In [76]:

```
import pandas as pd
import os
print(os.getcwd())
```

```
C:\Users\spark\Desktop\Indesign Print
```

In [77]:

```
os.chdir('C:\\Users\\spark\\Desktop\\Data Science Projects\\Squid Game Analysis') #This is changing the working dire
country = pd.read_csv('Country.csv')
```

In [78]:

```
#Requires data cleansing since there are multiple names for each country
```

In [79]:

```
country.head()
```

Out[79]:

	user_location
0	Any pronouns
1	France
2	United Kingdom
3	Fujoshi ?셋/ Thai BL-obsessed/Always distracted...
4	South Africa

In [80]:

```
COUNTRYTOP10=country.value_counts(ascending=False)
```

In [81]:

```
COUNTRYTOP10.head(100)
COUNTRYTOP10[0:50]
```

Out[81]:

```
user_location
Los Angeles, CA                 853
London, England                 677
India                           644
United States                   638
USA                             610
London                          531
United Kingdom                  499
Twitter for Android             462
New York, NY                    369
Canada                          348
Dubai, United Arab Emirates     330
England, United Kingdom         321
Atlanta, GA                     282
Mumbai, India                   256
California, USA                 252
Twitter Web App                 250
New York, USA                   246
Chicago, IL                     238
UK                              232
Brooklyn, NY                    220
Australia                       217
Washington, DC                  204
New York                        202
Lagos, Nigeria                  176
Toronto, Ontario                173
Singapore                       171
New Delhi, India                169
San Francisco, CA               163
Los Angeles                     160
Earth                           157
Houston, TX                     156
Dhaka, Bangladesh               149
Malaysia                        145
Seattle, WA                     144
Lyon, France                    143
she/her                         143
Florida, USA                    143
```

```
54 Old Ife Rd, Ibadan          139
NYC                            135
South Africa                   134
Manchester, England            129
Worldwide                      129
Toronto                        127
Metaverse                      127
Philadelphia, PA               124
Nigeria                        123
London, UK                     122
Boston, MA                     122
Dallas, TX                     121
Twitter for iPhone             121
dtype: int64
```

In [82]:

```
#Lookign at the top value counts, we make a decision to choose top 8 counts which are:
#United States, United Kingdom, India, Canada, United Arab Emirates, Singapore, France, and South Korea
#Now, we need a dataframe that can be used to go through data cleansing.
```

In [83]:

```
top100dataframe=COUNTRYTOP10.to_frame('count')
```

In [84]:

```
top100dataframe
```

Out[84]:

```
                                                          count
                                         user_location
                                       Los Angeles, CA      853
                                       London, England      677
                                                 India      644
                                         United States      638
                                                   USA      610
                                                   ...      ...
                                  Anyway the wind Blows       1
                                      Any where i want        1
                           Maryland / Washington, DC          1
                                          Any trash can       1
 \nit fang sun kit\nrust bone bur\nbib tooth vamp\n+ He She    1
```

19672 rows × 1 columns

In [85]:

```
top100dataframe=top100dataframe.reset_index()
#This is to allow str.contains operations on column user_location
```

In [86]:

```
top100dataframe
```

Out[86]:

	user_location	count
0	Los Angeles, CA	853
1	London, England	677
2	India	644
3	United States	638
4	USA	610
...	...	...
19667	Anyway the wind Blows	1
19668	Any where i want	1
19669	Maryland / Washington, DC	1
19670	Any trash can	1
19671	\nit fang sun kit\nrust bone bur\nbib tooth va...	1

19672 rows × 2 columns

In [87]:

```
#First, we need to find the counts for the United States when there are both
# names of countries and states to count how many are in total.
#So we use str.contains to select only ones relevent to the name of the countries
#and states to add these counts for a final sum counts.
```

In [88]:

```
findUSA = top100dataframe
```

In [89]:

```
findUSA
```

Out[89]:

	user_location	count
0	Los Angeles, CA	853
1	London, England	677
2	India	644
3	United States	638
4	USA	610
...	...	...
19667	Anyway the wind Blows	1
19668	Any where i want	1
19669	Maryland / Washington, DC	1
19670	Any trash can	1
19671	\nit fang sun kit\nrust bone bur\nbib tooth va...	1

19672 rows × 2 columns

In [90]:

```
findUSA =findUSA.loc[findUSA['user_location'].str.contains(
    "America|United States|USA|U.S.|Alabama|Alaska|Arizona|California|CA|Colorado|Arkansas|California|Colorado|Conne
```

In [91]:
```
findUSA
```

Out[91]:

	user_location	count
0	Los Angeles, CA	853
3	United States	638
4	USA	610
8	New York, NY	369
14	California, USA	252
...	...	...
19647	Maryland crab	1
19651	Anywhere USA	1
19661	Marshalltown, Iowa	1
19663	Martinez, CA	1
19669	Maryland / Washington, DC	1

1358 rows × 2 columns

In [92]:
```
totalUSA= pd.Series(findUSA['count']).sum()
```

In [93]:
```
totalUSA
```

Out[93]:
```
8428
```

In [94]:
```
#Next, find total counts for United Kingdom
```

In [95]:
```
findunitedkingdom = top100dataframe
```

In [96]:
```
#using python string regex.
```

In [97]:
```
findunitedkingdom2= findunitedkingdom.loc[findunitedkingdom['user_location'].str.contains(
    "Lond|England|Wales|Scotland|Northern Ireland ", case=True)]
```

In [98]:
```
findunitedkingdom2.index
```

Out[98]:
```
Int64Index([    1,     5,    11,    40,    46,    68,    69,    71,    82,
               93,
            ...
            19370, 19446, 19457, 19476, 19540, 19543, 19544, 19613, 19618,
            19620],
           dtype='int64', length=855)
```

In [99]:
```
findUSA.index
```

Out[99]:
```
Int64Index([    0,     3,     4,     8,    14,    16,    21,    22,    27,
               36,
            ...
            19605, 19612, 19614, 19631, 19645, 19647, 19651, 19661, 19663,
            19669],
           dtype='int64', length=1358)
```

In [100]:
```
#I found some erors for finding UK so fixing these ones as well
```

In [101]:
```
findunitedkingdom3= findunitedkingdom2.loc[findunitedkingdom2['user_location'].str.contains("Singapore|Sydney", case
```

In [102]:

```
findunitedkingdom2.drop(69)
findunitedkingdom2.drop(4901)
findunitedkingdom2.drop(8644)
findunitedkingdom2.drop(17946)
findunitedkingdom2.drop(17991)
```

Out[102]:

	user_location	count
1	London, England	677
5	London	531
11	England, United Kingdom	321
40	Manchester, England	129
46	London, UK	122
...	...	...
19543	Merthyr Tydfil, South Wales	1
19544	Merton, London	1
19613	Mansfield Woodhouse England	1
19618	Marlow, England	1
19620	Marylebone, London ?늓?늕	1

854 rows × 2 columns

In [103]:

```
findunitedkingdom2
```

Out[103]:

	user_location	count
1	London, England	677
5	London	531
11	England, United Kingdom	321
40	Manchester, England	129
46	London, UK	122
...	...	...
19543	Merthyr Tydfil, South Wales	1
19544	Merton, London	1
19613	Mansfield Woodhouse England	1
19618	Marlow, England	1
19620	Marylebone, London ?늕?늖	1

855 rows × 2 columns

In [104]:

```
UK_SUM= pd.Series(findunitedkingdom2['count']).sum()
```

In [105]:

```
UK_SUM #Final total number of UK
```

Out[105]:

```
4449
```

In [106]:

```
#Next find for India total counts
findIndia = top100dataframe =COUNTRYTOP10.to_frame('count')
```

In [107]:

```
findIndia = findIndia.reset_index()
```

In [108]:
```
findIndia2= findIndia.loc[findIndia['user_location'].str.contains("India|inda", case=True)]
```

In [109]:
```
TotalIndia= pd.Series(findIndia2['count']).sum()
```

In [110]:
```
findIndia.reset_index()
```

Out[110]:

	index	user_location	count
0	0	Los Angeles, CA	853
1	1	London, England	677
2	2	India	644
3	3	United States	638
4	4	USA	610
...	...	...	...
19667	19667	Anyway the wind Blows	1
19668	19668	Any where i want	1
19669	19669	Maryland / Washington, DC	1
19670	19670	Any trash can	1
19671	19671	\nit fang sun kit\nrust bone bur\nbib tooth va...	1

19672 rows × 3 columns

In [111]:
```
TotalIndia
```

Out[111]:
```
2391
```

In [112]:
```
#Next, find one for South Korea.
```

In [113]:
```
findrepublicofKorea = top100dataframe
```

In [114]:

```
findrepublicofKorea=findrepublicofKorea.reset_index()
```

In [115]:

```
findrepublicofKorea2= findrepublicofKorea.loc[findrepublicofKorea['user_location'].str.contains(
    "Republic of Korea|republic of Korea", case=True)]
```

In [116]:

```
TotalSouthKorea= pd.Series(findrepublicofKorea2['count']).sum()
```

In [117]:

```
TotalSouthKorea
```

Out[117]:

```
81
```

In [118]:

```
findSaudiArabia= top100dataframe
```

In [119]:

```
findSaudiArabia.reset_index()
```

Out[119]:

	user_location	count
0	Los Angeles, CA	853
1	London, England	677
2	India	644
3	United States	638
4	USA	610
...	...	...
19667	Anyway the wind Blows	1
19668	Any where i want	1
19669	Maryland / Washington, DC	1
19670	Any trash can	1
19671	\nit fang sun kit\nrust bone bur\nbib tooth va...	1

19672 rows × 2 columns

In [120]:

```
findSaudiArabia = findSaudiArabia.reset_index()
```

In [121]:

```
findSaudiArabia= findSaudiArabia.loc[findSaudiArabia['user_location'].str.contains(
    "Saudi|Arabia", case=True)]
```

In [122]:

```
TotalSaudiArabia= pd.Series(findSaudiArabia['count']).sum()
```

In [123]:

```
TotalSaudiArabia
```

Out[123]:

```
55
```

In [124]:

```
#Find one for Canada
```

In [125]:

```
findCanada=top100dataframe.reset_index()
```

In [126]:

```
findCanada= findCanada.loc[findCanada['user_location'].str.contains(
    "Canada|canada", case=True)]
```

In [127]:

```
TotalCanada= pd.Series(findCanada['count']).sum()
```

In [128]:

```
TotalCanada
```

Out[128]:

```
789
```

In [129]:

```
#Find one for France
```

In [130]:

```
findFrance= top100dataframe.reset_index()
```

In [131]:

```
findFrance= findFrance.loc[findFrance['user_location'].str.contains(
    "France|france", case=True)]
```

In [132]:
```python
TotalFrance= pd.Series(findFrance['count']).sum()
```

In [133]:
```python
TotalFrance
```

Out[133]:
```
366
```

In [134]:
```python
#Using these total counts of each countries, create a seaborn visual represntation.

```

In [135]:
```python
#Print out using dictionary syntax.
Countriessv = {'Country Location':[
    'United States','United Kingdom','India','France', 'Canada','South Korea','Saudi Arabia'],
               'Sum Count':[8428,4449,2391,366,789,81,55]} #For creating a dataframe
```

In [136]:
```python
Countriessv= pd.DataFrame(Countriessv)
CountriessortedSumCount = Countriessv.sort_values(["Sum Count"], ascending = False)
```

In [137]:
```python
os.chdir('C:\\Users\\spark\\Desktop\\Indesign Print')
```

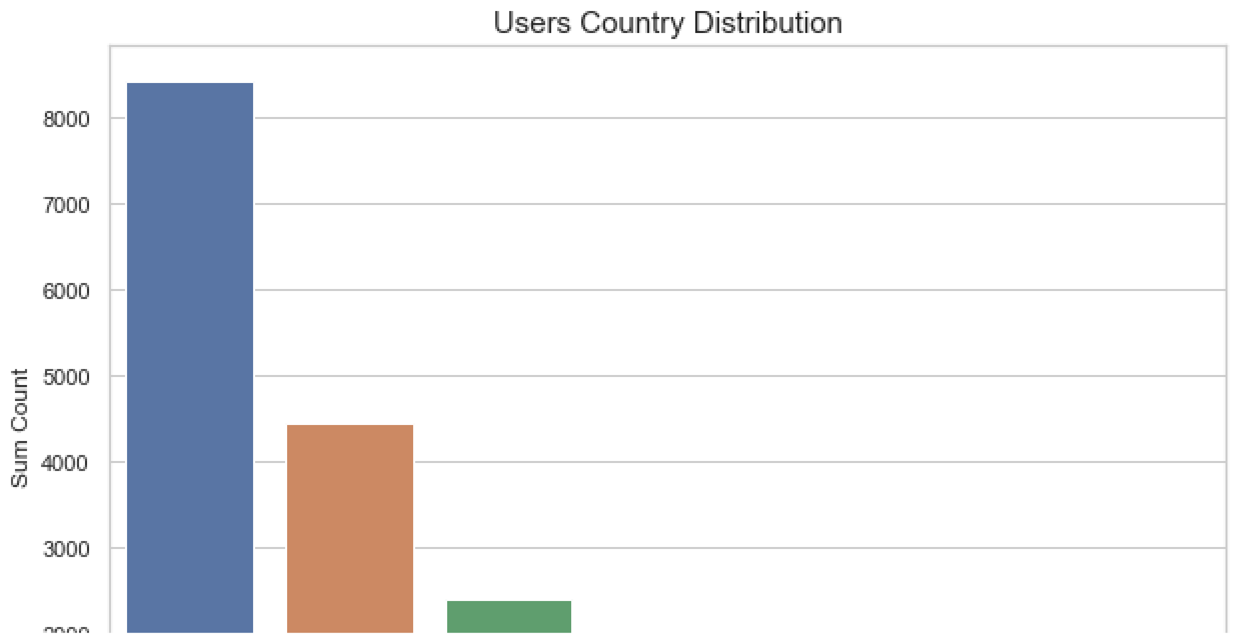
In [138]:
```python
CountriessortedSumCount
```

Out[138]:
```
   Country Location  Sum Count
0     United States       8428
1    United Kingdom       4449
2             India       2391
4            Canada        789
3            France        366
5       South Korea         81
6      Saudi Arabia         55
```

In [139]:

```python
import seaborn as sns
import matplotlib.pyplot as plt
plt.figure(figsize = (10,7))
sns.set_theme(style="whitegrid")
ax = sns.barplot(x="Country Location", y="Sum Count", data=CountriessortedSumCount)
plt.title("Users Country Distribution",fontsize = 15)
print(os.getcwd())
plt.savefig('test13.png',dpi=200)
```

```
C:\Users\Spark\Desktop\Indesign Print
```


Users Country Distribution
Sum Count