Data source: https://www.kaggle.com/passnyc/data-science-for-good (https://www.kaggle.com/passnyc/data-science-for-good)

In [1]:
```
%matplotlib inline
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
suicide = pd.read_csv('E:/data/suicide-rates-overview-1985-to-2016/master.csv'
)
suicide.sample(1)
```

Out[2]:

	country	year	sex	age	suicides_no	population	suicides/100k pop	country-year	HDI for year	gc
19829	Puerto Rico	1997	female	35-54 years	12	503717	2.38	Puerto Rico1997	NaN	48,

In [3]:
```
suicide['year'].head()
```

Out[3]:
```
0    1987
1    1987
2    1987
3    1987
4    1987
Name: year, dtype: int64
```

In [4]:
```
suicide.groupby(['country'])['year'].min().sort_values().tail(15)
```

Out[4]:
```
country
Latvia                    1995
South Africa              1996
Serbia                    1998
San Marino                1999
Cyprus                    1999
Montenegro                2000
Maldives                  2000
Fiji                      2001
United Arab Emirates      2005
Turkey                    2009
Oman                      2009
Nicaragua                 2010
Cabo Verde                2011
Bosnia and Herzegovina    2011
Mongolia                  2016
Name: year, dtype: int64
```

In [5]:
```
suicide = suicide[~suicide.country.isin(['South Africa','Serbia','San Marino',
'Cyprus','Montenegro','Maldives','Fiji','United Arab Emirates','Turkey','Oman'
,'Nicaragua','Cabo Verde','Bosnia and Herzegovina','Mongolia'])]
```

In [6]:
```
suicide.groupby(['country'])['year'].max().sort_values().head(15)
```

Out[6]:
```
country
Dominica                1985
Saint Kitts and Nevis   1992
Macau                   1994
Kiribati                2001
Sri Lanka               2006
Azerbaijan              2007
Albania                 2010
Philippines             2011
Aruba                   2011
Jamaica                 2011
Trinidad and Tobago     2011
Bahamas                 2013
Barbados                2013
Guyana                  2013
New Zealand             2013
Name: year, dtype: int64
```

In [7]:
```
suicide = suicide[~suicide.country.isin(['Dominica','Saint Kitts and Nevis','M
acau','Kiribati','Sri Lanka','Azerbaijan','Albania','Philippines','Aruba','Jam
aica','Trinidad and Tobago'])]
```

In [8]:
```
suicide = suicide.drop(['HDI for year','country-year'],axis=1)
```

In [9]:
```
suicide.head(1)
```

Out[9]:

	country	year	sex	age	suicides_no	population	suicides/100k pop	gdp_for_year ($)	gdp_pe
264	Antigua and Barbuda	1985	female	15-24 years	0	7709	0.0	240,923,926	

In [10]:
```
list(suicide)
```

Out[10]:
```
['country',
 'year',
 'sex',
 'age',
 'suicides_no',
 'population',
 'suicides/100k pop',
 ' gdp_for_year ($) ',
 'gdp_per_capita ($)',
 'generation']
```

In [11]:
```
suicide.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 24812 entries, 264 to 27819
Data columns (total 10 columns):
country               24812 non-null object
year                  24812 non-null int64
sex                   24812 non-null object
age                   24812 non-null object
suicides_no           24812 non-null int64
population            24812 non-null int64
suicides/100k pop     24812 non-null float64
 gdp_for_year ($)     24812 non-null object
gdp_per_capita ($)    24812 non-null int64
generation            24812 non-null object
dtypes: float64(1), int64(4), object(5)
memory usage: 1.6+ MB
```

In [12]:
```
suicide['sex'] = suicide['sex'].astype('category')
suicide['age']=suicide['age'].astype('category')
suicide['generation']=suicide['generation'].astype('category')
suicide['gdp_for_year ($)']=suicide[' gdp_for_year ($) '].astype(str)
suicide=suicide.drop(' gdp_for_year ($) ',axis=1)
```

In [13]:
```
suicide.info()
```

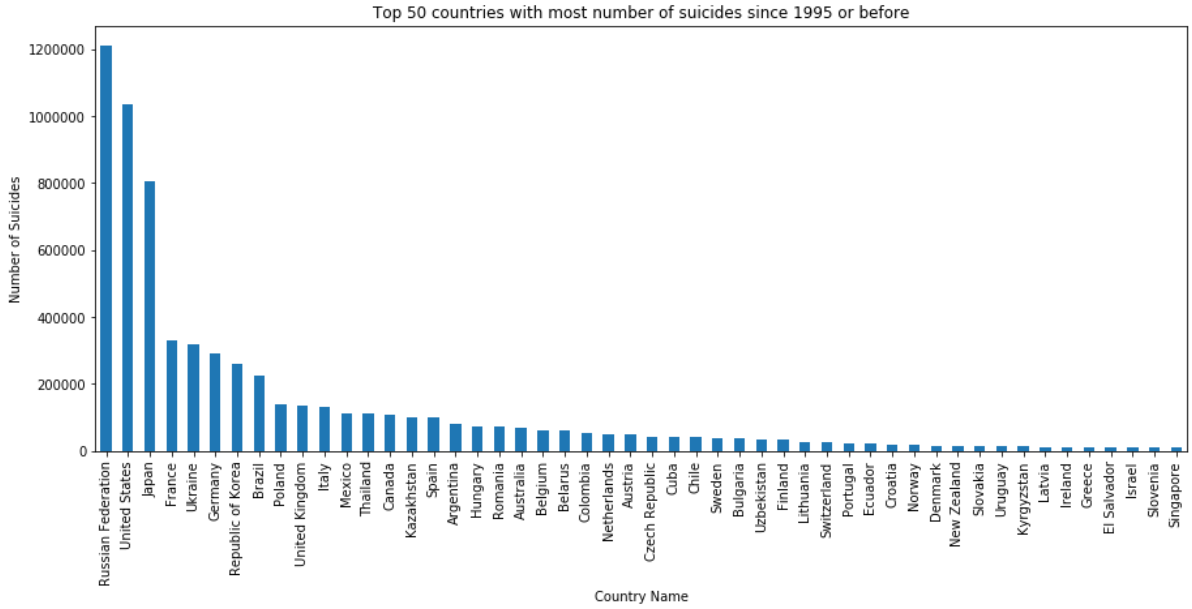
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 24812 entries, 264 to 27819
Data columns (total 10 columns):
country               24812 non-null object
year                  24812 non-null int64
sex                   24812 non-null category
age                   24812 non-null category
suicides_no           24812 non-null int64
population            24812 non-null int64
suicides/100k pop     24812 non-null float64
gdp_per_capita ($)    24812 non-null int64
generation            24812 non-null category
gdp_for_year ($)      24812 non-null object
dtypes: category(3), float64(1), int64(4), object(2)
memory usage: 1.4+ MB
```

In [14]:
```
suicide['gdp_for_year ($)']=suicide['gdp_for_year ($)'].str.replace(",","")
```

In [15]:
```
suicide['gdp_for_year ($)']=suicide['gdp_for_year ($)'].astype('int64')
suicide['gdp_for_year ($)']=suicide['gdp_for_year ($)']
suicide['gdp_for_year ($)'].sample(2)
```

Out[15]:
```
13682    6157459594824
16749     707906744575
Name: gdp_for_year ($), dtype: int64
```

In [16]:
```
plt.figure(figsize=(15,6))
suicide.groupby(['country'])['suicides_no'].sum().sort_values(ascending=False)
.head(50).plot(kind='bar');
plt.ylabel('Number of Suicides')
plt.xlabel('Country Name')
plt.title('Top 50 countries with most number of suicides since 1995 or before'
);
```

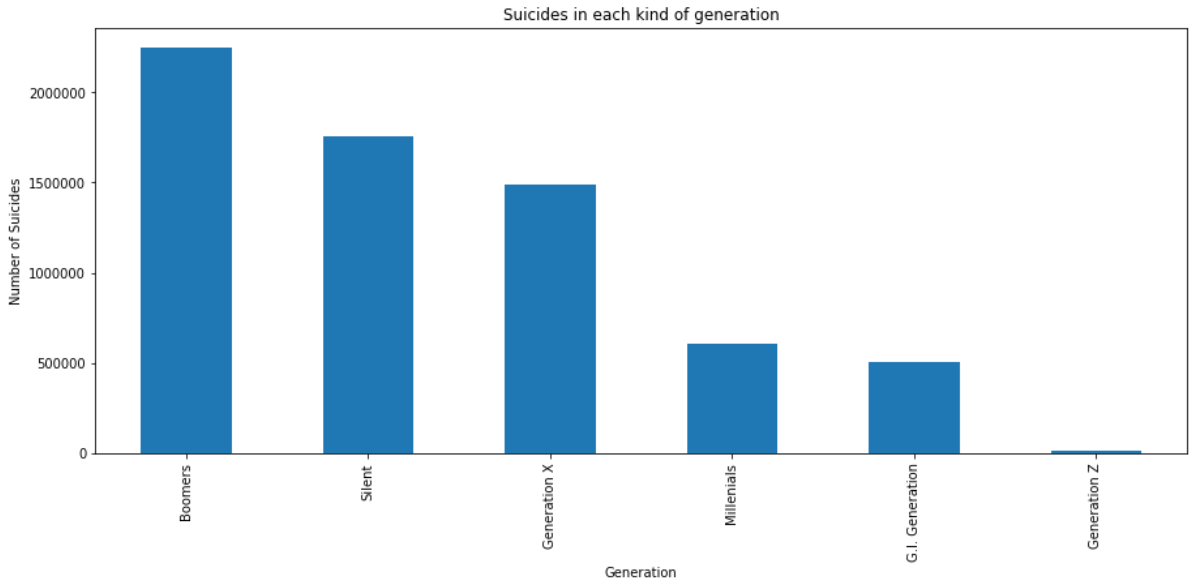


In [17]:
```
suicide.groupby(['country'])['suicides_no'].sum().sort_values(ascending=False)
.head(10)
```

Out[17]:
```
country
Russian Federation    1209742
United States         1034013
Japan                  806902
France                 329127
Ukraine                319950
Germany                291262
Republic of Korea      261730
Brazil                 226613
Poland                 139098
United Kingdom         136805
Name: suicides_no, dtype: int64
```

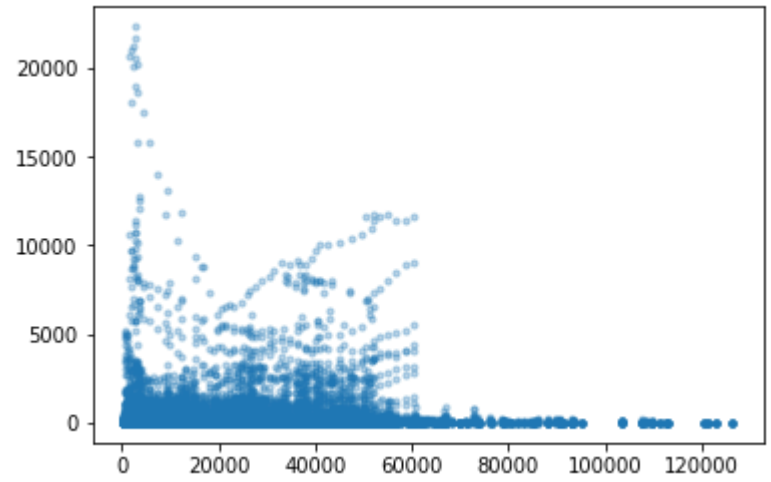
In [18]:
```
Top10 = suicide[suicide.country.isin(['Russian Federation','United States','Ja
pan','France','Ukraine','Germany','Republic of Korea','Brazil','Poland','Unite
d Kingdom'])]
```

In [19]:
```
plt.figure(figsize=(15,6))
suicide.groupby(['generation'])['suicides_no'].sum().sort_values(ascending=False).plot(kind='bar');
plt.ylabel('Number of Suicides')
plt.xlabel('Generation')
plt.title('Suicides in each kind of generation');
```



In [20]:
```
suicide['gdp_per_capita ($)'].replace(0, np.nan, inplace=True)
```

In [21]:
```
plt.scatter(x=suicide['gdp_per_capita ($)'],y=suicide['suicides_no'], alpha=0.3,s=10);
```



In [22]:
```
By_Country = suicide.groupby(['country'],as_index=False).agg({'suicides_no':'sum','population':'mean','suicides/100k pop':'mean','gdp_per_capita ($)':'mean','gdp_for_year ($)':'mean'})
```

In [23]:
```
By_Country
```

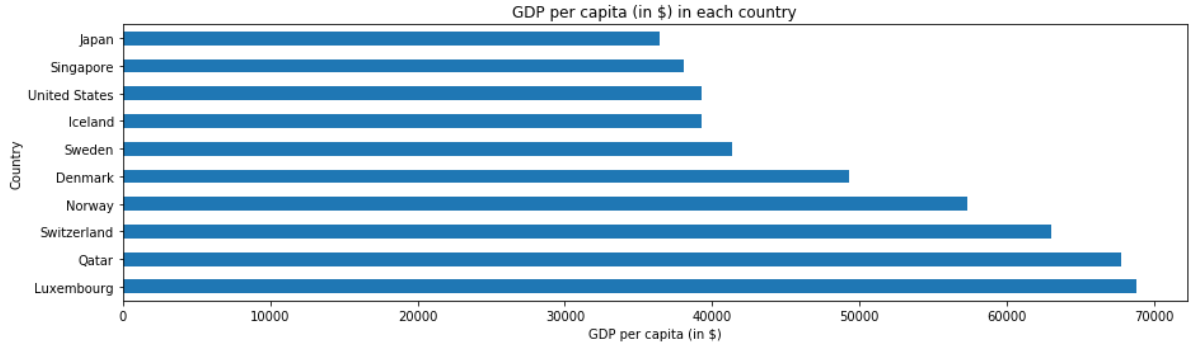
Out[23]:

	country	gdp_for_year ($)	gdp_per_capita ($)	suicides_no	suicides/100k pop	population
0	Antigua and Barbuda	8.035452e+08	10448.185185	11	0.552901	6.142679e+03
1	Argentina	2.742565e+11	7914.096774	82219	10.469328	2.784907e+06
2	Armenia	5.386592e+09	1873.919463	1905	3.275872	2.595576e+05
3	Australia	6.327501e+11	32776.400000	70111	12.992806	1.506605e+06
4	Austria	2.660162e+11	34261.780105	50073	23.759764	6.383589e+05
5	Bahamas	7.613828e+09	25836.391304	93	1.247391	2.375742e+04
6	Bahrain	1.608647e+10	18039.523810	463	1.854127	6.648383e+04
7	Barbados	3.090574e+09	12251.000000	177	2.970433	2.071230e+04
8	Belarus	3.067641e+10	3333.904762	59892	31.075913	7.832234e+05
9	Belgium	3.184024e+11	32066.741935	62761	21.237903	8.153296e+05
10	Belize	9.560371e+08	4006.464286	348	6.230625	1.871731e+04
11	Brazil	1.022561e+12	6091.483871	226613	5.846022	1.305401e+07
12	Bulgaria	2.714460e+10	3640.433333	36388	19.489111	6.442777e+05
13	Canada	9.131876e+11	30887.482759	107561	12.467586	2.373713e+06
14	Chile	1.112543e+11	7493.064516	40895	10.542043	1.149530e+06
15	Colombia	1.444638e+11	3708.967742	53080	5.401586	2.978339e+06
16	Costa Rica	1.918353e+10	5149.700000	6792	7.093667	2.838244e+05
17	Croatia	4.317814e+10	10355.870229	18429	22.835267	3.501158e+05
18	Cuba	4.608462e+10	4351.166667	41418	21.222049	8.742713e+05
19	Czech Republic	1.220562e+11	12369.546584	43687	18.487547	8.202066e+05
20	Denmark	2.530342e+11	49299.909091	15297	14.097159	4.249326e+05
21	Ecuador	3.960132e+10	3286.258065	20660	6.304328	9.256129e+05
22	El Salvador	1.354975e+10	2550.666667	11683	10.541458	4.329295e+05
23	Estonia	1.444586e+10	11376.095238	7034	27.276905	1.075032e+05
24	Finland	1.753008e+11	35468.275862	33677	22.770431	4.078324e+05
25	France	1.781194e+12	31481.466667	329127	21.675694	4.640991e+06
26	Georgia	7.776764e+09	1893.136364	3224	4.228712	3.545409e+05
27	Germany	2.742233e+12	35164.230769	291262	15.559904	6.489986e+06
28	Greece	1.766047e+11	17019.387097	12368	4.064839	8.497207e+05
29	Grenada	5.762393e+08	6209.406452	38	2.132258	7.571890e+03
...	...	...	...	...	...	...
46	Mexico	6.803078e+11	7138.451613	111139	4.707500	7.454191e+06
47	Netherlands	5.425437e+11	35714.670157	50833	10.645340	1.244401e+06
48	New Zealand	8.269044e+10	22279.482759	14383	14.391724	2.962369e+05

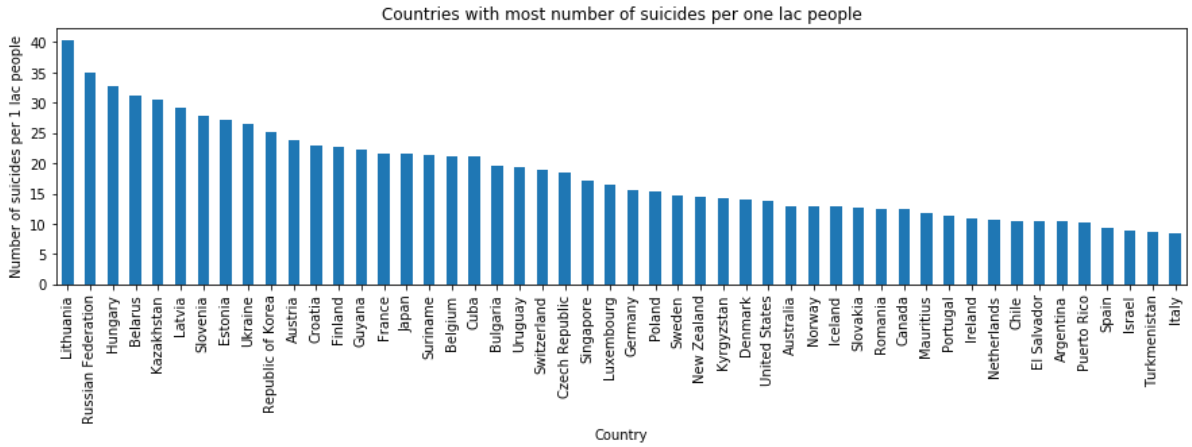
	country	gdp_for_year ($)	gdp_per_capita ($)	suicides_no	suicides/100k pop	population
49	Norway	2.525657e+11	57319.600000	16992	12.941000	3.554808e+05
50	Panama	2.020910e+10	6626.520000	3483	5.813533	2.342189e+05
51	Paraguay	1.168725e+10	2365.777778	4783	4.217191	3.846974e+05
52	Poland	2.944583e+11	8146.458333	139098	15.269514	3.007570e+06
53	Portugal	1.389200e+11	14176.296296	24061	11.337531	8.063544e+05
54	Puerto Rico	6.322157e+10	18352.645161	9043	10.186237	2.832974e+05
55	Qatar	1.100206e+11	67756.449438	574	1.787360	1.228682e+05
56	Republic of Korea	6.734208e+11	14801.258065	261730	25.135618	3.642325e+06
57	Romania	9.608785e+10	4791.449102	72777	12.489671	1.721754e+06
58	Russian Federation	8.843229e+11	6518.814815	1209742	34.892377	1.139137e+07
59	Saint Lucia	8.409455e+08	5789.035714	230	7.202738	1.163845e+04
60	Saint Vincent and Grenadines	4.884483e+08	4954.640000	124	5.755433	8.125540e+03
61	Seychelles	8.564692e+08	10655.722222	98	7.480093	6.448222e+03
62	Singapore	1.254921e+11	38050.258065	10089	17.045645	2.533399e+05
63	Slovakia	5.379573e+10	10526.000000	13437	12.568788	4.236125e+05
64	Slovenia	3.590006e+10	18642.238095	10615	27.827857	1.597961e+05
65	Spain	8.565680e+11	20982.161290	100202	9.432957	3.300164e+06
66	Suriname	1.943388e+09	4351.964286	2166	21.316429	3.464068e+04
67	Sweden	3.549738e+11	41357.575419	37795	14.658436	7.075339e+05
68	Switzerland	4.565306e+11	62981.761905	26217	19.024087	5.952143e+05
69	Thailand	2.095240e+11	3572.161677	110643	7.073862	4.744911e+06
70	Turkmenistan	1.157956e+10	2618.103448	8624	8.605546	3.321173e+05
71	Ukraine	8.389111e+10	1867.535714	319950	26.582321	3.828777e+06
72	United Kingdom	1.816067e+12	31908.354839	136805	7.502473	4.674107e+06
73	United States	1.051071e+13	39269.612903	1034013	13.819812	2.165061e+07
74	Uruguay	2.337827e+10	7622.071429	13138	19.461190	2.502052e+05
75	Uzbekistan	2.286003e+10	976.181818	34803	8.099129	1.842510e+06

76 rows × 6 columns

In [24]:
```
plt.figure(figsize=(15,4))
CIndex = By_Country.set_index('country')
CIndex['gdp_per_capita ($)'].sort_values(ascending=False).head(10).plot(kind=
'barh');
plt.xlabel('GDP per capita (in $)')
plt.ylabel('Country')
plt.title('GDP per capita (in $) in each country');
```

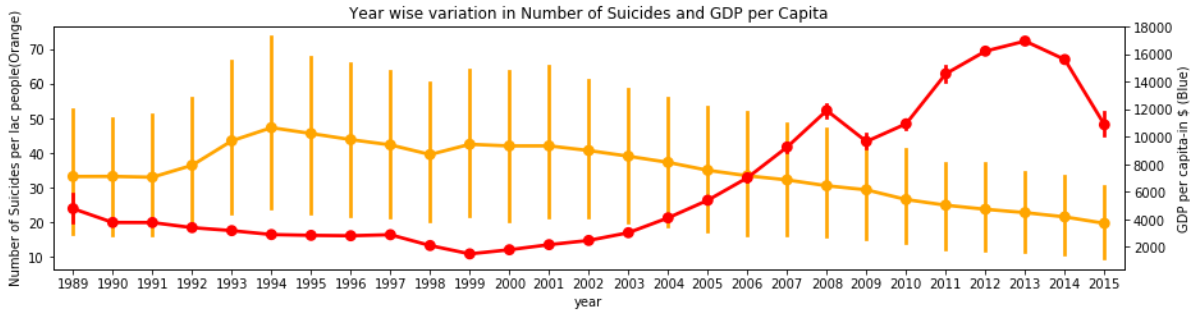


In [25]:
```
plt.figure(figsize=(15,3.5))
CIndex['suicides/100k pop'].sort_values(ascending=False).head(50).plot(kind='b
ar');
plt.ylabel('Number of suicides per 1 lac people')
plt.xlabel('Country')
plt.title('Countries with most number of suicides per one lac people');
```

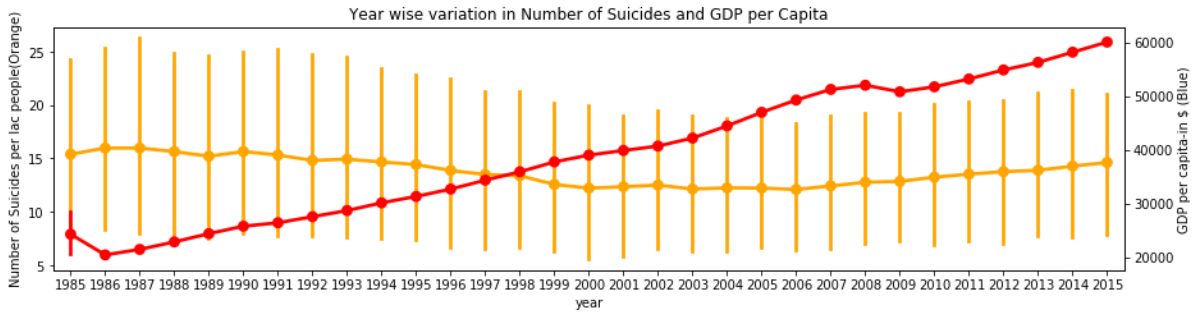


In [64]:
```
def yearwise_variation_in(x):
    plt.figure(figsize=(15,3.5));
    ax1 = sns.pointplot(x='year',y='suicides/100k pop',data=suicide[suicide['c
ountry']==x], color = 'orange');
    ax2 = ax1.twinx()
    sns.pointplot(x='year',y=suicide['gdp_per_capita ($)'].rolling(window=5).m
ean(),data=suicide[suicide['country']==x], ax = ax2, color = 'red');
    ax1.set_ylabel('Number of Suicides per lac people(Orange)')
    ax2.set_ylabel('GDP per capita-in $ (Blue)');
    plt.title('Year wise variation in Number of Suicides and GDP per Capita');
```

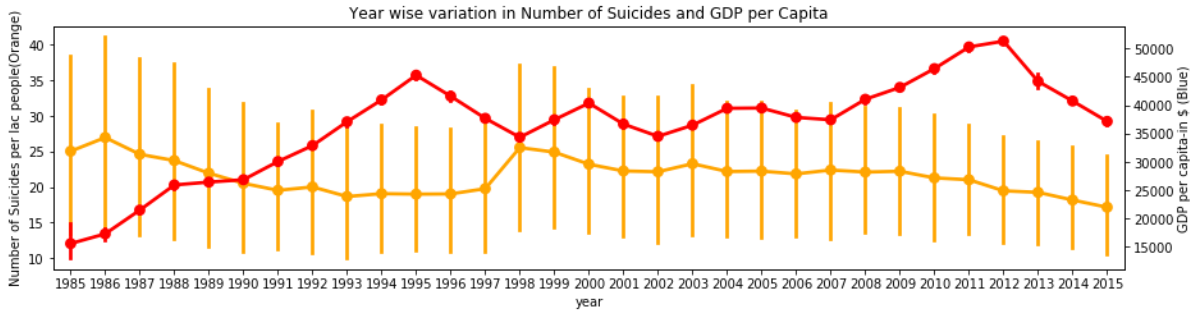
In [65]:
```
yearwise_variation_in('Russian Federation')
```



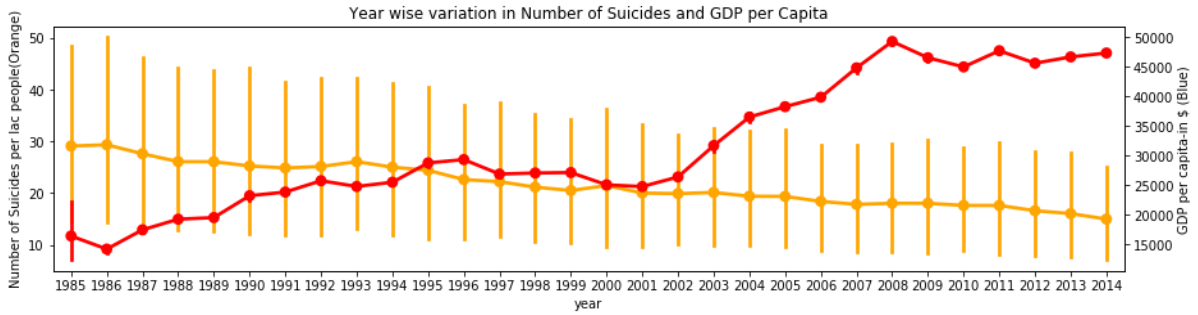
In [66]:
```
yearwise_variation_in('United States')
```



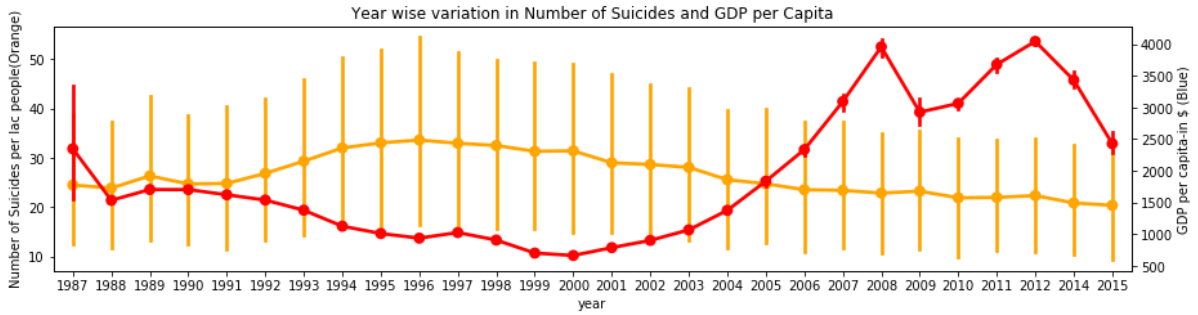
In [68]:
```
yearwise_variation_in('Japan')
```



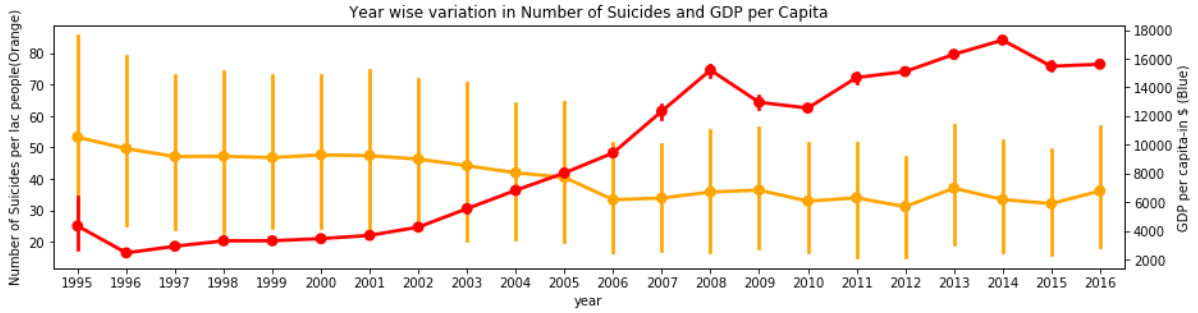
In [69]:
```
yearwise_variation_in('France')
```



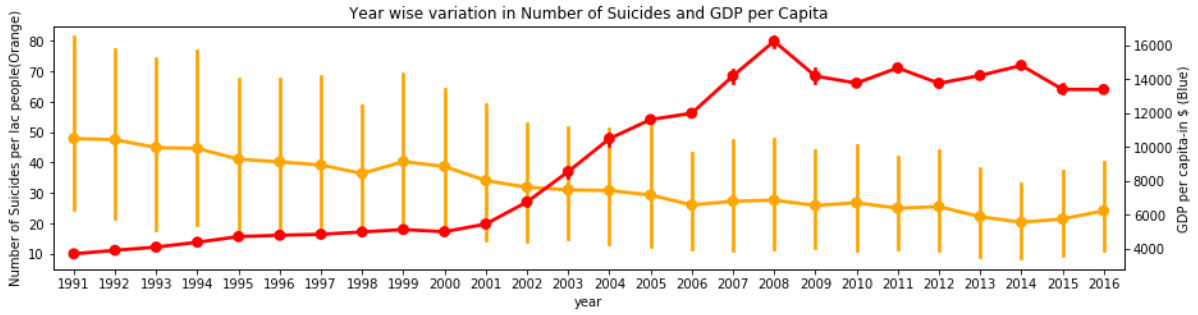
In [70]:
```
yearwise_variation_in('Ukraine')
```



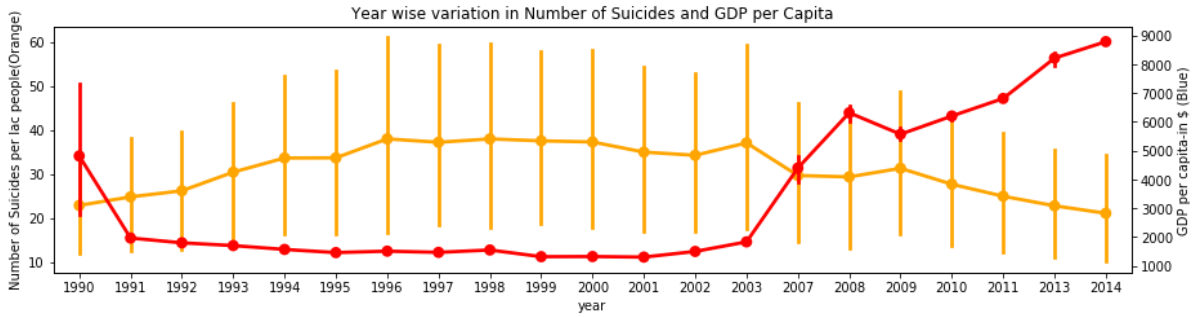
In [71]:
```
yearwise_variation_in('Lithuania')
```



In [72]:
```
yearwise_variation_in('Hungary')
```



In [73]:
```
yearwise_variation_in('Belarus')
```



In [35]:
```
By_Year = suicide.groupby(['year'],as_index=False).agg({'suicides_no':'sum','population':'sum','suicides/100k pop':'sum'})
By_Year.tail(1)
```

Out[35]:

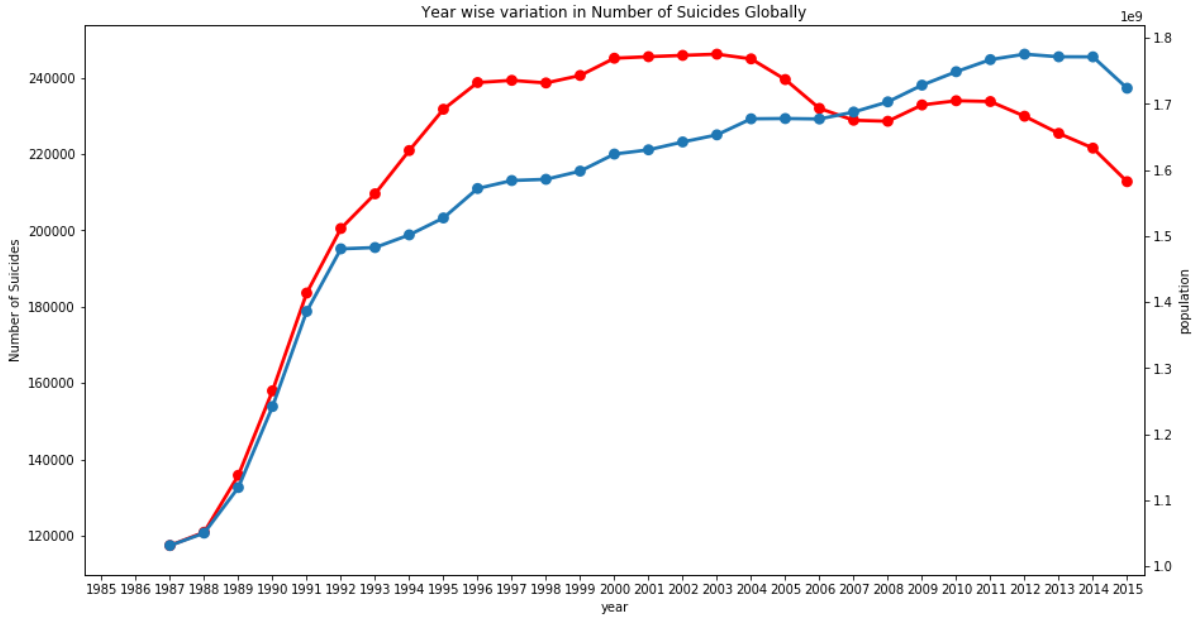
	year	suicides_no	suicides/100k pop	population
31	2016	15144	1915.85	129222963

In [36]:
```
By_Year=By_Year.drop(31)
By_Year.tail(1)
```

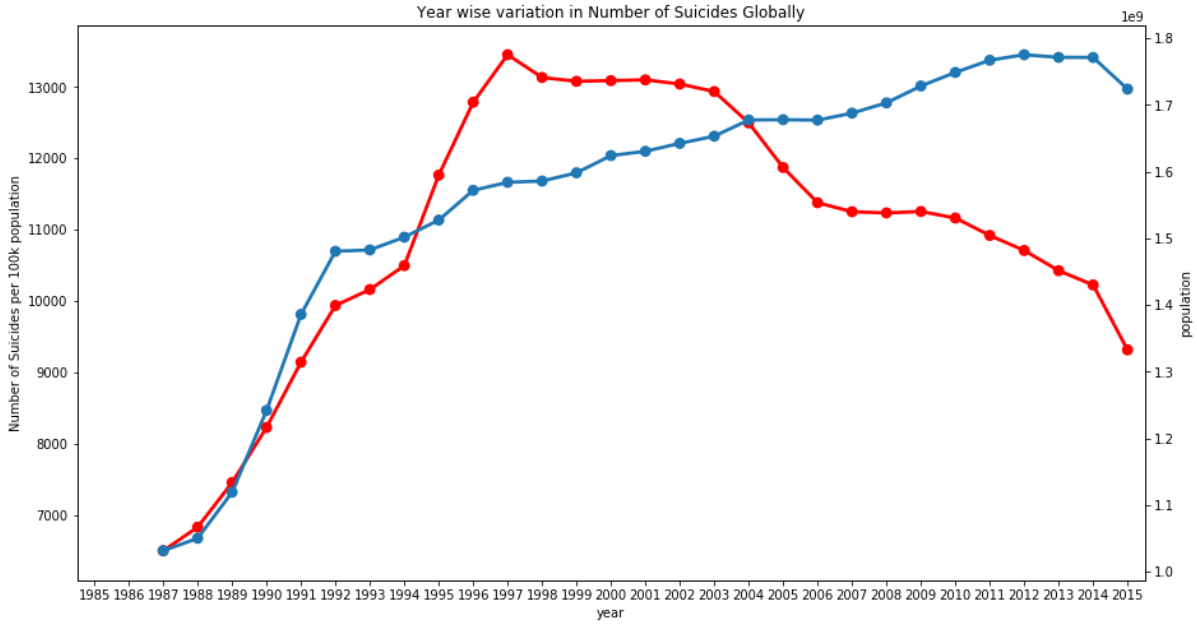
Out[36]:

	year	suicides_no	suicides/100k pop	population
30	2015	200209	7887.09	1640111082

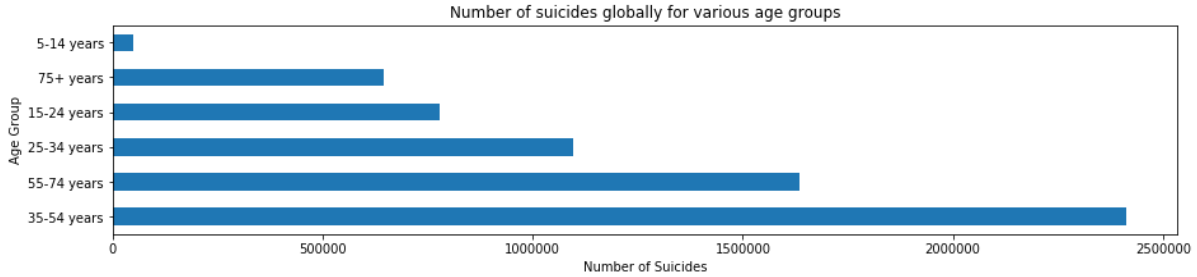
In [37]:
```
plt.figure(figsize=(15,8));
ax1 = sns.pointplot(x='year',y=By_Year['suicides_no'].rolling(window=3).mean(),data=By_Year, color = 'red');
ax2 = ax1.twinx()
sns.pointplot(x='year',y=By_Year['population'].rolling(window=3).mean(),data=By_Year, ax = ax2);
plt.xlabel('Year')
ax1.set_ylabel('Number of Suicides');
plt.title('Year wise variation in Number of Suicides Globally');
```



In [38]:
```
plt.figure(figsize=(15,8));
ax1 = sns.pointplot(x='year',y=By_Year['suicides/100k pop'].rolling(window=3).
mean(),data=By_Year, color = 'red');
ax2 = ax1.twinx()
sns.pointplot(x='year',y=By_Year['population'].rolling(window=3).mean(),data=B
y_Year, ax = ax2);
plt.xlabel('Year')
ax1.set_ylabel('Number of Suicides per 100k population');
plt.title('Year wise variation in Number of Suicides Globally');
```



In [39]:
```
plt.figure(figsize=(15,3))
suicide.groupby(['age'])['suicides_no'].sum().sort_values(ascending=False).hea
d(50).plot(kind='barh');
plt.xlabel('Number of Suicides')
plt.ylabel('Age Group')
plt.title('Number of suicides globally for various age groups');
```

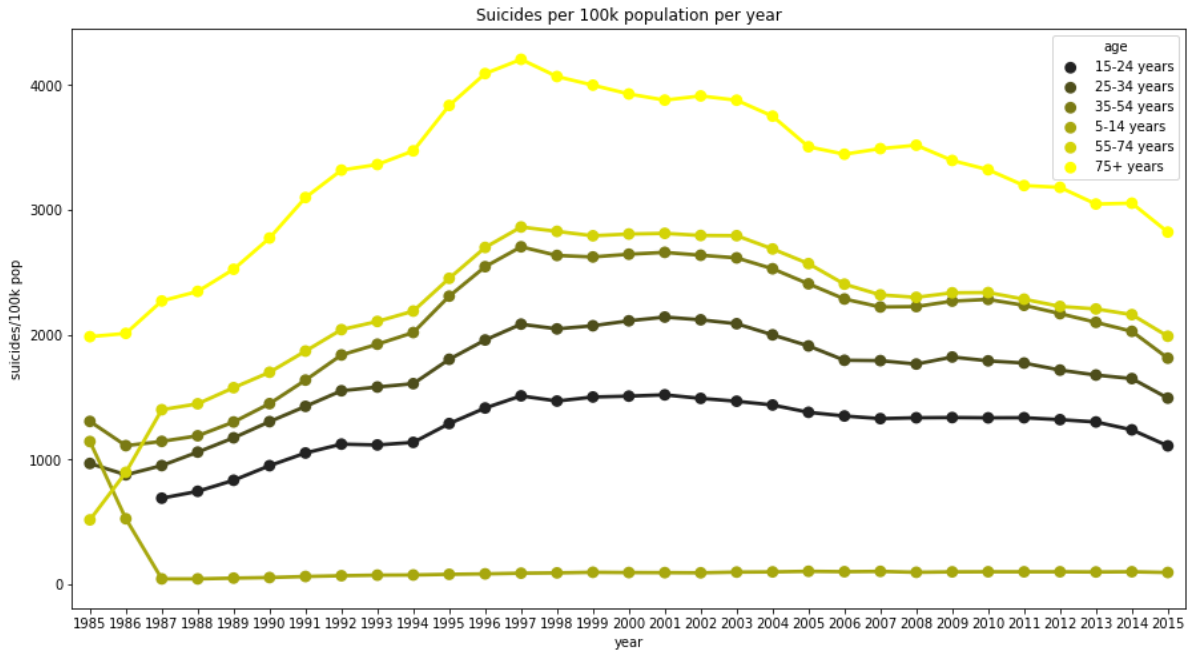


In [40]:
```
By_Age = suicide.groupby(['age','year'],as_index=False).agg({'suicides_no':'sum','population':'sum','suicides/100k pop':'sum'})
By_Age = By_Age[By_Age['year'] != 2016]
By_Age.sample(3)
```

Out[40]:

	age	year	suicides_no	suicides/100k pop	population
167	75+ years	1993	20494	3382.76	67786033
2	15-24 years	1987	16359	770.09	207211032
145	55-74 years	2003	59405	2796.44	289186419

In [41]:
```
plt.figure(figsize=(15,8));
sns.pointplot(x='year',y=By_Age['suicides/100k pop'].rolling(window=3).mean(),
data=By_Age, hue='age' ,color = 'yellow');
plt.title('Suicides per 100k population per year');
```

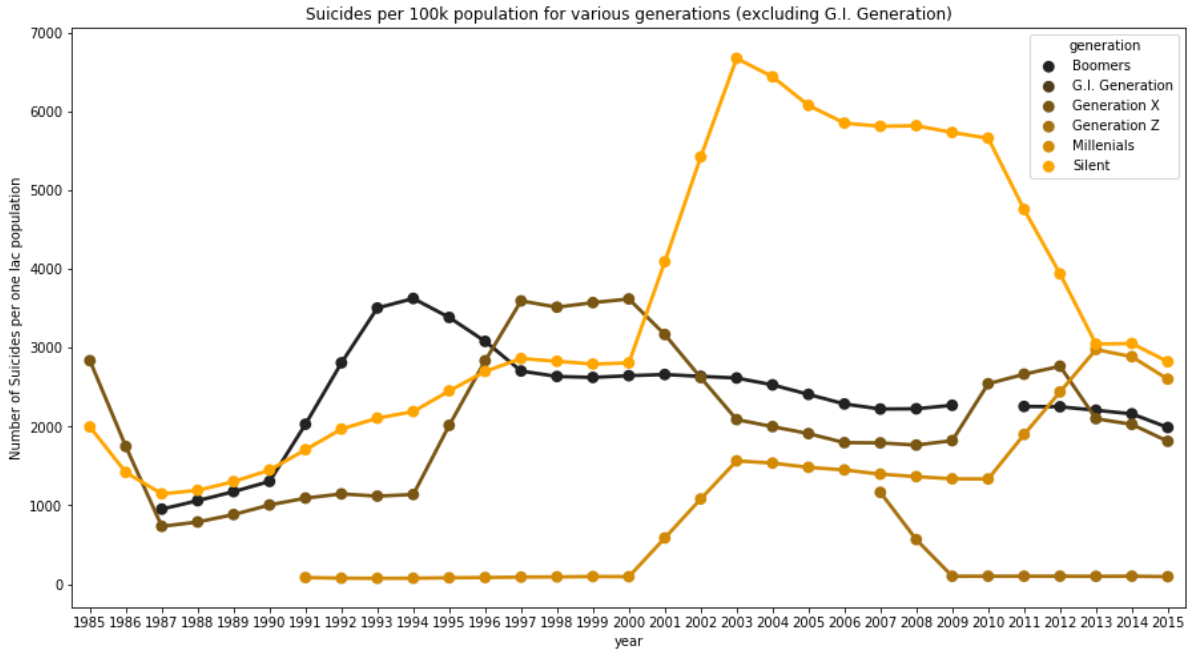


In [42]:
```
By_generation = suicide.groupby(['generation','year'],as_index=False).agg({'suicides_no':'sum','population':'sum','suicides/100k pop':'sum'})
By_generation = By_generation[By_generation['year'] != 2016]
By_generation.sample(3)
```

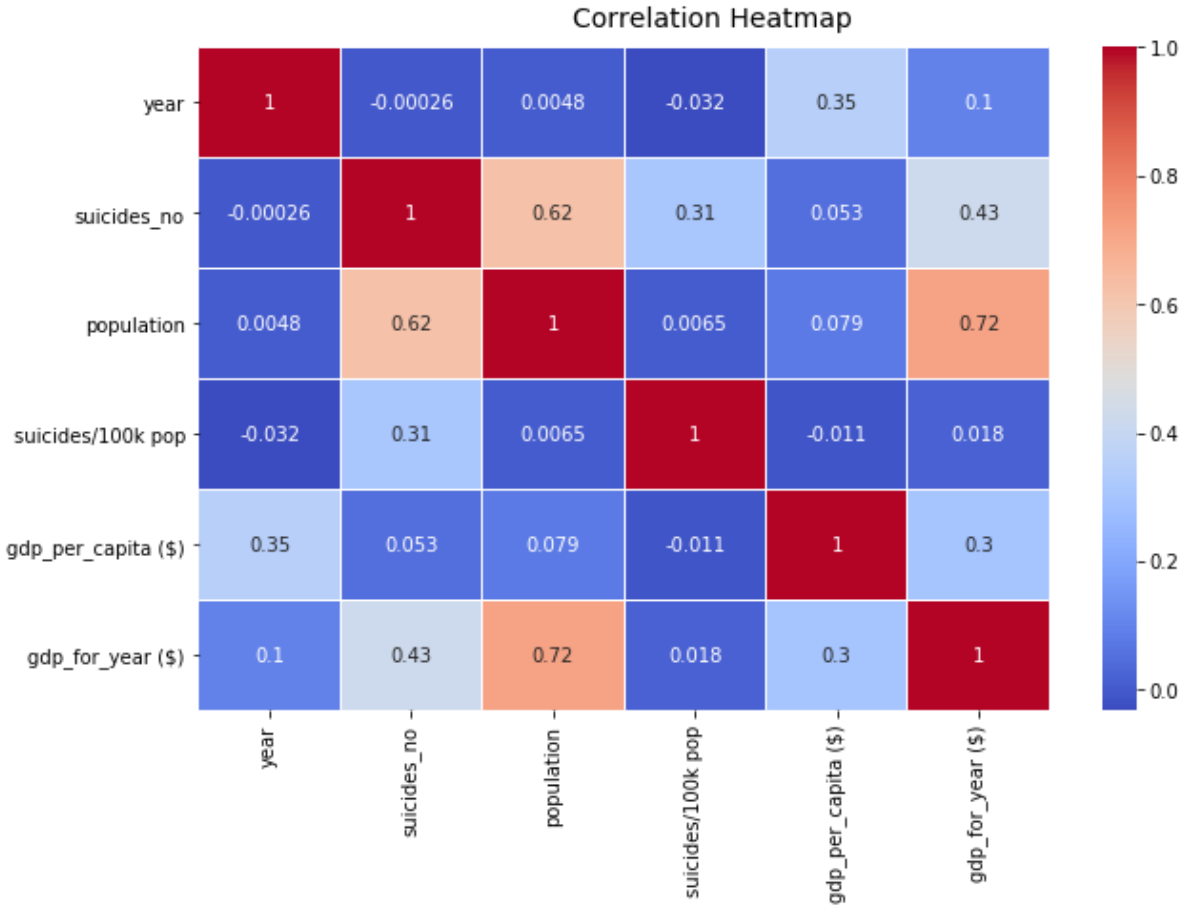
Out[42]:

	generation	year	suicides_no	suicides/100k pop	population
140	Silent	2011	23735	3007.83	112176540
47	Generation X	1985	16342	703.50	387618661
73	Generation X	2011	83676	2107.62	522020585

In [43]:
```
plt.figure(figsize=(15,8));
sns.pointplot(x='year',y=By_generation['suicides/100k pop'].rolling(window=3).
mean(),data=By_generation[By_generation['generation']!='G.I. Generation'], hue
='generation' ,color = 'orange');
plt.title('Suicides per 100k population for various generations (excluding G.
I. Generation)');
plt.ylabel('Number of Suicides per one lac population');
```

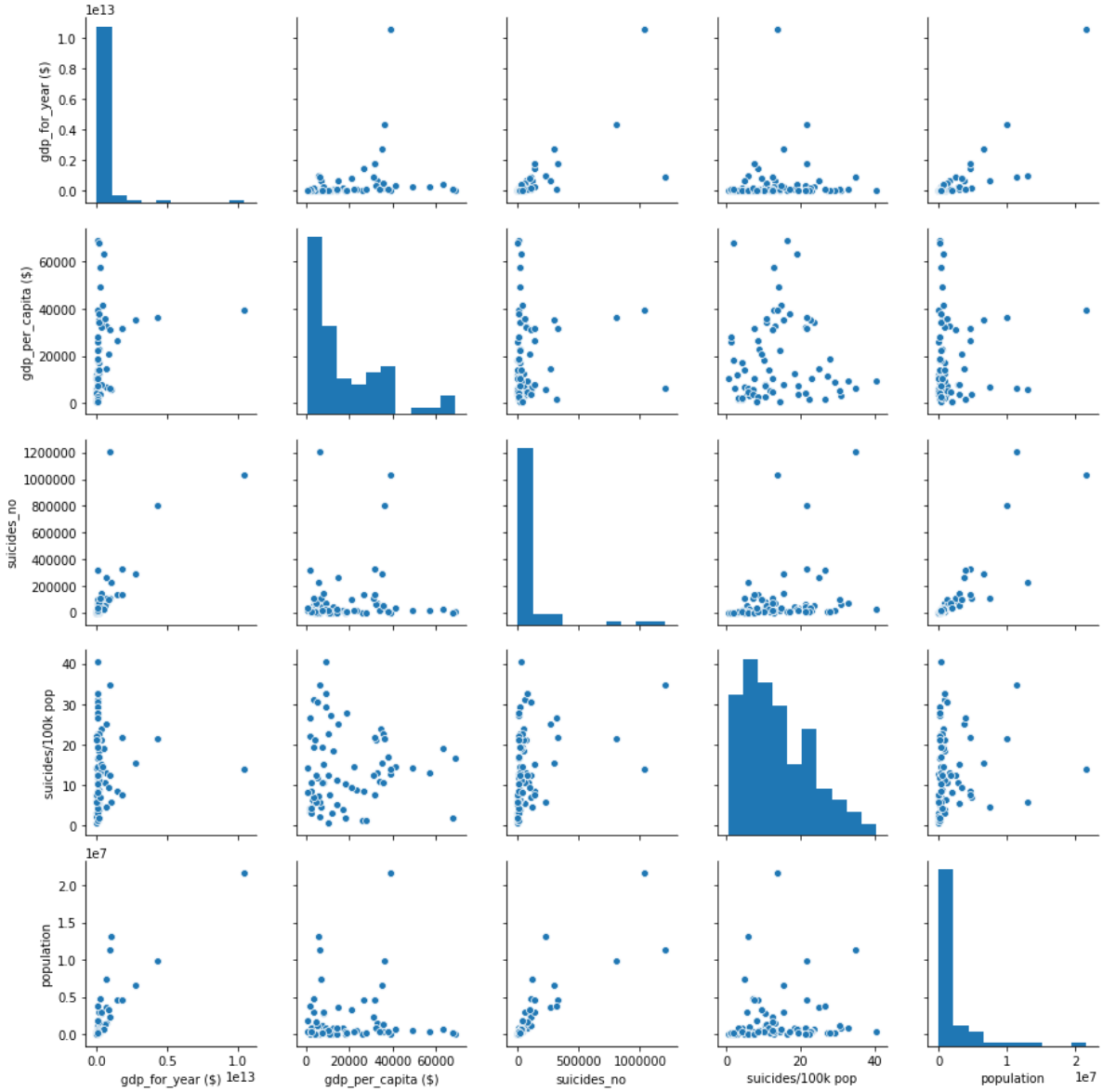


In [44]:
```
f, ax = plt.subplots(figsize=(10, 6))
corr = suicide.corr()
hm = sns.heatmap(corr, annot=True, ax=ax, cmap="coolwarm",linewidths=.05)
f.subplots_adjust(top=0.93)
t= f.suptitle('Correlation Heatmap', fontsize=14)
```



In [45]:
```
sns.pairplot(By_Country)
```

Out[45]: <seaborn.axisgrid.PairGrid at 0x10013a10>



In [ ]: