

```
In [1]:  import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.graph_objects as go
import warnings
warnings.filterwarnings('ignore')
data = pd.read_csv('C://Users//khage//OneDrive//Documents//Data Visualization
data
```

Out[1]:

	Rank	Name	Total Net Worth	\$ Last Change	\$ YTD Change	Country	Industry	Unnamed: 7	Unnamed: 8
0	1.0	Jeff Bezos	\$188B	+\$1.68B	-\$2.31B	United States	Technology	NaN	NaN
1	2.0	Elon Musk	\$170B	-\$2.89B	+\$773M	United States	Technology	NaN	NaN
2	3.0	Bernard Arnault	\$155B	+\$892M	+\$40.9B	France	Consumer	NaN	NaN
3	4.0	Bill Gates	\$144B	-\$1.32B	+\$12.2B	United States	Technology	NaN	NaN
4	5.0	Mark Zuckerberg	\$114B	+\$203M	+\$10.9B	United States	Technology	NaN	NaN
...
498	500.0	Odd Reitan	\$5.72B	-\$19.9M	+\$669M	Norway	Food & Beverage	NaN	NaN
499	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
500	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
501	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
502	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

503 rows × 11 columns



```
In [2]: data.head()
```

Out[2]:

	Rank	Name	Total Net Worth	\$ Last Change	\$ YTD Change	Country	Industry	Unnamed: 7	Unnamed: 8	Un
0	1.0	Jeff Bezos	\$188B	+\$1.68B	-\$2.31B	United States	Technology	NaN	NaN	
1	2.0	Elon Musk	\$170B	-\$2.89B	+\$773M	United States	Technology	NaN	NaN	
2	3.0	Bernard Arnault	\$155B	+\$892M	+\$40.9B	France	Consumer	NaN	NaN	
3	4.0	Bill Gates	\$144B	-\$1.32B	+\$12.2B	United States	Technology	NaN	NaN	
4	5.0	Mark Zuckerberg	\$114B	+\$203M	+\$10.9B	United States	Technology	NaN	NaN	



```
In [3]: #Getting some information about dataset
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503 entries, 0 to 502
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Rank                   499 non-null   float64
1   Name                   499 non-null   object
2   Total Net Worth        499 non-null   object
3   $ Last Change          499 non-null   object
4   $ YTD Change           499 non-null   object
5   Country                499 non-null   object
6   Industry               499 non-null   object
7   Unnamed: 7             0 non-null     float64
8   Unnamed: 8             0 non-null     float64
9   Unnamed: 9             0 non-null     float64
10  Unnamed: 10            0 non-null     float64
dtypes: float64(5), object(6)
memory usage: 43.4+ KB
```

```
In [4]: data.drop(["Unnamed: 7", "Unnamed: 8", "Unnamed: 9", "Unnamed: 10"], inplace=True)
data.drop([499, 500, 501, 502], inplace=True, axis=0) #Drop null rows at the end.
data.fillna(0, inplace=True, axis=1) #Replace null values with 0.
data=data.replace(['$', " "], '', regex=True) #Remove special characters and un
data["Total Net Worth"]=data["Total Net Worth"].replace("B", "", regex=True) #R
data["Total Net Worth"]=pd.to_numeric(data["Total Net Worth"], errors="coerce"
display(data.head())
```

	Rank	Name	Total Net Worth	\$ Last Change	\$ YTD Change	Country	Industry
0	1.0	JeffBezos	188.0	+1.68B	-2.31B	UnitedStates	Technology
1	2.0	ElonMusk	170.0	-2.89B	+773M	UnitedStates	Technology
2	3.0	BernardArnault	155.0	+892M	+40.9B	France	Consumer
3	4.0	BillGates	144.0	-1.32B	+12.2B	UnitedStates	Technology
4	5.0	MarkZuckerberg	114.0	+203M	+10.9B	UnitedStates	Technology

```
In [14]: #Changing of Data types

def change(data, col):
    data[col] = data[col].replace({'$': ''}, regex = True)

def clear(x):
    if 'M' in x:
        return float(x.rstrip('M')) / 1000
    elif 'k' in x:
        return float(x.rstrip('k')) / 10 ** 6
    else:
        return float(x.rstrip('B'))

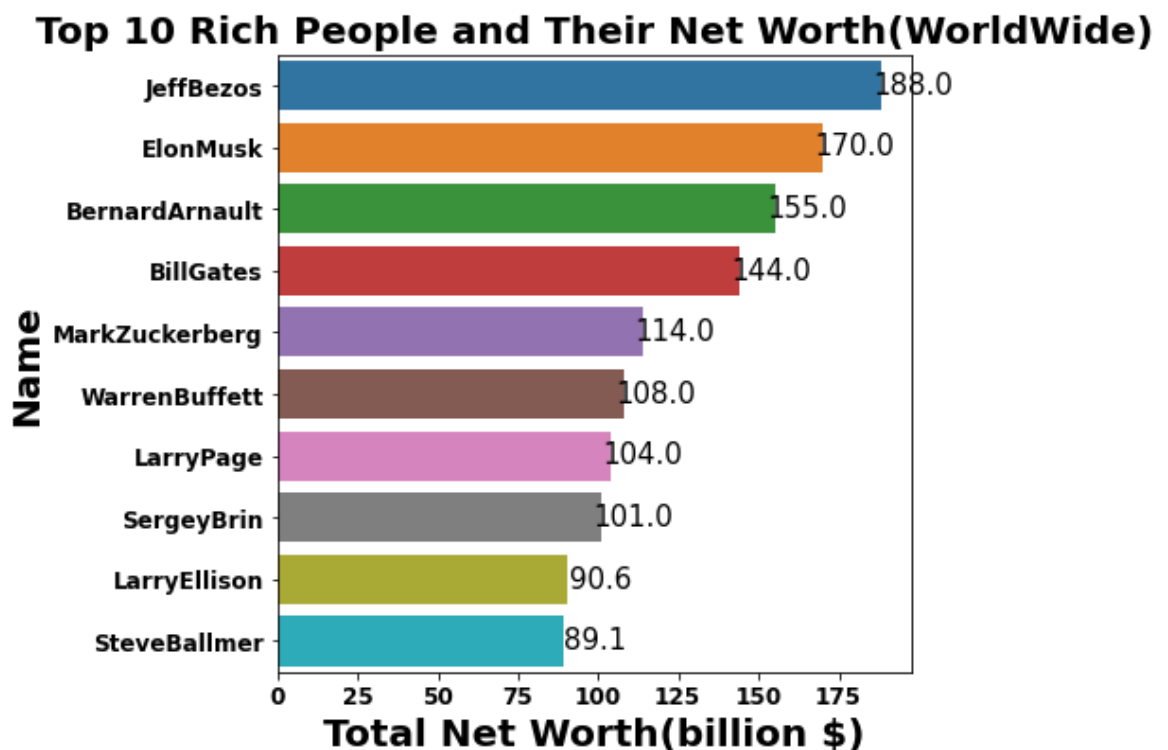
change(data, '$ Last Change')
data['$ Last Change'] = data['$ Last Change'].apply(clear)
```

```
In [5]: fig= plt.figure(figsize=(6,6))
top10_person=data[:10]
plots=sns.barplot(x='Total Net Worth',y='Name',data=top10_person)

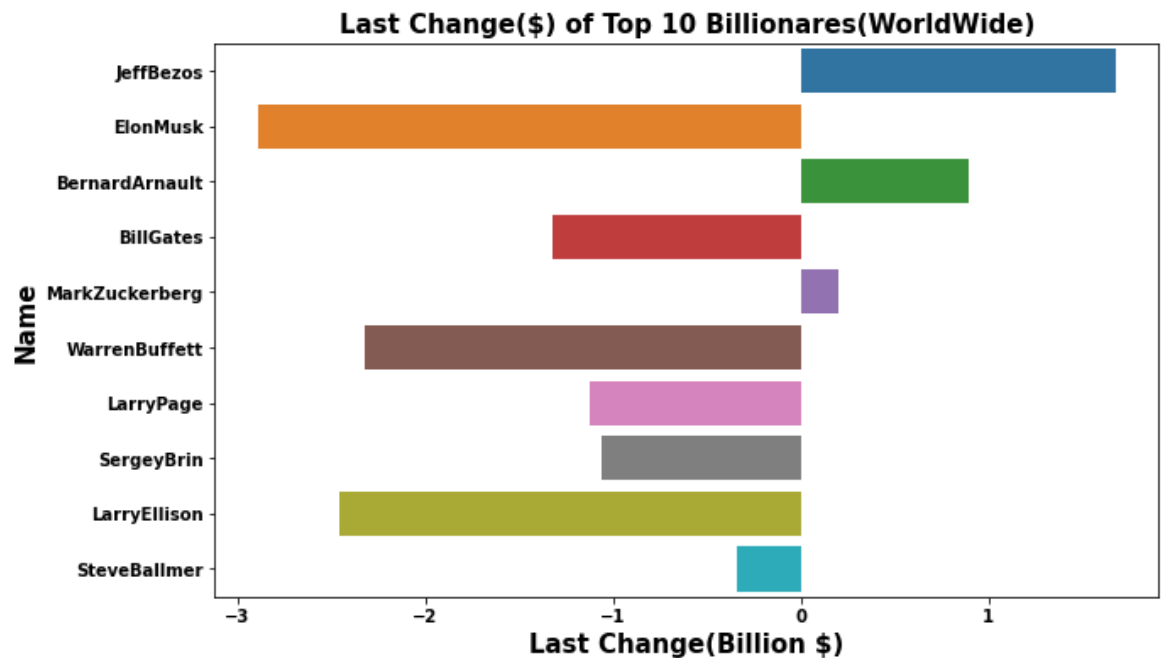
plt.title("Top 10 Rich People and Their Net Worth(WorldWide)",fontsize=20,for
plt.ylabel("Name",fontsize=20,fontweight='bold')
plt.xlabel("Total Net Worth(billion $)",fontsize=20,fontweight='bold')

for bar in plots.patches:
    plots.annotate(bar.get_width(),(bar.get_x()+bar.get_width()+7,
                                   bar.get_y()+0.7),size=15,ha='center',
                   va='center',xytext=(5,10),textcoords='offset points')

plt.xticks(fontweight='bold',fontsize=12)
plt.yticks(fontweight='bold',fontsize=12)
plt.show()
```



```
In [15]: fig=plt.figure(figsize=(10,6))
sns.barplot(x='$ Last Change',y='Name',data=data[:10],orient='h')
plt.xticks(fontweight='bold')
plt.yticks(fontweight='bold')
plt.xlabel("Last Change(Billion $)",fontweight='bold',fontsize=15)
plt.ylabel("Name",fontweight='bold',fontsize=15)
plt.title("Last Change($) of Top 10 Billionares(WorldWide)",fontweight='bold')
plt.show()
```



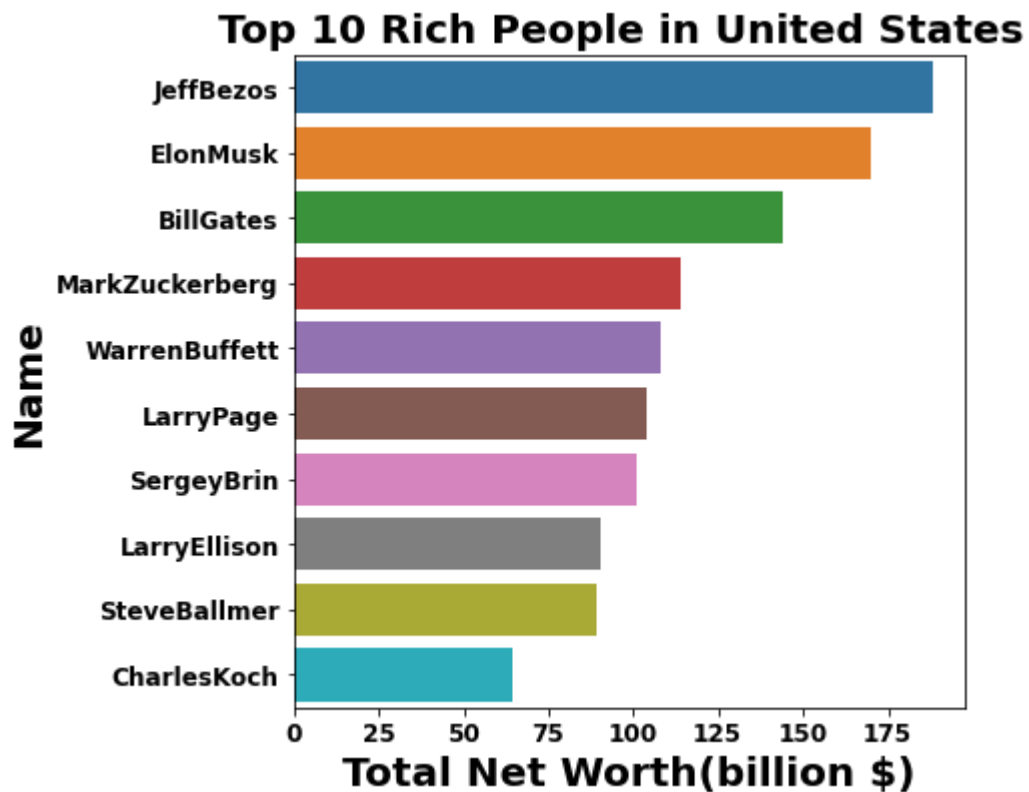
Here Most of the People lost their Worth by 1B or 2B dollar.These Values are not constant though. It changes Everyday.

```
In [7]: #Making Dataset for Country United States Of America.
US=data[data['Country']=='UnitedStates']
```

```
In [8]: fig= plt.figure(figsize=(6,6))
sns.barplot(x='Total Net Worth',y='Name',data=US[:10])

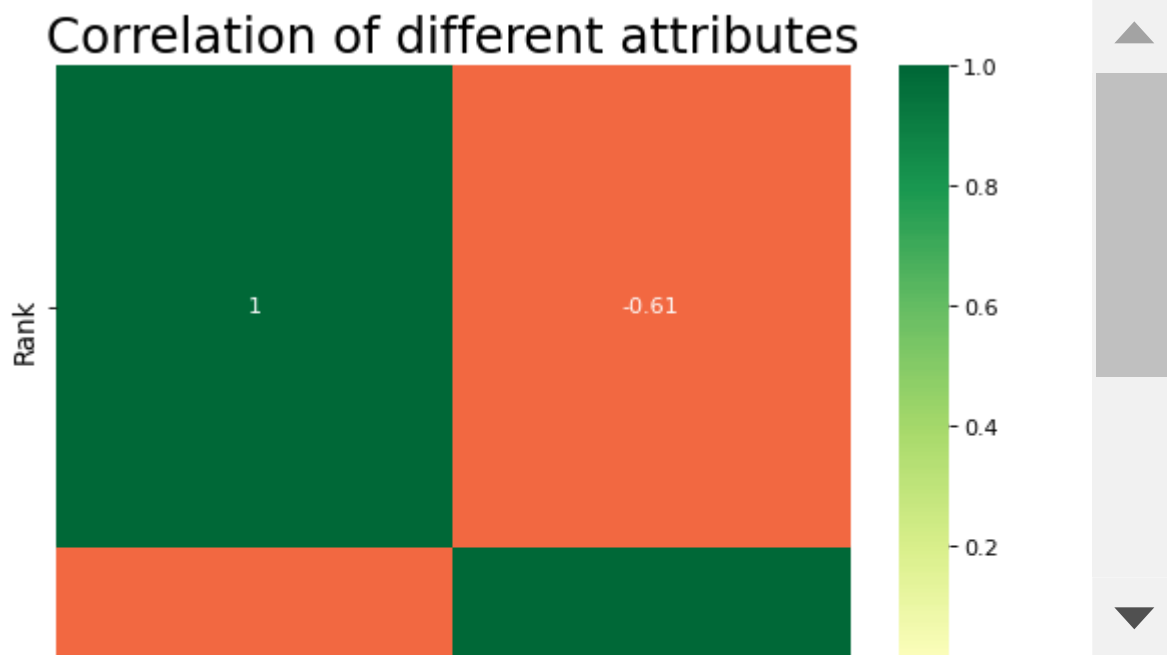
plt.title("Top 10 Rich People in United States ",fontsize=20,fontweight='bold')
plt.ylabel("Name",fontsize=20,fontweight='bold')
plt.xlabel("Total Net Worth(billion $)",fontsize=20,fontweight='bold')

plt.xticks(fontweight='bold',fontsize=12)
plt.yticks(fontweight='bold',fontsize=12)
plt.show()
```



```
In [9]: # Plot
plt.figure(figsize=(8,8), dpi= 80)
sns.heatmap(data.corr(), xticklabels=data.corr().columns, yticklabels=data.co

# Decorations
plt.title('Correlation of different attributes', fontsize=22)
plt.xticks(fontsize=12)
plt.yticks(fontsize=12)
plt.show()
```

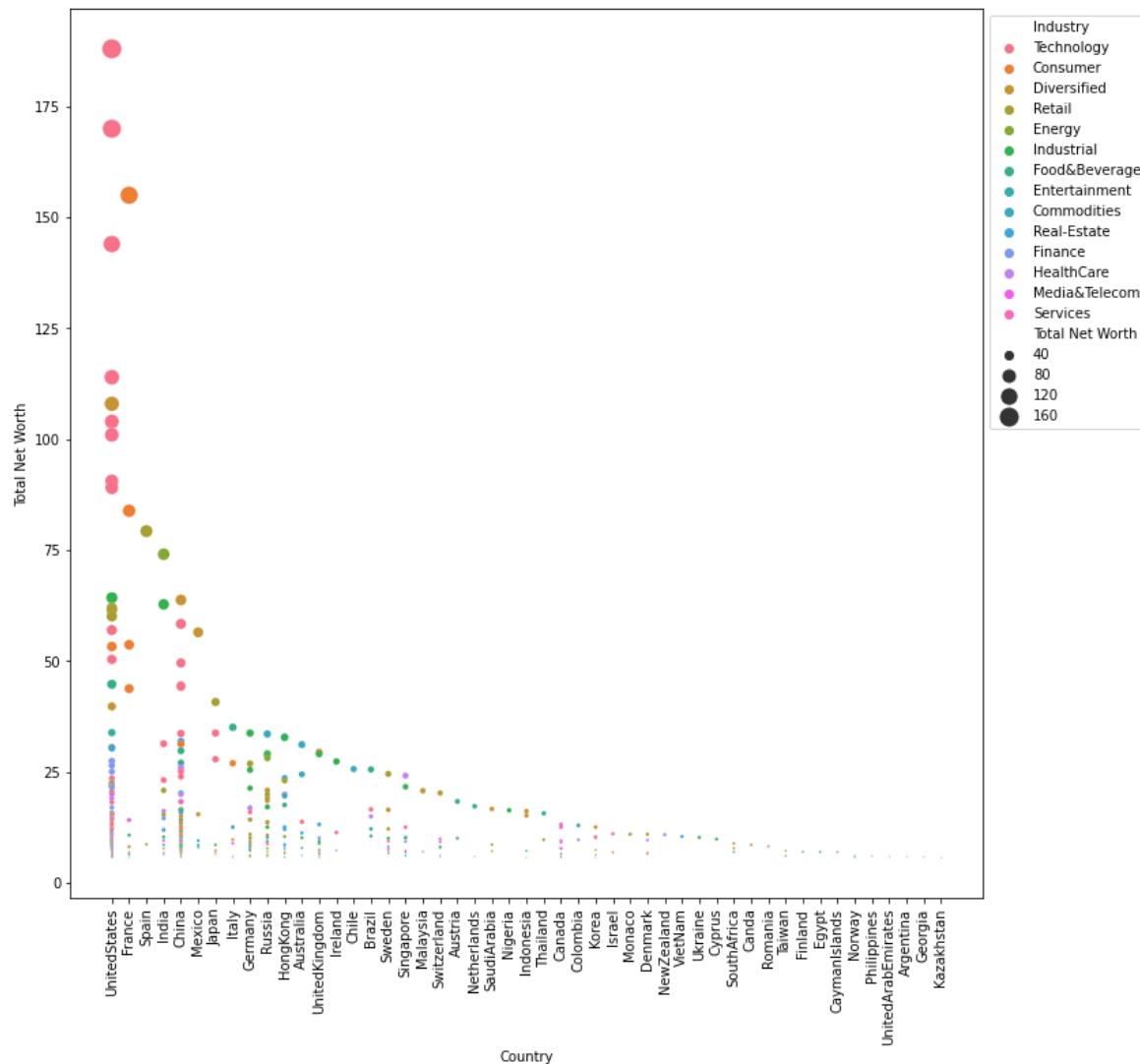


This heatmap is the correlation between rank and total net worth which shows there is negative correlation between total net worth and rank meaning, low rank represents high net worth and vice versa.

```
In [10]: plt.figure(figsize=(12, 12))
plt.legend(fontsize=10)
plt.tick_params(labelsize=10)
ax=sns.scatterplot(x=data['Country'],y=data['Total Net Worth'],hue=data['Industry'])
plt.xticks(rotation=90)
ax.legend(loc='upper left',bbox_to_anchor=(1,1))
```

No handles with labels found to put in legend.

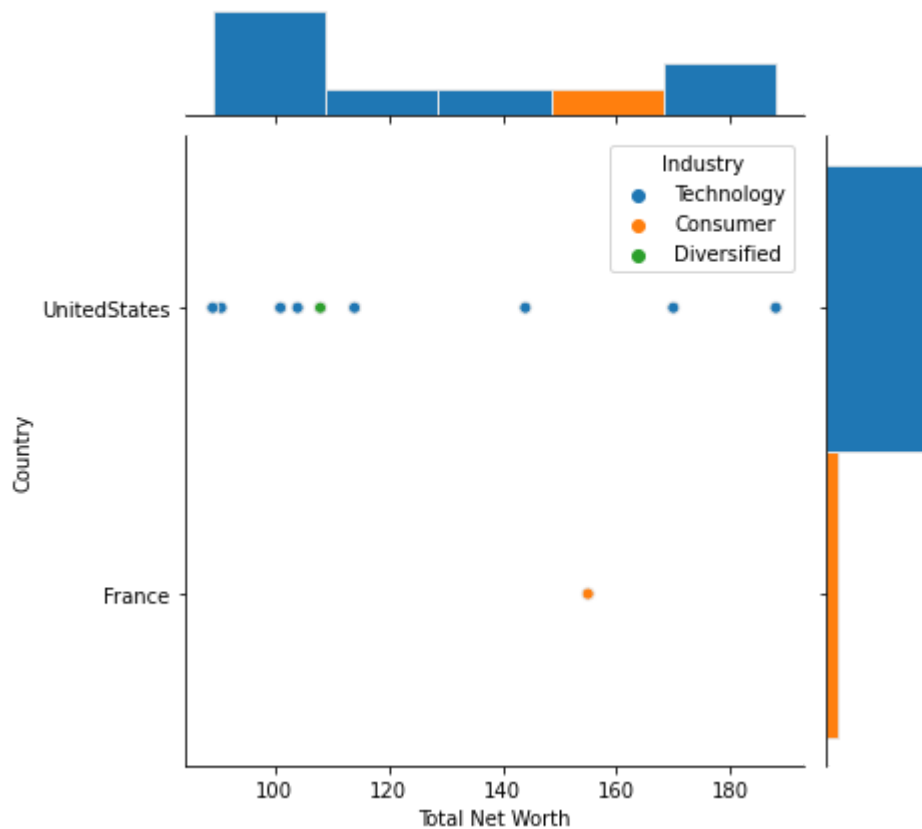
Out[10]: <matplotlib.legend.Legend at 0x288cefa52e0>



From the above scatter plot, we can see that, in the US, a lot of rich people are in Technology which is different from France and Spain.

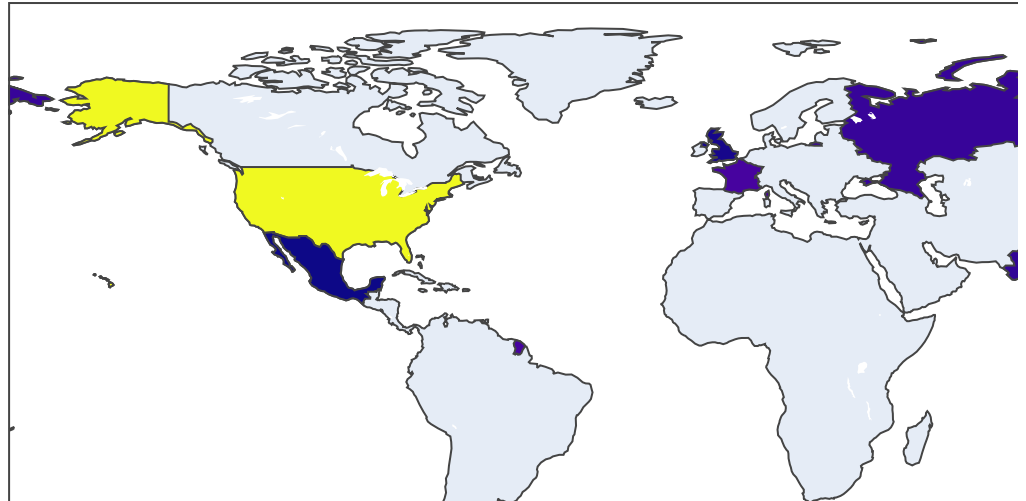
```
In [11]: g = sns.JointGrid(data=data[:10], x="Total Net Worth", y="Country", hue='Industry')
g.plot(sns.scatterplot, sns.histplot, alpha=1, edgecolor=".9", linewidth=.9)
```

```
Out[11]: <seaborn.axisgrid.JointGrid at 0x288cf086730>
```



From above jointplot, we can see that among top 10 richest people, 9 richest people are from United States and one from France. Technology is the main industry that most of the people are rich.

```
In [12]: ▶ import plotly_express as px
data1=pd.DataFrame(data.groupby('Country')['Total Net Worth'].sum().sort_valu
data1['Country'] = data1.index
data1['Country codes']=['USA','CHN','FRA','RUS','GER','IND','HKG','GBR','JPN'
fig = px.choropleth(data1,locations='Country codes',color='Total Net Worth',s
fig.show()
```



```
In [ ]: ▶
```