In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

Read data from olympics.csv

In [2]:

```python
olympics = pd.read_csv('olympics.csv', index_col=0, skiprows=1)
```

In [3]:

```python
olympics
```

Out[3]:

	№ Summer	01 !	02 !	03 !	Total	№ Winter	01 !.1	02 !.1	03 !.1	Total.1	№ Games	01 !.2	02 !.2	03 !.2	Combined total
Afghanistan (AFG)	13	0	0	2	2	0	0	0	0	0	13	0	0	2	2
Algeria (ALG)	12	5	2	8	15	3	0	0	0	0	15	5	2	8	15
Argentina (ARG)	23	18	24	28	70	18	0	0	0	0	41	18	24	28	70
Armenia (ARM)	5	1	2	9	12	6	0	0	0	0	11	1	2	9	12
Australasia (ANZ) [ANZ]	2	3	4	5	12	0	0	0	0	0	2	3	4	5	12
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
Independent Olympic Participants (IOP) [IOP]	1	0	1	2	3	0	0	0	0	0	1	0	1	2	3
Zambia (ZAM)															

Data Preparation and Cleaning

In [4]:

```
olympics.index
```

Out[4]:

```
Index(['Afghanistan (AFG)', 'Algeria (ALG)', 'Argentina (ARG)',
       'Armenia (ARM)', 'Australasia (ANZ) [ANZ]', 'Australia (AUS) [AUS] [Z]',
       'Austria (AUT)', 'Azerbaijan (AZE)', 'Bahamas (BAH)', 'Bahrain (BRN)',
       ...
       'Uzbekistan (UZB)', 'Venezuela (VEN)', 'Vietnam (VIE)',
       'Virgin Islands (ISV)', 'Yugoslavia (YUG) [YUG]',
       'Independent Olympic Participants (IOP) [IOP]', 'Zambia (ZAM) [ZAM]',
       'Zimbabwe (ZIM) [ZIM]', 'Mixed team (ZZX) [ZZX]', 'Totals'],
      dtype='object', length=147)
```

In [5]:

```
olympics.shape
```

Out[5]:

```
(147, 15)
```

In [6]:

```
olympics.head()
```

Out[6]:

	№ Summer	01 !	02 !	03 !	Total	№ Winter	01 !.1	02 !.1	03 !.1	Total.1	№ Games	01 !.2	02 !.2	03 !.2	Combined total
Afghanistan (AFG)	13	0	0	2	2	0	0	0	0	0	13	0	0	2	2
Algeria (ALG)	12	5	2	8	15	3	0	0	0	0	15	5	2	8	15
Argentina (ARG)	23	18	24	28	70	18	0	0	0	0	41	18	24	28	70
Armenia (ARM)	5	1	2	9	12	6	0	0	0	0	11	1	2	9	12
Australasia (ANZ) [ANZ]	2	3	4	5	12	0	0	0	0	0	2	3	4	5	12

In [7]:

```
olympics.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 147 entries, Afghanistan (AFG) to Totals
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   № Summer        147 non-null    int64
 1   01 !            147 non-null    int64
 2   02 !            147 non-null    int64
 3   03 !            147 non-null    int64
 4   Total           147 non-null    int64
 5   № Winter        147 non-null    int64
 6   01 !.1          147 non-null    int64
 7   02 !.1          147 non-null    int64
 8   03 !.1          147 non-null    int64
 9   Total.1         147 non-null    int64
 10  № Games         147 non-null    int64
 11  01 !.2          147 non-null    int64
 12  02 !.2          147 non-null    int64
 13  03 !.2          147 non-null    int64
 14  Combined total  147 non-null    int64
dtypes: int64(15)
memory usage: 18.4+ KB
```

In [8]:

```
olympics.isna().sum()
```

Out[8]:

```
№ Summer          0
01 !              0
02 !              0
03 !              0
Total             0
№ Winter          0
01 !.1            0
02 !.1            0
03 !.1            0
Total.1           0
№ Games           0
01 !.2            0
02 !.2            0
03 !.2            0
Combined total    0
dtype: int64
```

Modifying column headers

In [9]:

```
olympics.columns = ["No Summer", "Summer - Gold", "Summer - Silver", "Summer - Bronze", "Summer - Total'
                   "No Winter", "Winter - Gold", "Winter - Silver", "Winter - Bronze", "Winter - Total",
                   "No Total", "Total - Gold", "Total - Silver", "Total - Bronze", "Total - Medals"]
```

Modifying index values

In [10]:

```
names_ids = olympics.index.str.split('\s\(') # split the index by '('

olympics.index = names_ids.str[0] # the [0] element is the country name (new index)
```

Droping totals row.

In [11]:

```
olympics.drop("Totals", inplace=True)
```

Display list of Country names which has never won a Gold in Summer

In [12]:

```
print(list(olympics[olympics["Summer - Gold"] == 0].index))
```

```
['Afghanistan', 'Bahrain', 'Barbados', 'Bermuda', 'Bohemia', 'Botswana', 'British West Indies', 'Ivory Coas
t', 'Cyprus', 'Djibouti', 'Eritrea', 'Gabon', 'Ghana', 'Guatemala', 'Guyana', 'Haiti', 'Iceland', 'Iraq',
'Kuwait', 'Kyrgyzstan', 'Lebanon', 'Liechtenstein', 'Macedonia', 'Malaysia', 'Mauritius', 'Moldova', 'Monte
negro', 'Namibia', 'Netherlands Antilles', 'Niger', 'Paraguay', 'Philippines', 'Puerto Rico', 'Qatar', 'Sau
di Arabia', 'Senegal', 'Singapore', 'Sri Lanka', 'Sudan', 'Tajikistan', 'Tanzania', 'Togo', 'Tonga', 'Vietn
am', 'Virgin Islands', 'Independent Olympic Participants', 'Zambia']
```

Display list of Country names which has never won a Gold in either Summer or Winter Games

In [13]:

```
summer_0 = olympics["Summer - Gold"] == 0
winter_0 =olympics["Winter - Gold"] == 0
print(list(olympics[np.logical_or(summer_0, winter_0)].index))
```

```
['Afghanistan', 'Algeria', 'Argentina', 'Armenia', 'Australasia', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Barb
ados', 'Bermuda', 'Bohemia', 'Botswana', 'Brazil', 'British West Indies', 'Burundi', 'Cameroon', 'Chile',
'Colombia', 'Costa Rica', 'Ivory Coast', 'Cuba', 'Cyprus', 'Denmark', 'Djibouti', 'Dominican Republic', 'Ec
uador', 'Egypt', 'Eritrea', 'Ethiopia', 'Gabon', 'Georgia', 'Ghana', 'Greece', 'Grenada', 'Guatemala', 'Guy
ana', 'Haiti', 'Hong Kong', 'Hungary', 'Iceland', 'India', 'Indonesia', 'Iran', 'Iraq', 'Ireland', 'Israe
l', 'Jamaica', 'Kenya', 'North Korea', 'Kuwait', 'Kyrgyzstan', 'Latvia', 'Lebanon', 'Liechtenstein', 'Lithu
ania', 'Luxembourg', 'Macedonia', 'Malaysia', 'Mauritius', 'Mexico', 'Moldova', 'Mongolia', 'Montenegro',
'Morocco', 'Mozambique', 'Namibia', 'Netherlands Antilles', 'New Zealand', 'Niger', 'Nigeria', 'Pakistan',
'Panama', 'Paraguay', 'Peru', 'Philippines', 'Portugal', 'Puerto Rico', 'Qatar', 'Romania', 'Russian Empir
e', 'Saudi Arabia', 'Senegal', 'Serbia', 'Serbia and Montenegro', 'Singapore', 'South Africa', 'Sri Lanka',
'Sudan', 'Suriname', 'Syria', 'Chinese Taipei', 'Tajikistan', 'Tanzania', 'Thailand', 'Togo', 'Tonga', 'Tri
nidad and Tobago', 'Tunisia', 'Turkey', 'Uganda', 'United Arab Emirates', 'Uruguay', 'Venezuela', 'Vietna
m', 'Virgin Islands', 'Yugoslavia', 'Independent Olympic Participants', 'Zambia', 'Zimbabwe', 'Mixed team']
```

Display name of the Country which has never won a Gold in Summer Games but won in Winter Games

In [14]:

```
winter_1 = olympics["Winter - Gold"] != 0
olympics[ summer_0 & winter_1].index[0]
```

Out[14]:

```
'Liechtenstein'
```

Which country has won maximum number of medals

In [15]:

```
olympics[olympics["Total - Medals"] == max(olympics["Total - Medals"])].index[0]
```

Out[15]:

```
'United States'
```

Which country has won the most gold medals in winter games? Plot a pie chart showing Gold, Silver, Bronze totals for this country

In [16]:

```
winter_gold = olympics[olympics["Winter - Gold"] == max(olympics["Winter - Gold"])]
```

In [17]:

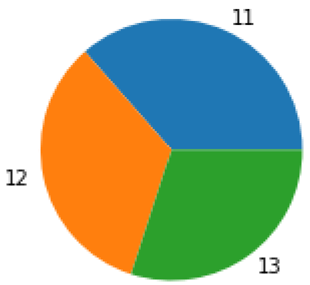
```
norway_medals = winter_gold.melt(var_name="Medal", value_name="Count")[11:14]
```

In [18]:

```
ax = norway_medals.plot(kind = "pie", y ="Count", legend = "", figsize = (20,3))

ax.set_ylabel("")
```

Out[18]:

```
Text(0, 0.5, '')
```



Which country had the biggest difference between their summer and winter gold medal counts?

In [19]:

```
difference = abs(olympics["Summer - Gold"] - olympics["Winter - Gold"])
difference.idxmax()
```

Out[19]:

```
'United States'
```

In [20]:

```
olympics[olympics.index == difference.idxmax()]
```

Out[20]:

	No Summer	Summer - Gold	Summer - Silver	Summer - Bronze	Summer - Total	No Winter	Winter - Gold	Winter - Silver	Winter - Bronze	Winter - Total	No Total	Total - Gold	Total - Silver	Total - Bronze	Tota Med
United States	26	976	757	666	2399	22	96	102	84	282	48	1072	859	750	26

Which country has the biggest difference between their summer gold medal counts and winter gold medal counts relative to their total gold medal count?

Only include countries that have won at least 1 gold in both summer and winter.

In [21]:

```
gold_winners = olympics[(olympics['Summer - Gold'] > 0) & (olympics['Winter - Gold'] > 0)].copy()
gold_winners['Gold Medal Diff'] = abs(gold_winners['Summer - Gold'] - gold_winners['Winter - Gold'])
gold_winners['Relative Gold'] = gold_winners['Gold Medal Diff']/gold_winners['Total - Medals']

gold_winners['Relative Gold'].idxmax()
```

Out[21]:

```
'China'
```

Write a function that creates a Series called "Points" which is a weighted value where each gold medal counts for 3 points, silver medals for 2 points, and bronze medals for 1 point. The function should return only the column (a Series object) which you created, with the country names as indices.

In [22]:

```
def points() :
    total_pts = olympics["Total - Gold"] * 3 + olympics["Total - Silver"] * 2 + olympics["Total - Bronze
    return total_pts.sort_values(ascending = False)
```

Identify top 5 medal winners (countries). Store their details in a new Data frame. Plot a bar chart showing their Gold, Silver, Bronze counts

In [23]:

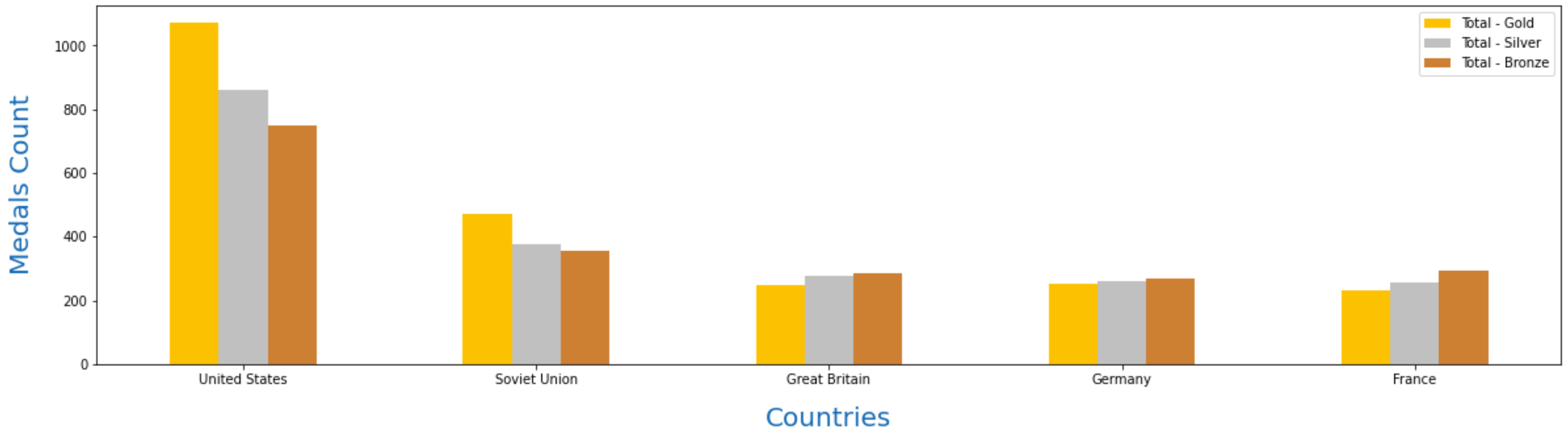
```
top_5 = olympics.loc[points().head().index]
total_med = top_5[["Total - Gold", "Total - Silver", "Total - Bronze"]]
```

In [24]:

```
colors = ["#FCC201","#C0C0C0","#cd7f32"]

ax = total_med.plot(kind = "bar", color = colors, figsize = (20, 5), rot = 0)

ax.set_xlabel("Countries", labelpad = 15,
              fontdict = {'fontsize':20, 'fontweight':2, 'color':'#1369B5'})

ax.set_ylabel("Medals Count", labelpad = 15,
              fontdict = {'fontsize':20, 'fontweight':2, 'color':'#1369B5'})
```

Out[24]:

```
Text(0, 0.5, 'Medals Count')
```


Total - Gold
Total - Silver
Total - Bronze
Medals Count
United States
Soviet Union
Great Britain
Germany
France
Countries