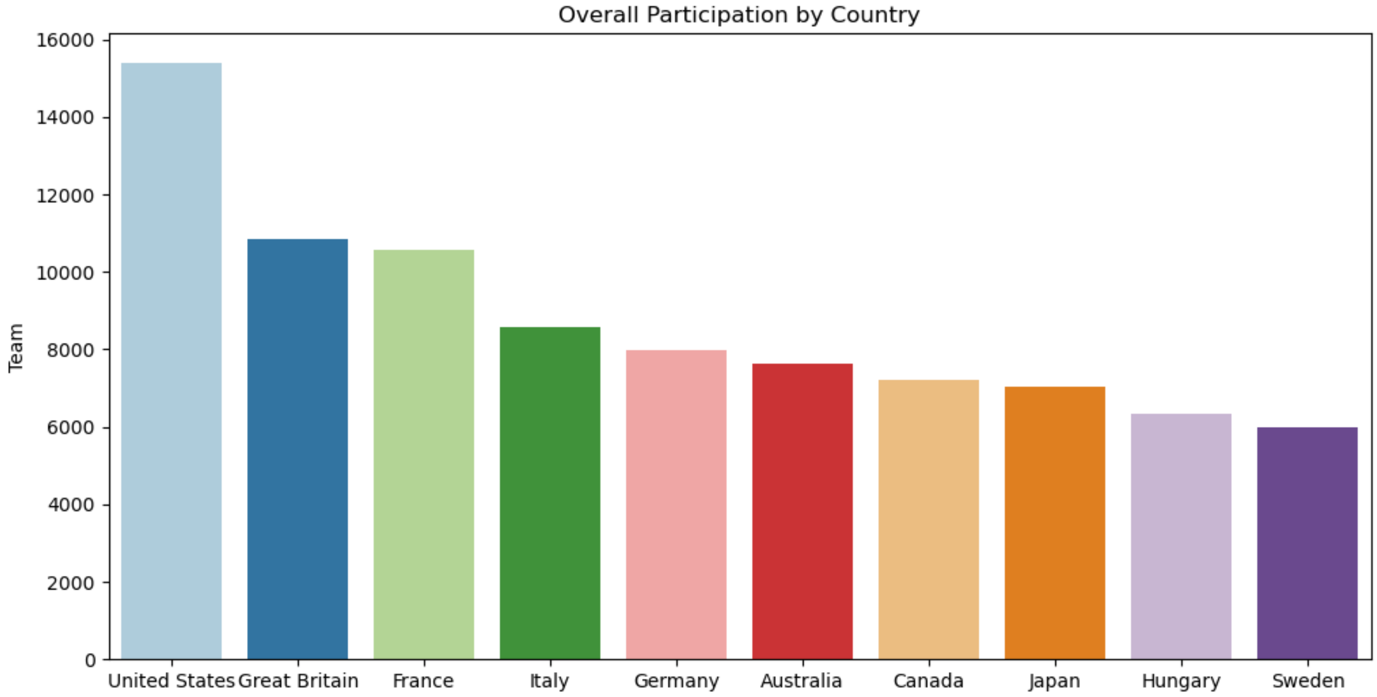
```
United States    15382
Great Britain    10857
France           10559
Italy             8575
Germany           7975
Australia         7614
Canada            7198
Japan             7020
Hungary           6326
Sweden            5994
Name: Team, dtype: int64
```

In [22]:

```
#Plot a grapgh for top 10 countries

plt.figure(figsize=(12,6))
plt.title('Overall Participation by Country')
sns.barplot(x=top_10_countries.index, y= top_10_countries, palette = 'Paired')
```

Out[22]:

```
<Axes: title={'center': 'Overall Participation by Country'}, ylabel='Team'>
```

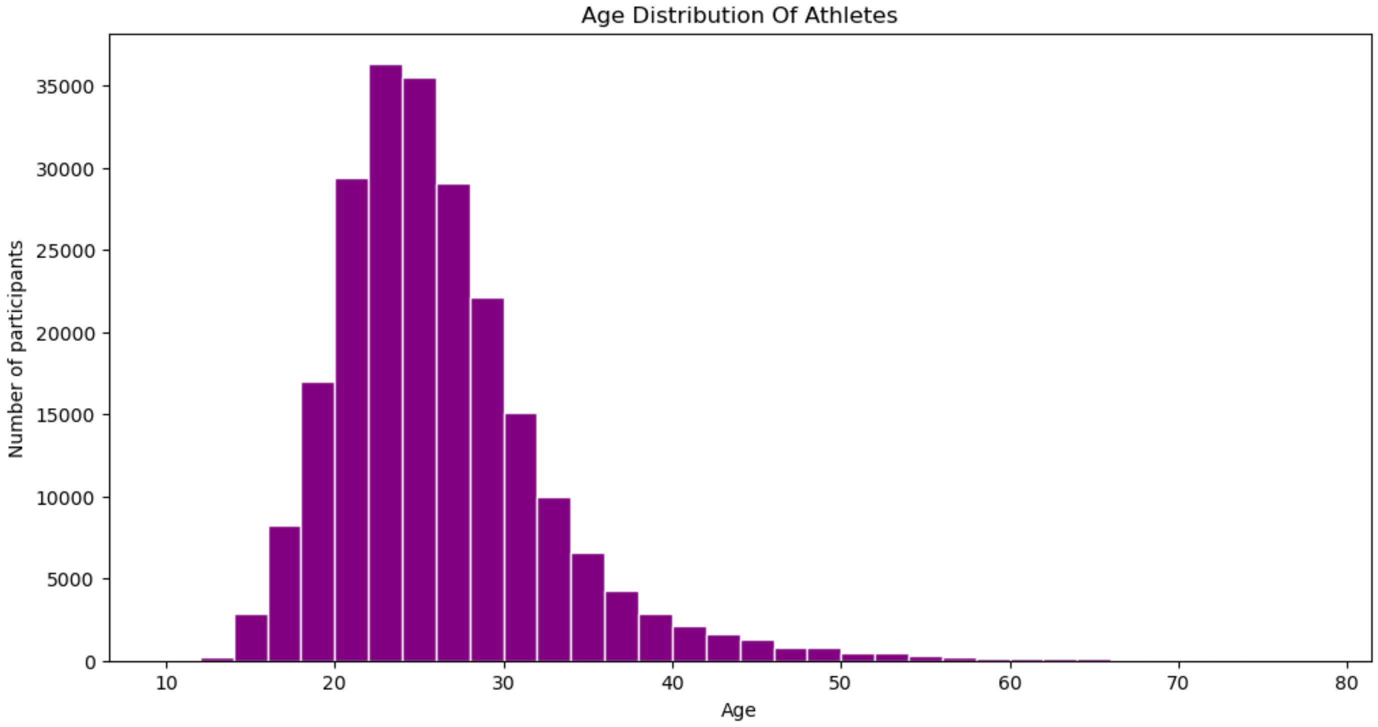


In [23]:

```
#Age distribution of the participants

plt.figure(figsize=(12,6))
plt.title("Age Distribution Of Athletes")
plt.xlabel("Age")
plt.ylabel("Number of participants")
plt.hist(df.Age, bins = np.arange(10,80,2), color = "purple", edgecolor="white")
```

Out[23]:

```
(array([7.0000e+00, 2.0800e+02, 2.8590e+03, 8.2140e+03, 1.7025e+04,
        2.9375e+04, 3.6343e+04, 3.5487e+04, 2.9081e+04, 2.2137e+04,
        1.5108e+04, 9.9900e+03, 6.6050e+03, 4.2350e+03, 2.8960e+03,
        2.1330e+03, 1.6420e+03, 1.2660e+03, 8.3400e+02, 7.6500e+02,
        4.9200e+02, 4.5400e+02, 2.7400e+02, 2.1300e+02, 1.7600e+02,
        1.5900e+02, 1.2000e+02, 1.1400e+02, 5.8000e+01, 8.5000e+01,
        6.1000e+01, 3.2000e+01, 1.6000e+01, 9.0000e+00]),
 array([10., 12., 14., 16., 18., 20., 22., 24., 26., 28., 30., 32., 34.,
        36., 38., 40., 42., 44., 46., 48., 50., 52., 54., 56., 58., 60.,
        62., 64., 66., 68., 70., 72., 74., 76., 78.]),
 <BarContainer object of 34 artists>)
```



In [24]:

```
df.head()
```

Out[24]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Mec
0	A Dijiang	M	24.0	China	CHN	1992 Summer	1992	Summer	Barcelona	Basketball	Basketball Men's Basketball	N
1	A Lamusi	M	23.0	China	CHN	2012 Summer	2012	Summer	London	Judo	Judo Men's Extra-Lightweight	N
2	Gunnar Nielsen Aaby	M	24.0	Denmark	DEN	1920 Summer	1920	Summer	Antwerpen	Football	Football Men's Football	N
3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's Tug-Of-War	G
4	Cornelia "Cor" Aalten (-Strannood)	F	18.0	Netherlands	NED	1932 Summer	1932	Summer	Los Angeles	Athletics	Athletics Women's 100 metres	N

In [25]:

```
#summer olympics sports
```

```
sports = df[df.Season=="Summer"].Sport.unique()
sports
```

Out[25]:

```
array(['Basketball', 'Judo', 'Football', 'Tug-Of-War', 'Athletics',
       'Swimming', 'Badminton', 'Sailing', 'Gymnastics',
       'Art Competitions', 'Handball', 'Weightlifting', 'Wrestling',
       'Water Polo', 'Hockey', 'Rowing', 'Fencing', 'Equestrianism',
       'Shooting', 'Boxing', 'Taekwondo', 'Cycling', 'Diving', 'Canoeing',
       'Tennis', 'Modern Pentathlon', 'Golf', 'Softball', 'Archery',
       'Volleyball', 'Synchronized Swimming', 'Table Tennis', 'Baseball',
       'Rhythmic Gymnastics', 'Rugby Sevens', 'Trampolining',
       'Beach Volleyball', 'Triathlon', 'Rugby', 'Lacrosse', 'Polo',
       'Cricket', 'Ice Hockey', 'Racquets', 'Motorboating', 'Croquet',
       'Figure Skating', 'Jeu De Paume', 'Roque', 'Basque Pelota',
       'Alpinism', 'Aeronautics', 'Cycling Road', 'Artistic Gymnastics',
       'Karate', 'Baseball/Softball', 'Trampoline Gymnastics',
       'Marathon Swimming', 'Canoe Slalom', 'Surfing', 'Canoe Sprint',
       'Cycling BMX Racing', 'Equestrian', 'Artistic Swimming',
       'Cycling Track', 'Skateboarding', 'Cycling Mountain Bike',
       '3x3 Basketball', 'Cycling BMX Freestyle', 'Sport Climbing'],
      dtype=object)
```

In [26]:

```
#Male and Female Participants

gender_count = df.Sex.value_counts()
gender_count
```

Out[26]:

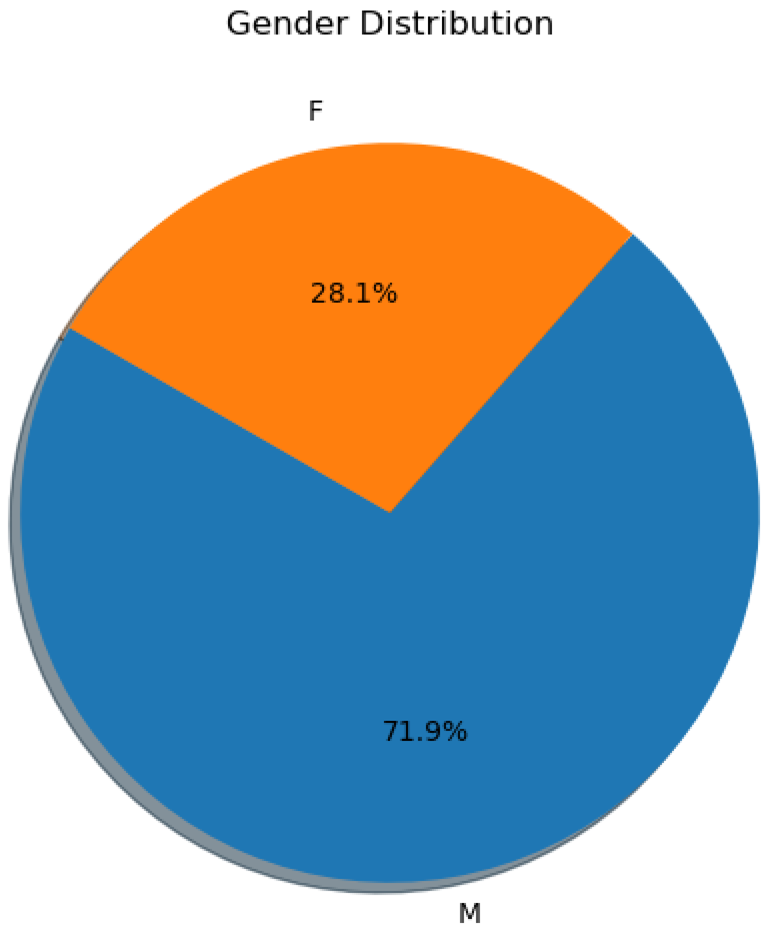
```
M    170964
F     66709
Name: Sex, dtype: int64
```

In [27]:

```
#Pie plot fro male and Female athletes

plt.figure(figsize = (12,6))
plt.title("Gender Distribution")
plt.pie(gender_count, labels= gender_count.index, autopct = '%1.1f%%', startangle=150, s
```

Out[27]:

```
([<matplotlib.patches.Wedge at 0x25603b5fd60>,
  <matplotlib.patches.Wedge at 0x25603b403d0>],
 [Text(0.1811434361817322, -1.084982513927425, 'M'),
  Text(-0.18114353776512454, 1.084982496967548, 'F')],
 [Text(0.0988055106445812, -0.5918086439604135, '71.9%'),
  Text(-0.09880556605370429, 0.5918086347095715, '28.1%')])
```


Gender Distribution

In [28]:

```
#Total Medals

df.Medal.value_counts()
```

Out[28]:

```
Bronze    12276
Gold      12259
Silver    12002
Name: Medal, dtype: int64
```

In [29]:

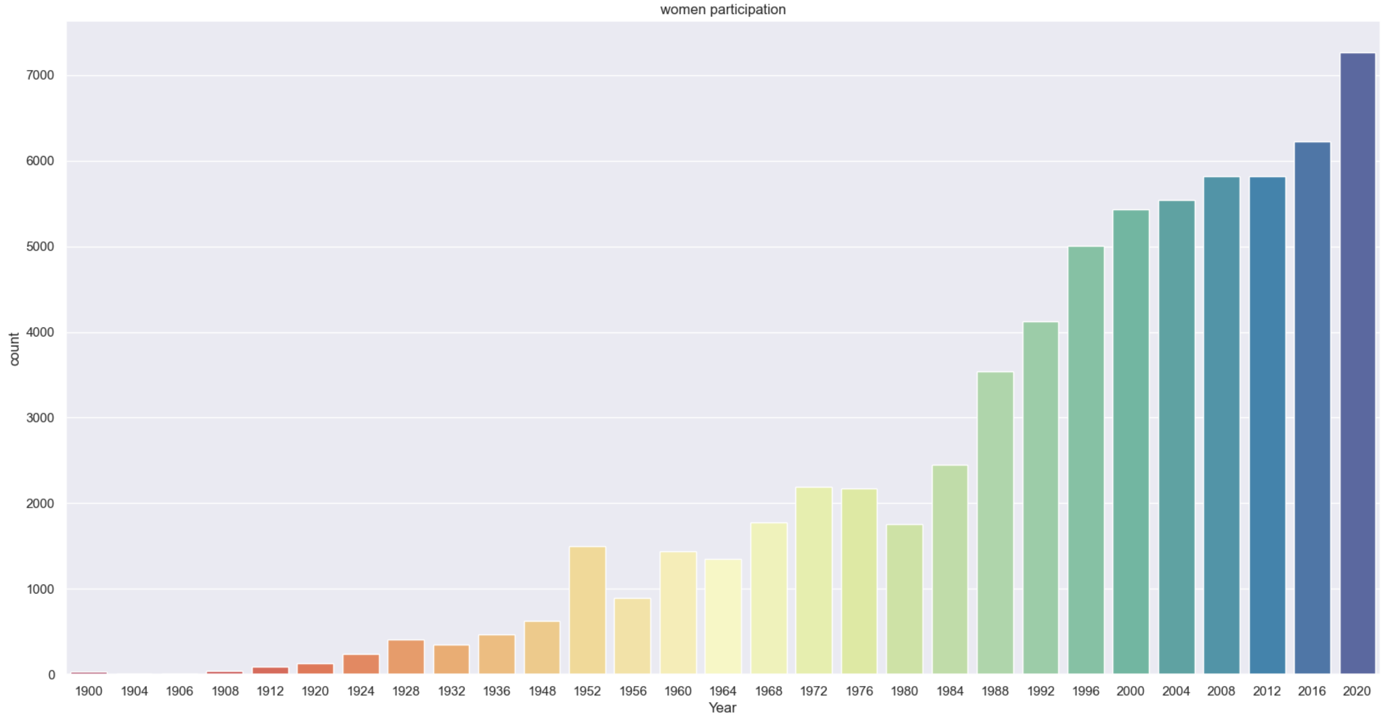
```
#Women Participation

women = df[(df.Sex=='F')]
```

In [30]:

```
sns.set(style = "darkgrid")
plt.figure(figsize=(20,10))
sns.countplot(x='Year', data=women, palette="Spectral")
plt.title('women participation')
```

Out[30]:

```
Text(0.5, 1.0, 'women participation')
```

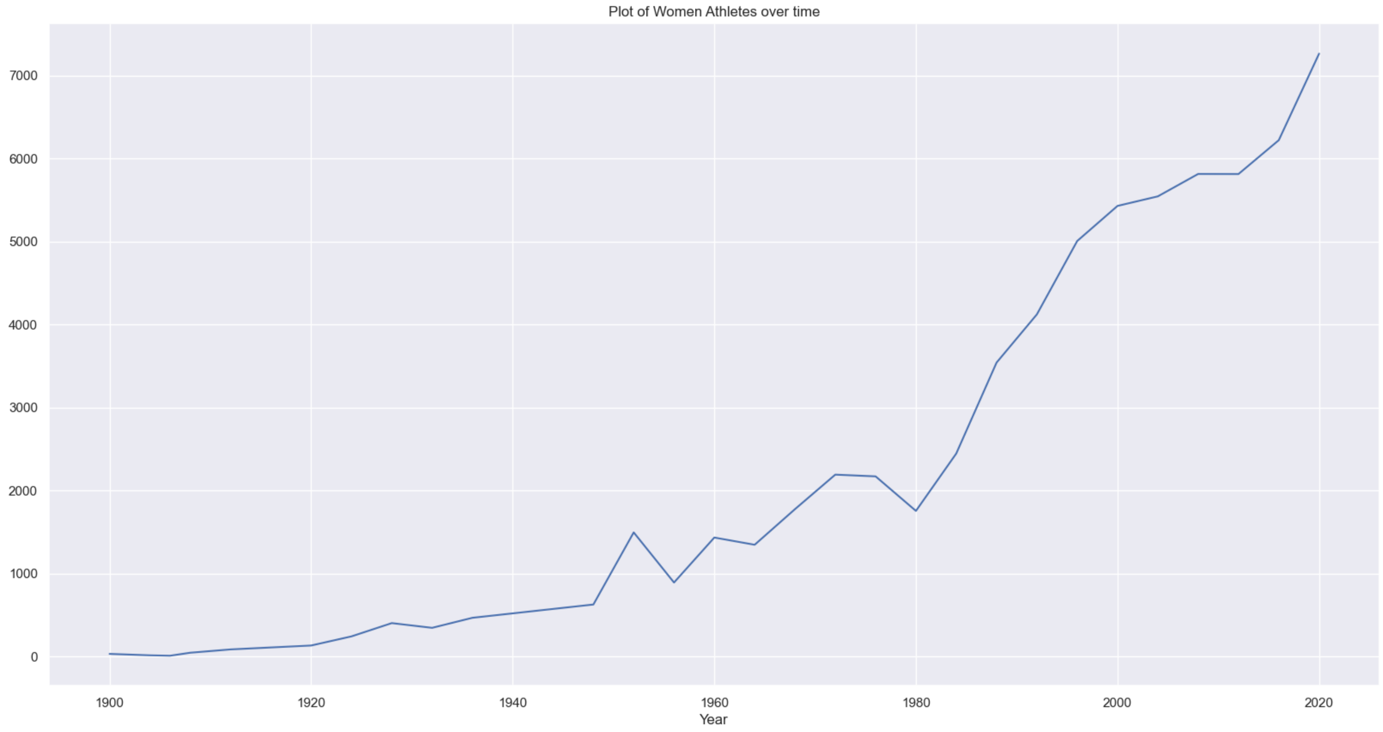


In [31]:

```
#Women Participation over the years

part = women.groupby('Year')['Sex'].value_counts()
plt.figure(figsize=(20,10))
part.loc[:,'F'].plot()
plt.title("Plot of Women Athletes over time")
```

Out[31]:

```
Text(0.5, 1.0, 'Plot of Women Athletes over time')
```



In [32]:

```
#Gold Medal
gold_medals = df[(df.Medal=='Gold')]
gold_medals.head()
```

Out[32]:

	Name	Sex	Age	Team	NOC	Games	Year	Season	City	Sport	Event	Meda
3	Edgar Lindenau Aabye	M	34.0	Denmark/Sweden	DEN	1900 Summer	1900	Summer	Paris	Tug-Of-War	Tug-Of-War Men's	Golc

													Tug-Of-War	
16	Paavo Johannes Aaltonen	M	28.0			Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Team All-Around	Golc
18	Paavo Johannes Aaltonen	M	28.0			Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Horse Vault	Golc
22	Paavo Johannes Aaltonen	M	28.0			Finland	FIN	1948 Summer	1948	Summer	London	Gymnastics	Gymnastics Men's Pommelled Horse	Golc
33	Ragnhild Margrethe Aamodt	F	27.0			Norway	NOR	2008 Summer	2008	Summer	Beijing	Handball	Handball Women's Handball	Golc

In [33]:

```
#take only the values that are differennt from NaN.

gold_medals = gold_medals[np.isfinite(gold_medals['Age'])]
```

In [34]:

```
#gold beyond 60

gold_medals['Name'][gold_medals['Age']>60].count()
```

Out[34]:

```
6
```

In [35]:

```
sporting_event = gold_medals['Sport'][gold_medals['Age']>60]
sporting_event
```

Out[35]:

```
85970     Art Competitions
86948                Roque
157206             Archery
185948             Archery
191460            Shooting
214409             Archery
Name: Sport, dtype: object
```

In [36]:

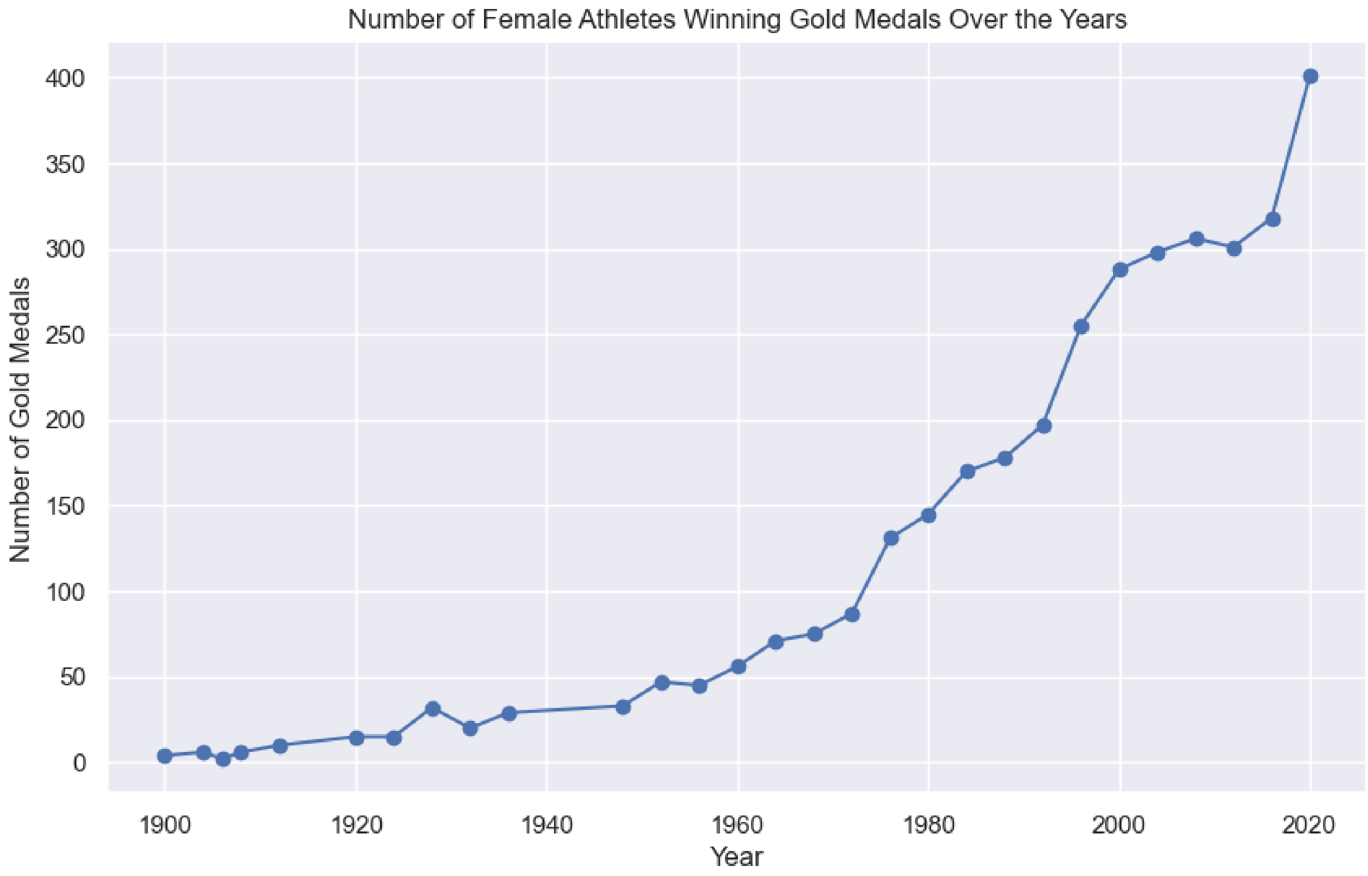
```
 #Total number of female athlete who won Gold Medal

female_participants = df[(df.Sex=='F') & (df.Medal=='Gold')][['Medal','Year']]
female_participants = female_participants.groupby('Year').count().reset_index()
female_participants.tail()
```

Out[36]:

	Year	Medal
24	2004	298
25	2008	306
26	2012	301
27	2016	318
28	2020	401

In [37]:

```
plt.figure(figsize=(10, 6))
plt.plot(female_participants['Year'], female_participants['Medal'], marker='o')
plt.title('Number of Female Athletes Winning Gold Medals Over the Years')
plt.xlabel('Year')
plt.ylabel('Number of Gold Medals')
plt.grid(True)
plt.show()
```



In [38]:

```
#golg medals from each country

gold_medals.Region.value_counts().reset_index(name='Medal').head()
```

Out[38]:

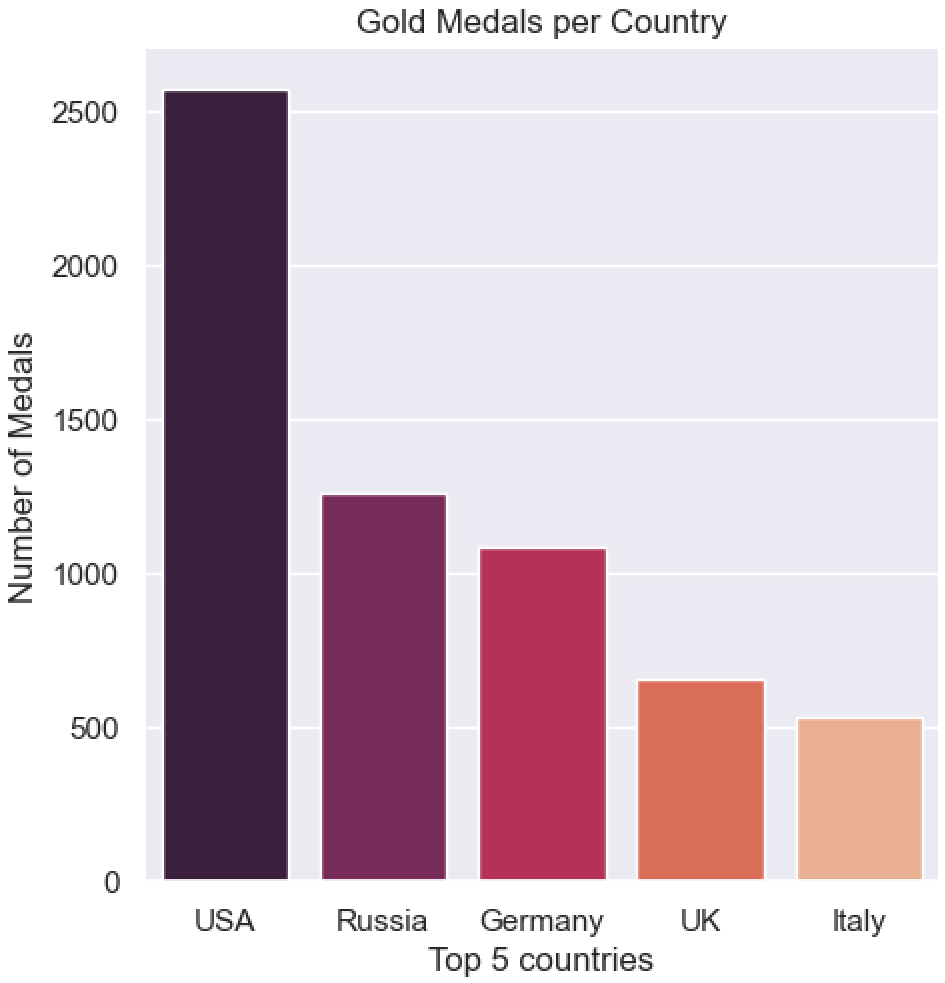
	index	Medal
0	USA	2574
1	Russia	1261
2	Germany	1088
3	UK	657
4	Italy	532

In [39]:

```
totalgoldmedals = gold_medals.Region.value_counts().reset_index(name='Medal').head()
g = sns.catplot(x="index", y="Medal", data= totalgoldmedals, height=5, kind= "bar", pale
g.despine(left= True)
g.set_xlabels("Top 5 countries")
g.set_ylabels("Number of Medals")
plt.title("Gold Medals per Country")
```

Out[39]:

```
Text(0.5, 1.0, 'Gold Medals per Country')
```



In [40]:

```
#Tokio

max_year = df.Year.max()
print(max_year)

team_names = df[(df.Year==max_year) & (df.Medal == "Gold")].Team
team_names.value_counts().head(10)
```

Out[40]:

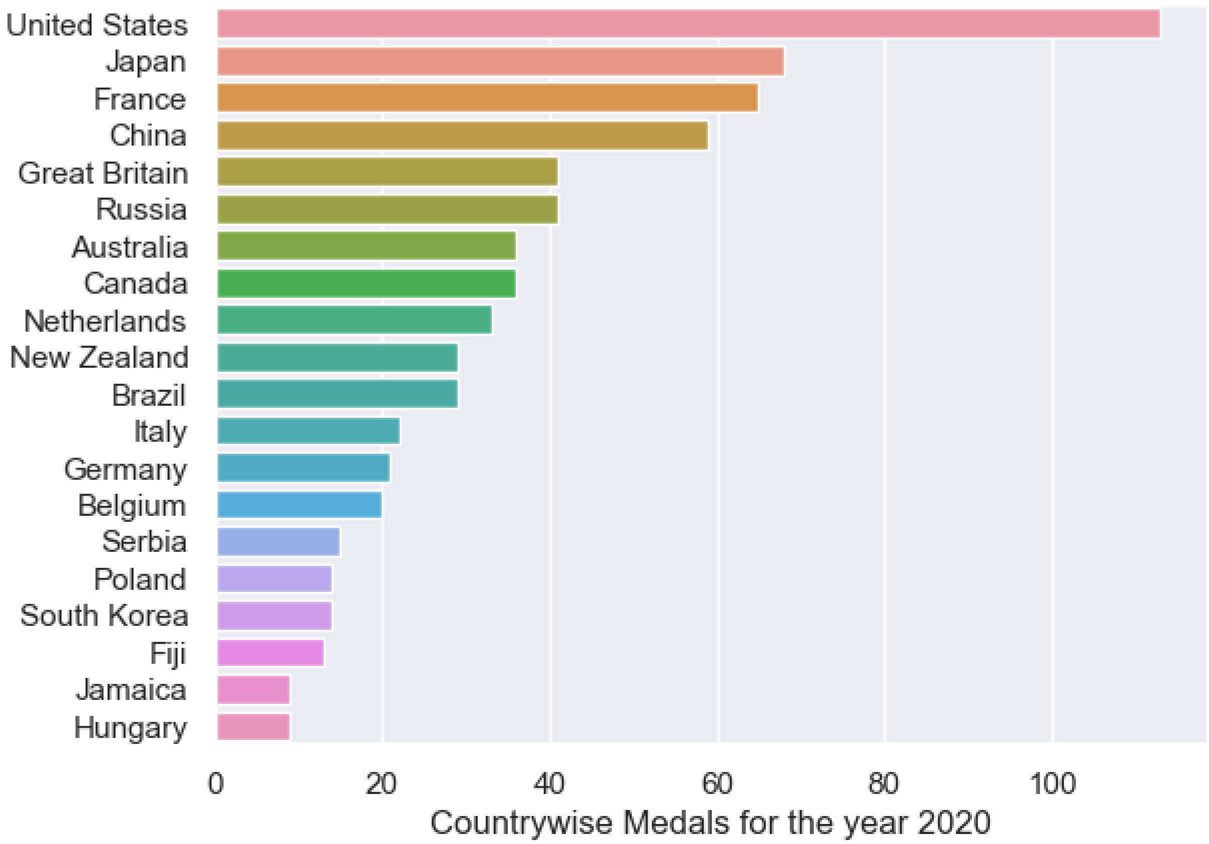
```
2020
United States    113
Japan             68
France            65
China             59
Great Britain     41
Russia            41
Australia         36
Canada            36
Netherlands       33
New Zealand       29
Name: Team, dtype: int64
```

In [41]:

```
sns.barplot(x= team_names.value_counts().head(20), y=team_names.value_counts().head(20).

plt.ylabel(None)
plt.xlabel("Countrywise Medals for the year 2020")
```

Out[41]:

```
Text(0.5, 0, 'Countrywise Medals for the year 2020')
```



CONCLUSSION

```
In [42]: 1) Total rows in  dataset = 237673
         2) Total columns in dataset = 14
         3) Null values are present in columns - Age, Medal, Region, Notes
         4) Top 10 Countries -> US, Great Britain, France, Italy, Germany, Australia, Canada, Jap
         5) Mostly Participants are of age between 20 to 30.
         6) Males = 170964 (71.9%)
         7) Females = 66709 (28.1%)
         8) Total Medals ->
                     Gold = 12259
                     Silver = 12002
                     Bronze = 12276
         9) Gold won by India -> 1928, 1932, 1936, 1948, 1952, 1956, 1964, 1980, 2008, 2020.
         10) Indiaa won highest no. of gold in 1948
         11) In year 2016 and 2020 Women participation was on peak.
         12) In 1950's and 1970's women participation had a downfall.
         13) 6 gold medals were won by athlete whose age is greater thaan 60.
         14) Women won gold medal highest in 2020 that is 400.
         15) Gold Medal per country -- USA > Russia > Germany > UK > Italy.
         !6) In last Olympics - USA won highesst number of Gold Than Japan Then
```

```
  Cell In[42], line 1
    1) Total rows in  dataset = 237673
     ^
SyntaxError: unmatched ')'
```