

```
In [52]: 1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 import plotly.express as px
```

```
In [53]: 1 df= pd.read_csv("Global internet users.csv")
2 df.head()
```

Out[53]:

	Unnamed: 0	Entity	Code	Year	Cellular Subscription	Internet Users(%)	No. of Internet Users	Broadband Subscription
0	0	Afghanistan	AFG	1980	0.0	0.0	0	0.0
1	1	Afghanistan	AFG	1981	0.0	0.0	0	0.0
2	2	Afghanistan	AFG	1982	0.0	0.0	0	0.0
3	3	Afghanistan	AFG	1983	0.0	0.0	0	0.0
4	4	Afghanistan	AFG	1984	0.0	0.0	0	0.0

```
In [54]: 1 df['Entity'].unique()
```

Out[54]: array(['Afghanistan', 'Albania', 'Algeria', 'American Samoa', 'Andorra', 'Angola', 'Antigua and Barbuda', 'Argentina', 'Armenia', 'Aruba', 'Australia', 'Austria', 'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh', 'Barbados', 'Belarus', 'Belgium', 'Belize', 'Benin', 'Bermuda', 'Bhutan', 'Bolivia', 'Bosnia and Herzegovina', 'Botswana', 'Brazil', 'British Virgin Islands', 'Brunei', 'Bulgaria', 'Burkina Faso', 'Burundi', 'Cambodia', 'Cameroon', 'Canada', 'Cape Verde', 'Cayman Islands', 'Central African Republic', 'Chad', 'Chile', 'China', 'Colombia', 'Comoros', 'Congo', 'Costa Rica', 'Cote d'Ivoire', 'Croatia', 'Cuba', 'Curacao', 'Cyprus', 'Czechia', 'Democratic Republic of Congo', 'Denmark', 'Djibouti', 'Dominica', 'Dominican Republic', 'East Asia and Pacific', 'Ecuador', 'Egypt', 'El Salvador', 'Equatorial Guinea', 'Eritrea', 'Estonia', 'Eswatini', 'Ethiopia', 'Europe and Central Asia', 'European Union', 'Faeroe Islands', 'Fiji', 'Finland', 'France', 'French Polynesia', 'Gabon', 'Gambia', 'Georgia', 'Germany', 'Ghana', 'Gibraltar', 'Greece', 'Greenland', 'Grenada', 'Guam', 'Guatemala', 'Guinea', 'Guinea-Bissau', 'Guyana', 'Haiti', 'Honduras', 'Hungary', 'Iceland', 'India', 'Indonesia', 'Israel', 'Italy', 'Jamaica', 'Japan', 'Jordan', 'Kazakhstan', 'Kenya', 'Kiribati', 'Kuwait', 'Kyrgyzstan', 'Laos', 'Latvia', 'Lebanon', 'Lesotho', 'Lithuania', 'Luxembourg', 'Madagascar', 'Malawi', 'Malaysia', 'Maldives', 'Mali', 'Malta', 'Mauritania', 'Mauritius', 'Mexico', 'Micronesia', 'Moldova', 'Monaco', 'Mongolia', 'Montenegro', 'Morocco', 'Mozambique', 'Myanmar', 'Namibia', 'Nauru', 'Nepal', 'Netherlands', 'New Zealand', 'Nicaragua', 'Niger', 'Nigeria', 'North Macedonia', 'Norway', 'Oman', 'Pakistan', 'Palau', 'Panama', 'Papua New Guinea', 'Paraguay', 'Peru', 'Philippines', 'Poland', 'Portugal', 'Puerto Rico', 'Qatar', 'Romania', 'Russia', 'Rwanda', 'Saint Kitts and Nevis', 'Saint Lucia', 'Saint Vincent and the Grenadines', 'Samoa', 'San Marino', 'Saudi Arabia', 'Senegal', 'Serbia', 'Sierra Leone', 'Singapore', 'Slovakia', 'Slovenia', 'Solomon Islands', 'South Africa', 'South Korea', 'Spain', 'Sri Lanka', 'Sudan', 'Suriname', 'Swaziland', 'Sweden', 'Switzerland', 'Taiwan', 'Tajikistan', 'Tanzania', 'Thailand', 'Timor-Leste', 'Togo', 'Tonga', 'Trinidad and Tobago', 'Tunisia', 'Turkey', 'Turkmenistan', 'Uganda', 'Ukraine', 'United Arab Emirates', 'United Kingdom', 'United States', 'Uruguay', 'Uzbekistan', 'Vanuatu', 'Venezuela', 'Vietnam', 'Yemen', 'Zambia', 'Zimbabwe'])

In [55]: 1 df.tail()

Out[55]:

	Unnamed: 0	Entity	Code	Year	Cellular Subscription	Internet Users(%)	No. of Internet Users	Broadband Subscription
8862	8862	Zimbabwe	ZWE	2016	91.793457	23.119989	3341464	1.217633
8863	8863	Zimbabwe	ZWE	2017	98.985077	24.400000	3599269	1.315694
8864	8864	Zimbabwe	ZWE	2018	89.404869	25.000000	3763048	1.406322
8865	8865	Zimbabwe	ZWE	2019	90.102287	25.100000	3854006	1.395818
8866	8866	Zimbabwe	ZWE	2020	88.755806	29.299999	4591211	1.368916

In [56]: 1 df.shape

Out[56]: (8867, 8)

In [57]: 1 df.isna().sum()

Out[57]: Unnamed: 0 0  
Entity 0  
Code 0  
Year 0  
Cellular Subscription 0  
Internet Users(%) 0  
No. of Internet Users 0  
Broadband Subscription 0  
dtype: int64

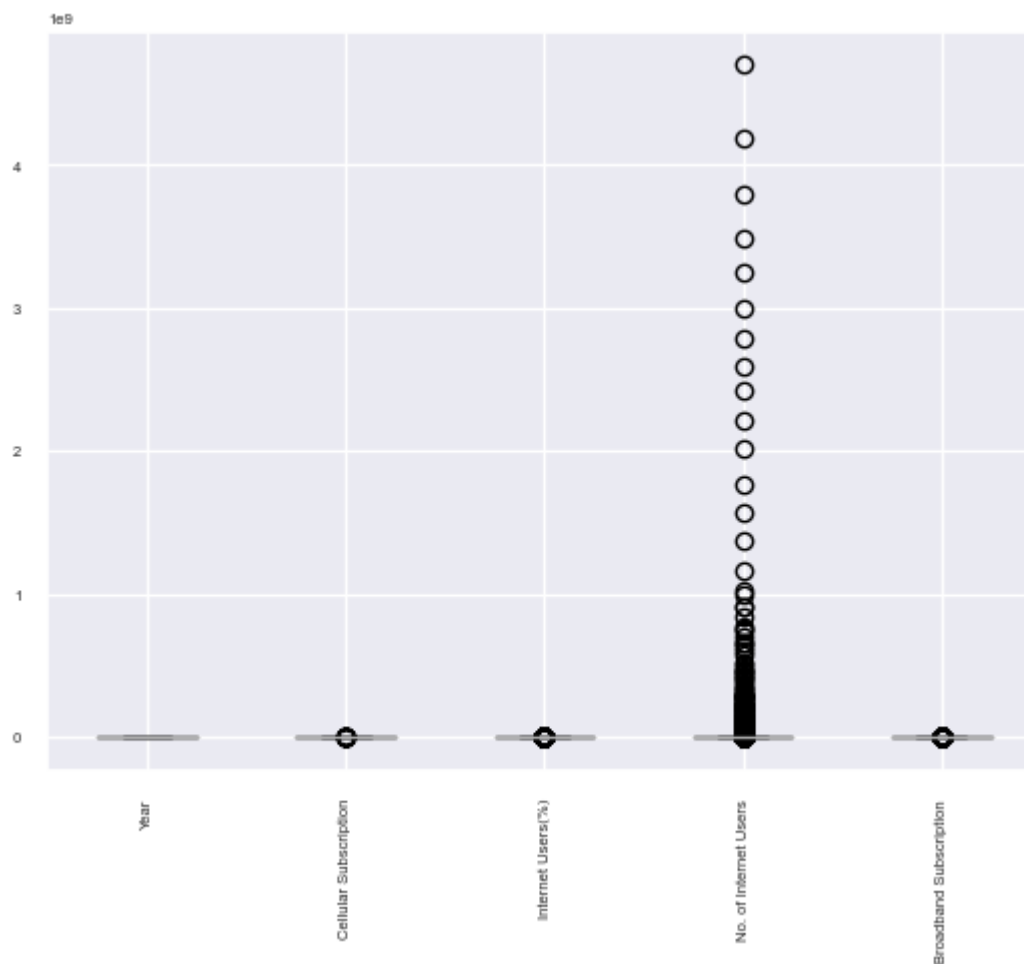
In [59]: 1 df=df.drop(['Unnamed: 0'], axis=1)

In [60]: 1 df.describe()

Out[60]:

	Year	Cellular Subscription	Internet Users(%)	No. of Internet Users	Broadband Subscription
count	8867.000000	8867.000000	8867.000000	8.867000e+03	8867.000000
mean	2000.151799	39.989614	17.043606	1.089138e+07	4.440695
std	11.812151	51.981410	26.883498	1.248841e+08	9.755705
min	1980.000000	0.000000	0.000000	0.000000e+00	0.000000
25%	1990.000000	0.000000	0.000000	0.000000e+00	0.000000
50%	2000.000000	5.501357	0.855662	1.004700e+04	0.000000
75%	2010.000000	82.231594	25.449939	8.664195e+05	2.007603
max	2020.000000	436.103027	100.000000	4.699886e+09	78.524361

```
In [61]: 1 # finding outliers
2
3 df.boxplot()
4 plt.xticks(rotation=90)
5 plt.show()
6
```



In [62]:

1 df

Out[62]:

	Entity	Code	Year	Cellular Subscription	Internet Users(%)	No. of Internet Users	Broadband Subscription
0	Afghanistan	AFG	1980	0.000000	0.000000	0	0.000000
1	Afghanistan	AFG	1981	0.000000	0.000000	0	0.000000
2	Afghanistan	AFG	1982	0.000000	0.000000	0	0.000000
3	Afghanistan	AFG	1983	0.000000	0.000000	0	0.000000
4	Afghanistan	AFG	1984	0.000000	0.000000	0	0.000000
...	...	...	...	...	...	...	...
8862	Zimbabwe	ZWE	2016	91.793457	23.119989	3341464	1.217633
8863	Zimbabwe	ZWE	2017	98.985077	24.400000	3599269	1.315694
8864	Zimbabwe	ZWE	2018	89.404869	25.000000	3763048	1.406322
8865	Zimbabwe	ZWE	2019	90.102287	25.100000	3854006	1.395818
8866	Zimbabwe	ZWE	2020	88.755806	29.299999	4591211	1.368916

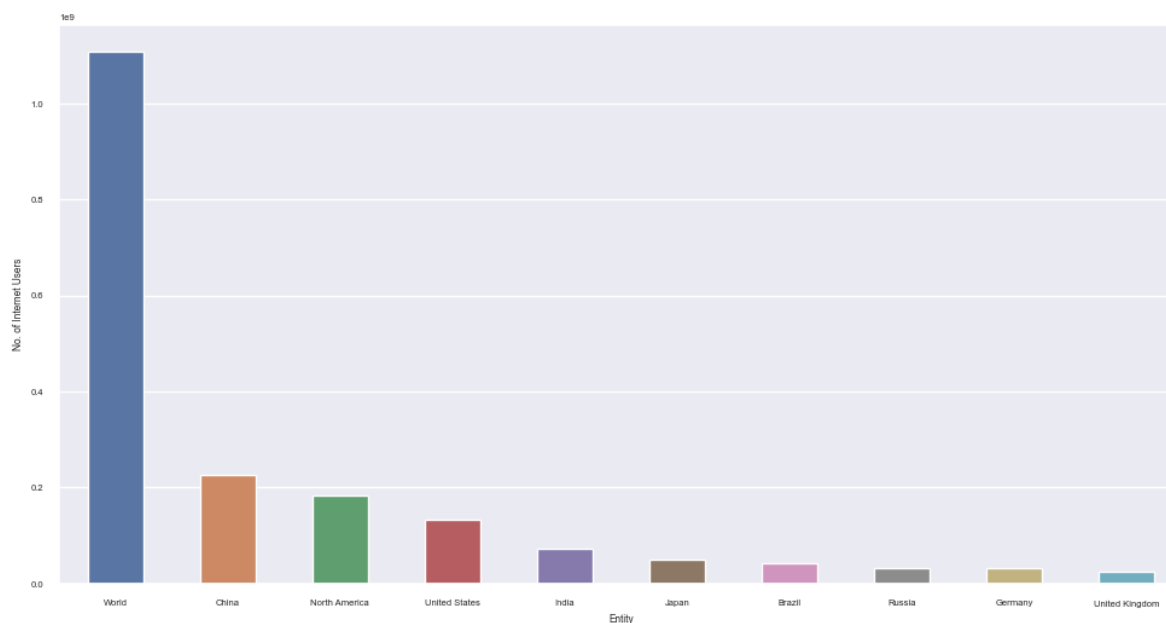
8867 rows × 7 columns

In [63]:

```

1 # Top 10 countries with highest average internet users
2 plt.figure(figsize=(12, 6))
3 d = df.groupby('Entity')['No. of Internet Users'].mean().round().nlargest(
4 sns.barplot(data=d, x = 'Entity', y = 'No. of Internet Users', width = 0.5
5 plt.show()

```

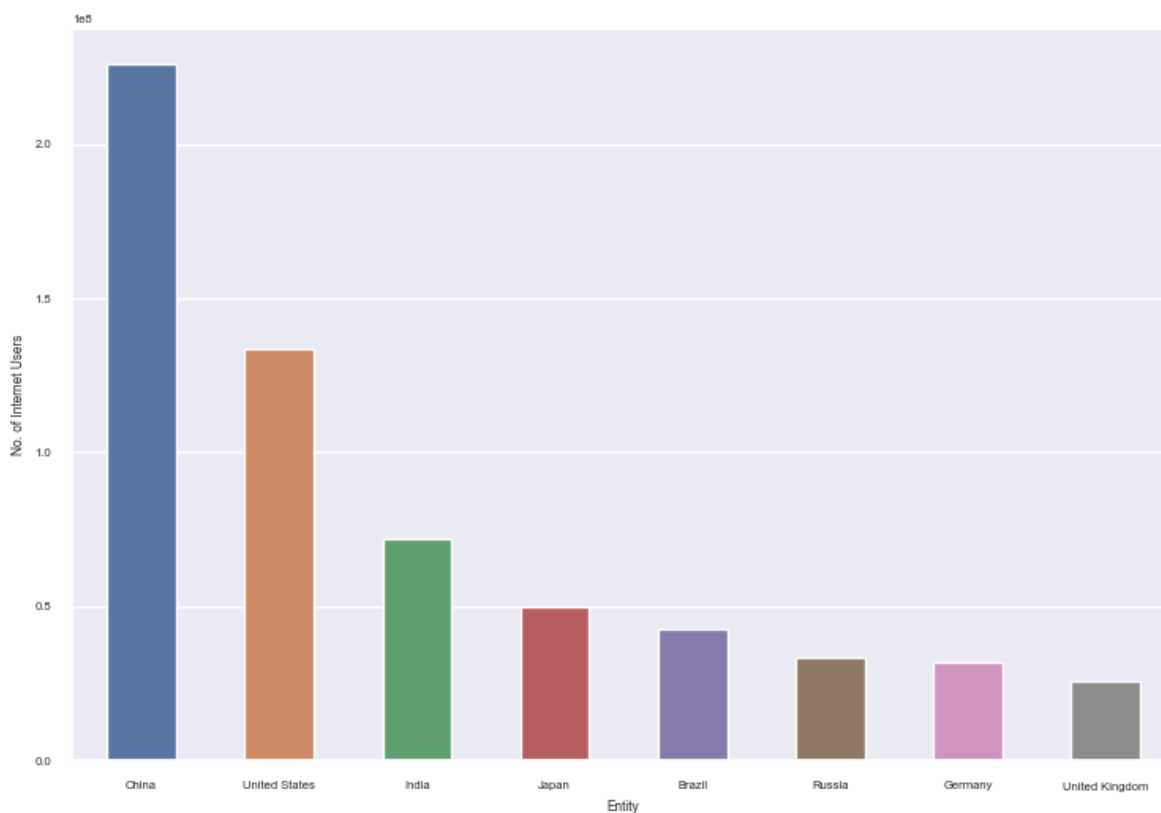


```
In [64]: 1 d = d.drop(d[(d['Entity']=='World')].index)
2         d = d.drop(d[(d['Entity']=='North America')].index)
3         d
```

Out[64]:

	Entity	No. of Internet Users
1	China	226095158.0
3	United States	133425326.0
4	India	71766932.0
5	Japan	49804800.0
6	Brazil	42340844.0
7	Russia	33302832.0
8	Germany	31658207.0
9	United Kingdom	25731958.0

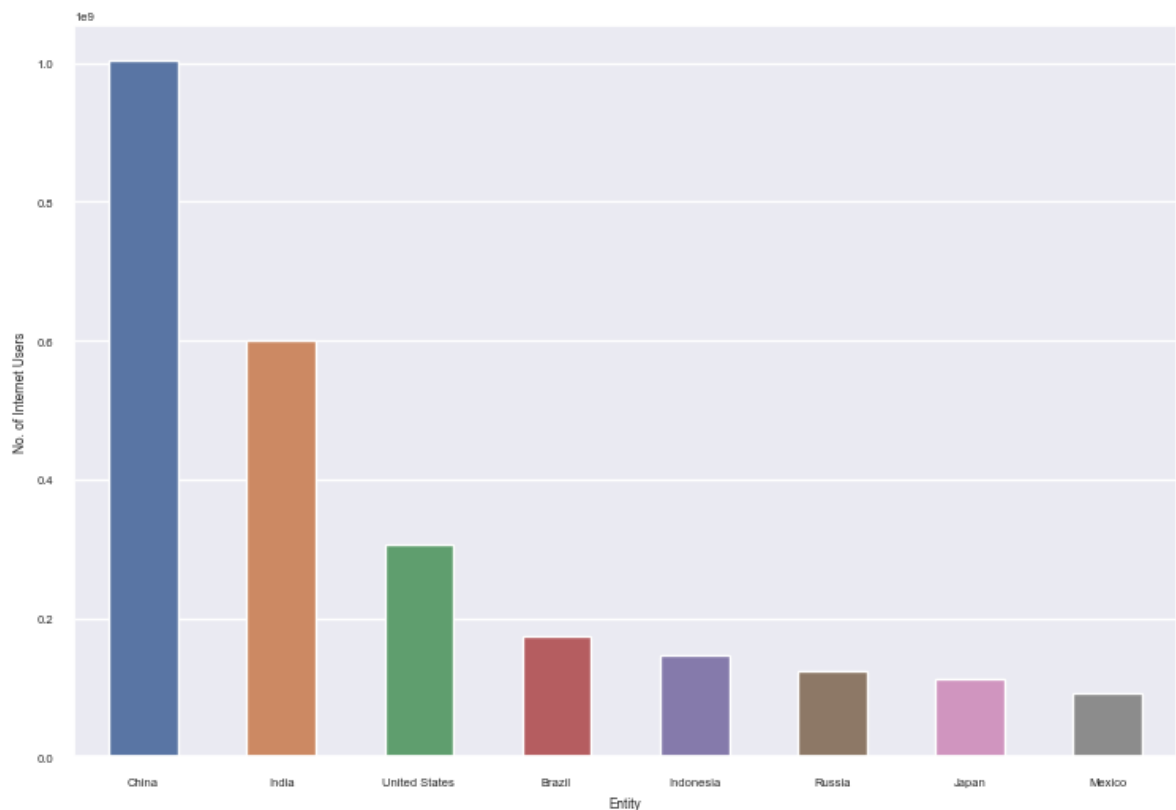
```
In [65]: 1 plt.figure(figsize=(9, 6))
2
3         sns.barplot(data=d, x = 'Entity', y = 'No. of Internet Users', width = 0.5)
4         ax.bar_label(ax.containers[0])
5         plt.show()
```



**China has more number of internet users**

## Lets check top 10 countries with highest internet users in 2020

```
In [66]: 1 u = df[df['Year']==2020].groupby('Entity')['No. of Internet Users'].mean()  
2 u = u.drop(u[(u['Entity']=='World')].index)  
3 u = u.drop(u[(u['Entity']=='North America')].index)  
4  
5 plt.figure(figsize=(9, 6))  
6 sns.barplot(data = u, x = 'Entity', y = 'No. of Internet Users', width = 0.8)  
7 ax.bar_label(ax.containers[0])  
8 plt.show()
```



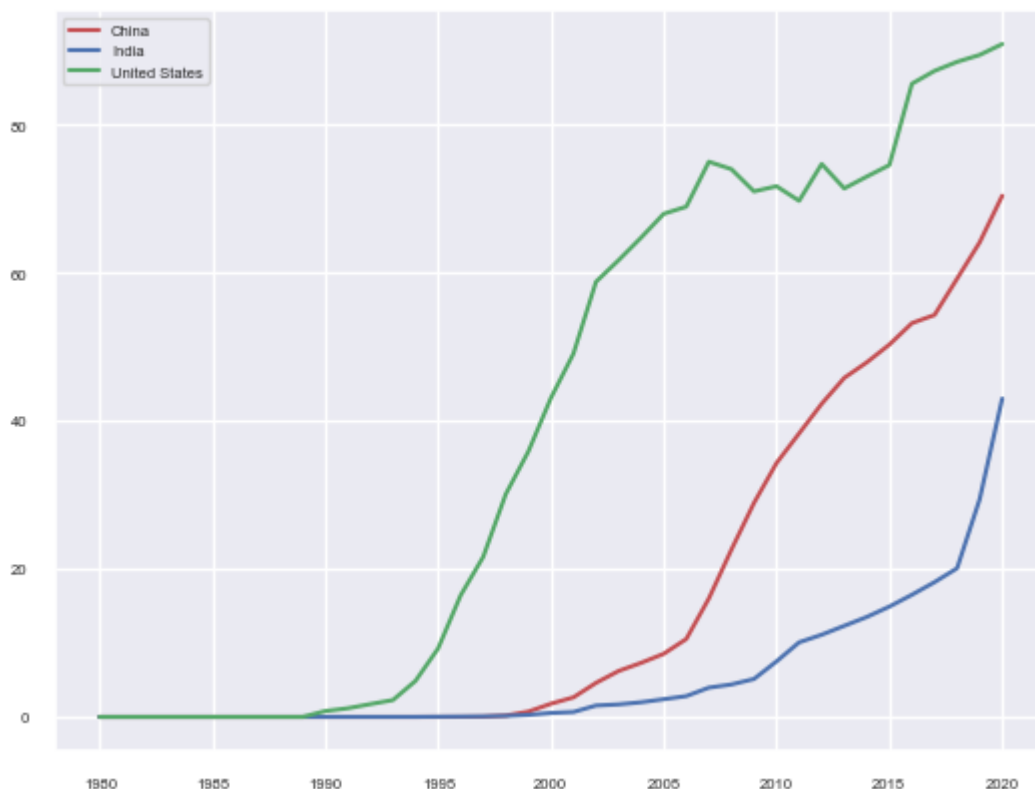
**In 2020 China and India has more number of internet users**

**Visualize China, India and United States Internet usage percentage**

```

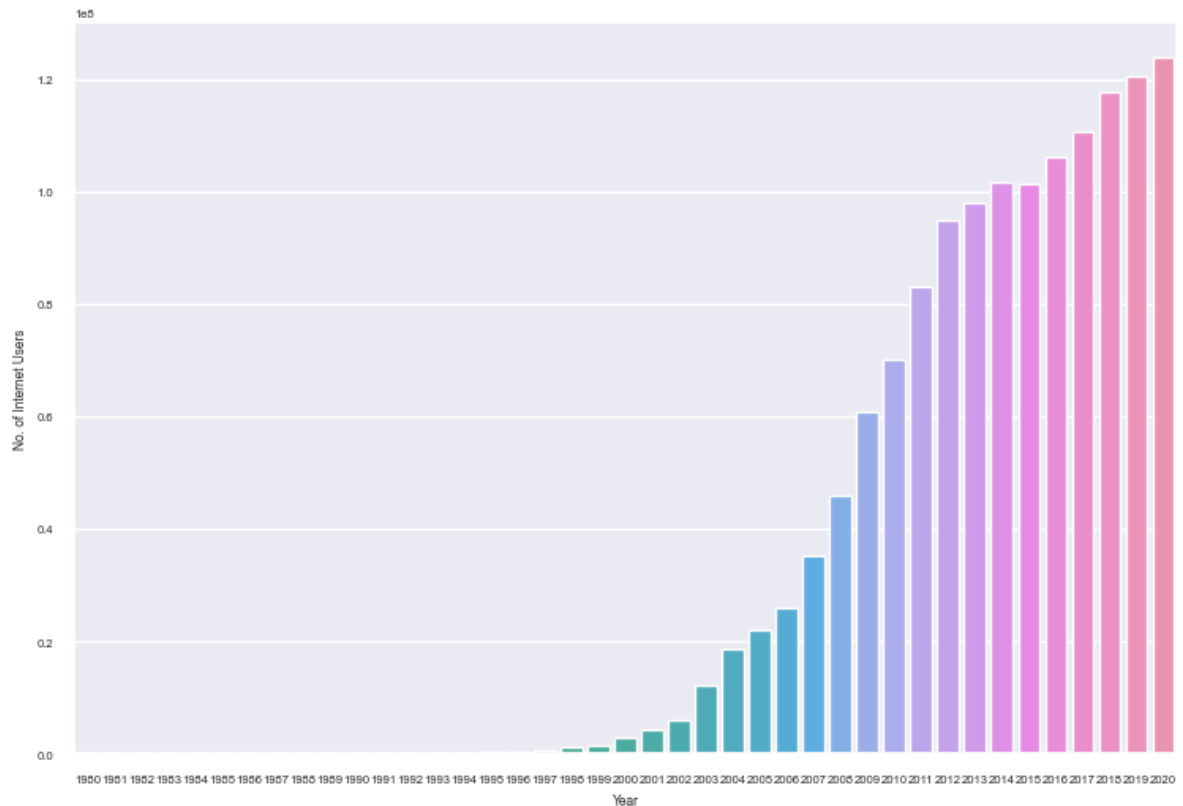
In [67]: 1 #firstly get X with China years and Y with China Internet Users (%)
2 x = df['Year'][df['Entity'] == 'China']
3 y = df['Internet Users(%)'][df['Entity'] == 'China']
4 plt.plot(x, y, color = 'r')
5 #get X with India years and Y with India Internet Users (%)
6 x = df['Year'][df['Entity'] == 'India']
7 y = df['Internet Users(%)'][df['Entity'] == 'India']
8 plt.plot(x, y, color = 'b')
9 #get X with United States years and Y with United States Internet Users (%)
10 x = df['Year'][df['Entity'] == 'United States']
11 y = df['Internet Users(%)'][df['Entity'] == 'United States']
12 plt.plot(x, y, color = 'g')
13 plt.legend(['China', 'India', 'United States'])
14 plt.show()
15

```



**Using Seaborn visualize dynamics of number of Internet users in Russia**

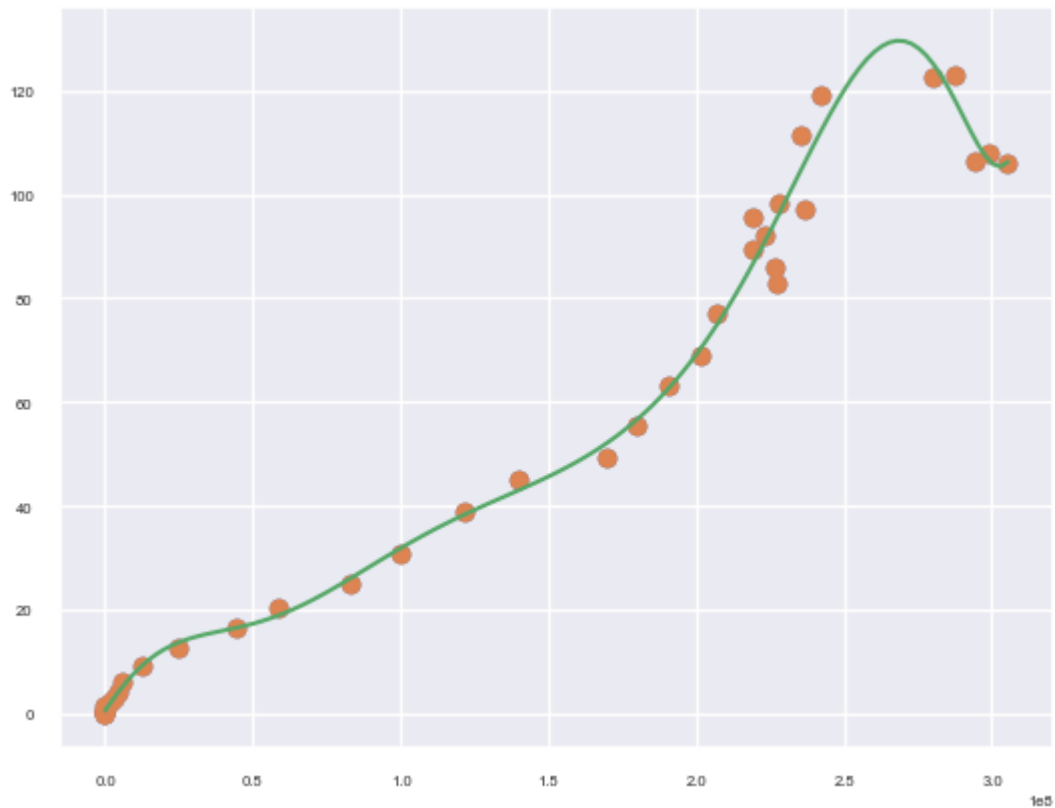
```
In [68]: 1 plt.figure(figsize=(9, 6))
2         sns.set(font_scale=.5)
3         sns.barplot(data = df, x = df['Year'][df['Entity']=='Russia'], y = df['No.
4         plt.show()
```



**Dependence of the number of Internet users and the number of subscriptions per 100 people**



```
In [69]: 1 X = df['No. of Internet Users'][df['Entity'] == 'United States']
2 y = df['Cellular Subscription'][df['Entity'] == 'United States']
3 plt.scatter(X, y)
4 mymodel = np.poly1d(np.polyfit(X, y, 10))
5
6 myline = np.linspace(int(sorted(X)[0]), int(sorted(X)[-1]), int(sorted(y)[
7
8 plt.scatter(X, y)
9 plt.plot(myline, mymodel(myline), color = 'g')
10 plt.show()
11
```



```
In [70]: 1 # Mobile Phone Subscription Users per 100 people
2 fig = px.line(df,x='Year',y=['Cellular Subscription'],color='Entity',marke
3 fig.update_layout(
4     hovermode='x unified',
5     updatemenus=[
6         dict(
7             type = "buttons",
8             direction = "left",
9             buttons=list([
10                 dict(
11                     args=[{"yaxis.type": "linear"}],
12                     label="LINEAR",
13                     method="relayout"
14                 ),
15                 dict(
16                     args=[{"yaxis.type": "log"}],
17                     label="LOG",
18                     method="relayout"
19                 )
20             ]),
21         ),
22     ]
23 )
24 fig.show()
```

Mobile phone subscriptions per 100 people 1980 to 2020

