In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

In [5]:
```
covid=pd.read_csv("D:\\covid\\2020\\owid-covid-data.csv")
covid.head(3)
```

Out[5]:

	iso_code	continent	location	date	total_cases	new_cases	total_deaths	new_deaths	total_cases_per_million	new_cases_per_million	...	aged_70_older
0	AFG	Asia	Afghanistan	31-12-19	0	0	0	0	0.0	0.0	...	1.337
1	AFG	Asia	Afghanistan	01-01-20	0	0	0	0	0.0	0.0	...	1.337
2	AFG	Asia	Afghanistan	02-01-20	0	0	0	0	0.0	0.0	...	1.337

3 rows × 34 columns

In [6]:
```
type(covid)
```

Out[6]: pandas.core.frame.DataFrame

In [7]:
```
covid.tail(3)
```

Out[7]:

	iso_code	continent	location	date	total_cases	new_cases	total_deaths	new_deaths	total_cases_per_million	new_cases_per_million	...	aged_70_
30008	0	0	International	01-03-20	705	0	6	0	0.0	0.0	...	
30009	0	0	International	02-03-20	705	0	6	0	0.0	0.0	...	
30010	0	0	International	10-03-20	696	-9	7	1	0.0	0.0	...	

3 rows × 34 columns

In [8]:
```
covid.shape
```

Out[8]: (30011, 34)

In [9]:
```
describe=covid.describe()
describe
```

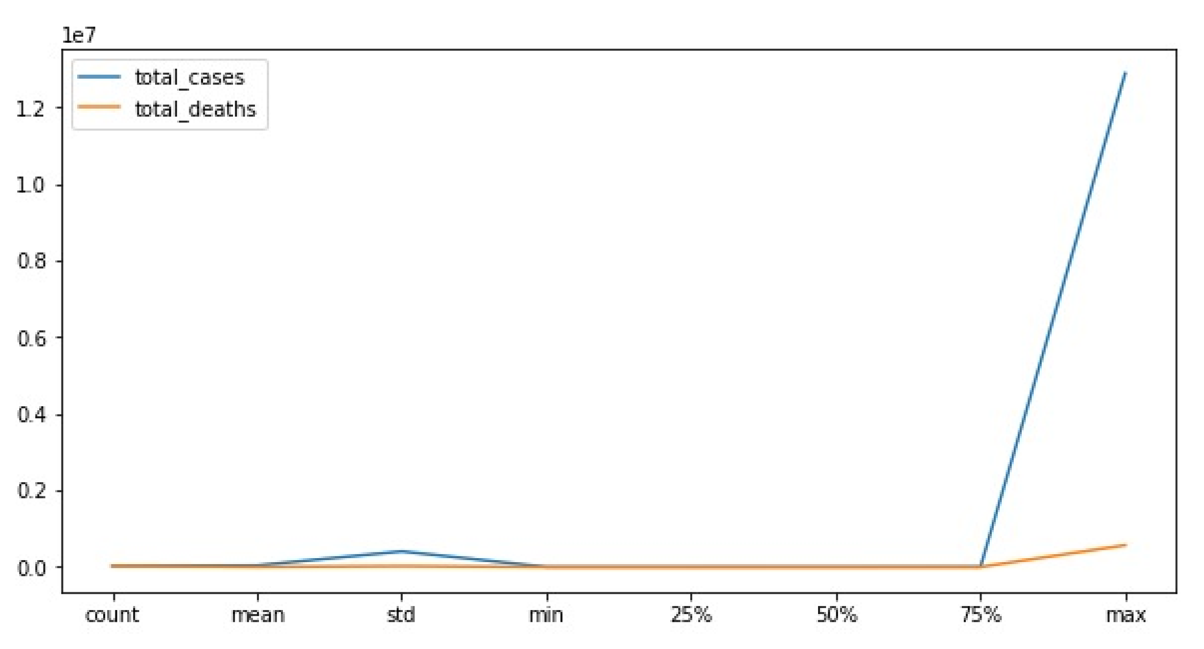
Out[9]:

	total_cases	new_cases	total_deaths	new_deaths	total_cases_per_million	new_cases_per_million	total_deaths_per_million	new_deaths_per_mil
count	3.001100e+04	30011.000000	30011.000000	30011.000000	30011.000000	30011.000000	30011.000000	30011.000
mean	4.023606e+04	858.886475	2268.698377	37.901836	925.527973	16.224623	36.147205	0.518
std	4.072162e+05	7792.183039	21960.423594	335.589895	2348.150844	62.064814	117.069163	2.964
min	0.000000e+00	-29726.000000	0.000000	-1918.000000	0.000000	-437.881000	0.000000	-41.023
25%	1.600000e+01	0.000000	0.000000	0.000000	4.463500	0.000000	0.000000	0.000
50%	3.180000e+02	4.000000	6.000000	0.000000	106.723000	0.561000	1.251000	0.000
75%	3.645500e+03	81.000000	81.000000	2.000000	694.388000	9.037500	15.523500	0.111
max	1.288802e+07	229759.000000	568736.000000	10489.000000	35958.293000	4944.376000	1237.551000	200.040

8 rows × 29 columns

In [10]:
```
#line plot
plt.figure(figsize=(10,5))
describe["total_cases"].plot(label="total_cases")
describe["total_deaths"].plot(label="total_deaths")
plt.legend()
```

Out[10]: <matplotlib.legend.Legend at 0x4138ab0>



In [11]:
```
covid["location"].value_counts()
```

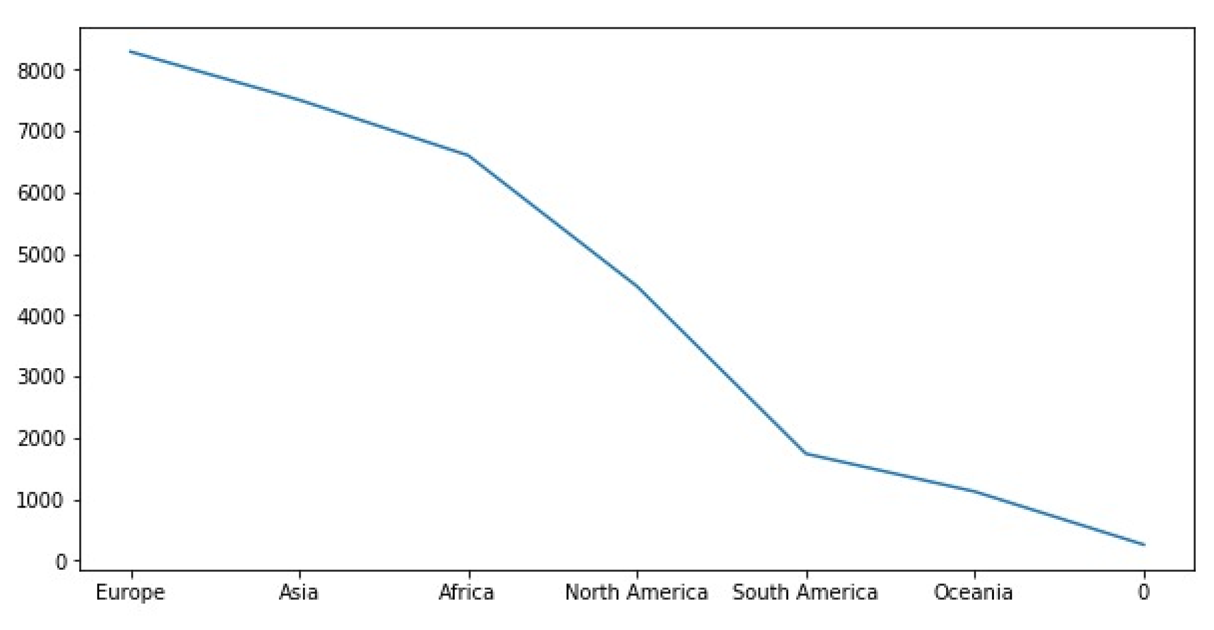
Out[11]:
```
Netherlands       196
Lithuania         196
France            196
Estonia           196
Czech Republic    196
                 ...
Tajikistan         74
Comoros            73
International      64
Lesotho            60
Hong Kong          13
Name: location, Length: 212, dtype: int64
```

In [12]:
```
covid["continent"].value_counts()
```

Out[12]:
```
Europe           8297
Asia             7510
Africa           6606
North America    4471
South America    1742
Oceania          1125
0                 260
Name: continent, dtype: int64
```

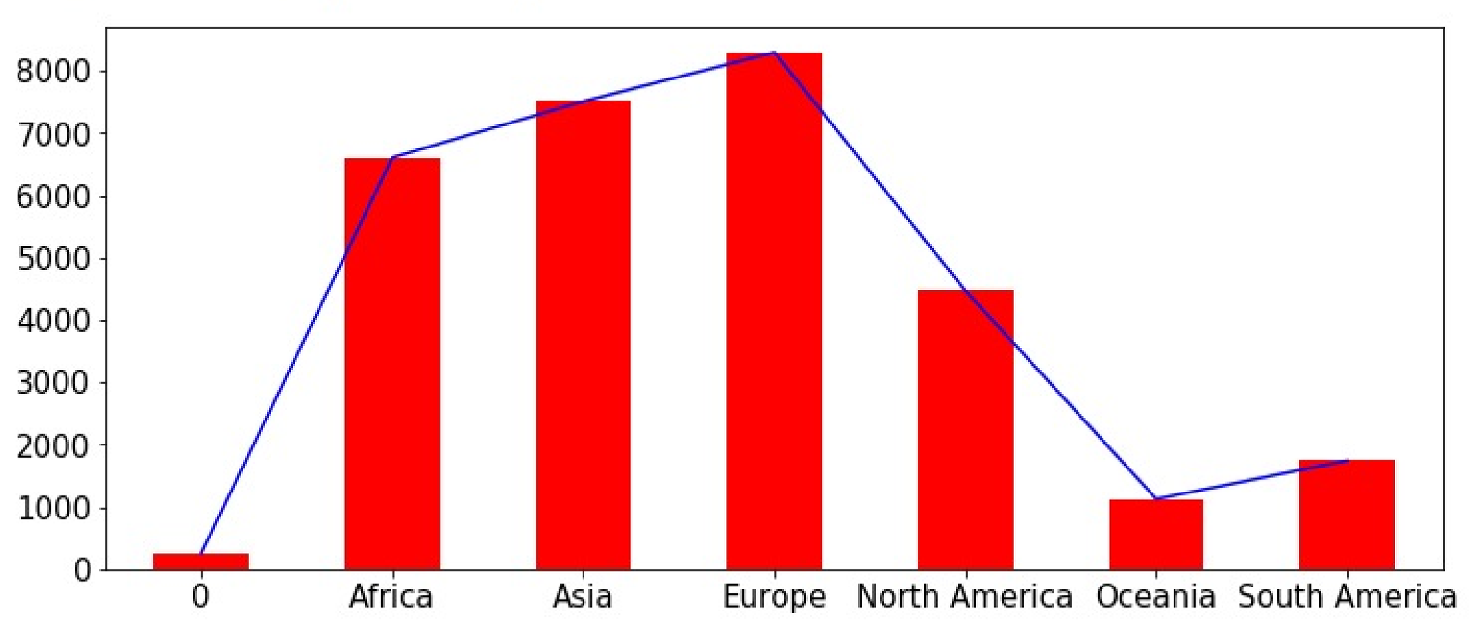
In [13]:
```
plt.figure(figsize=(10,5))
covid["continent"].value_counts().plot()
```

Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0xe4650f0>



In [14]:
```
data=covid["continent"].value_counts().sort_index()
plt.figure(figsize=(12,5))
data.plot(kind="bar", color="red",fontsize=15 )
data.plot(color="blue",fontsize=15 )
```

Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0xe4a3a30>



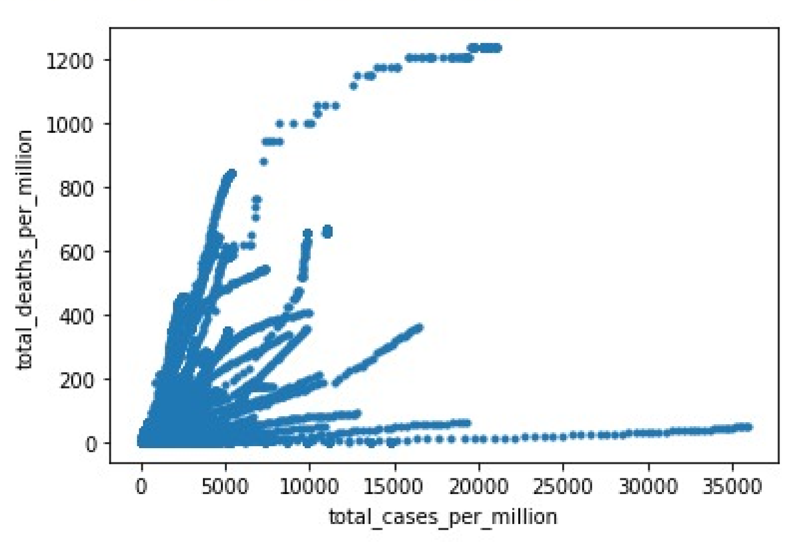
In [15]:
```
covid["total_deaths"].describe()
```

Out[15]:
```
count     30011.000000
mean       2268.698377
std       21960.423594
min           0.000000
25%           0.000000
50%           6.000000
75%          81.000000
max      568736.000000
Name: total_deaths, dtype: float64
```

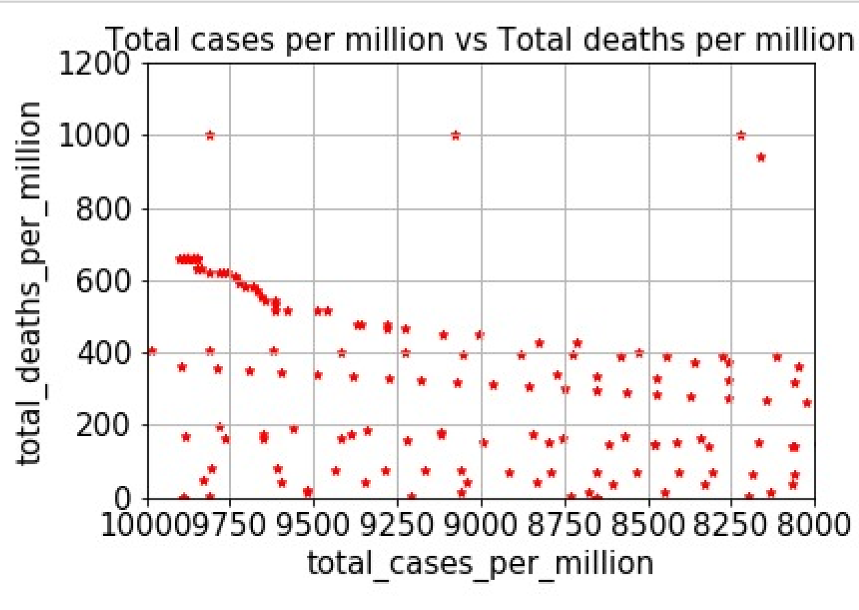
In [16]:
```
#Scatter plot
plt.figure(figsize=(15,15))
covid.plot.scatter(x="total_cases_per_million", y="total_deaths_per_million", s=10)
```

Out[16]: <matplotlib.axes._subplots.AxesSubplot at 0xe4f81b0>

<Figure size 1080x1080 with 0 Axes>

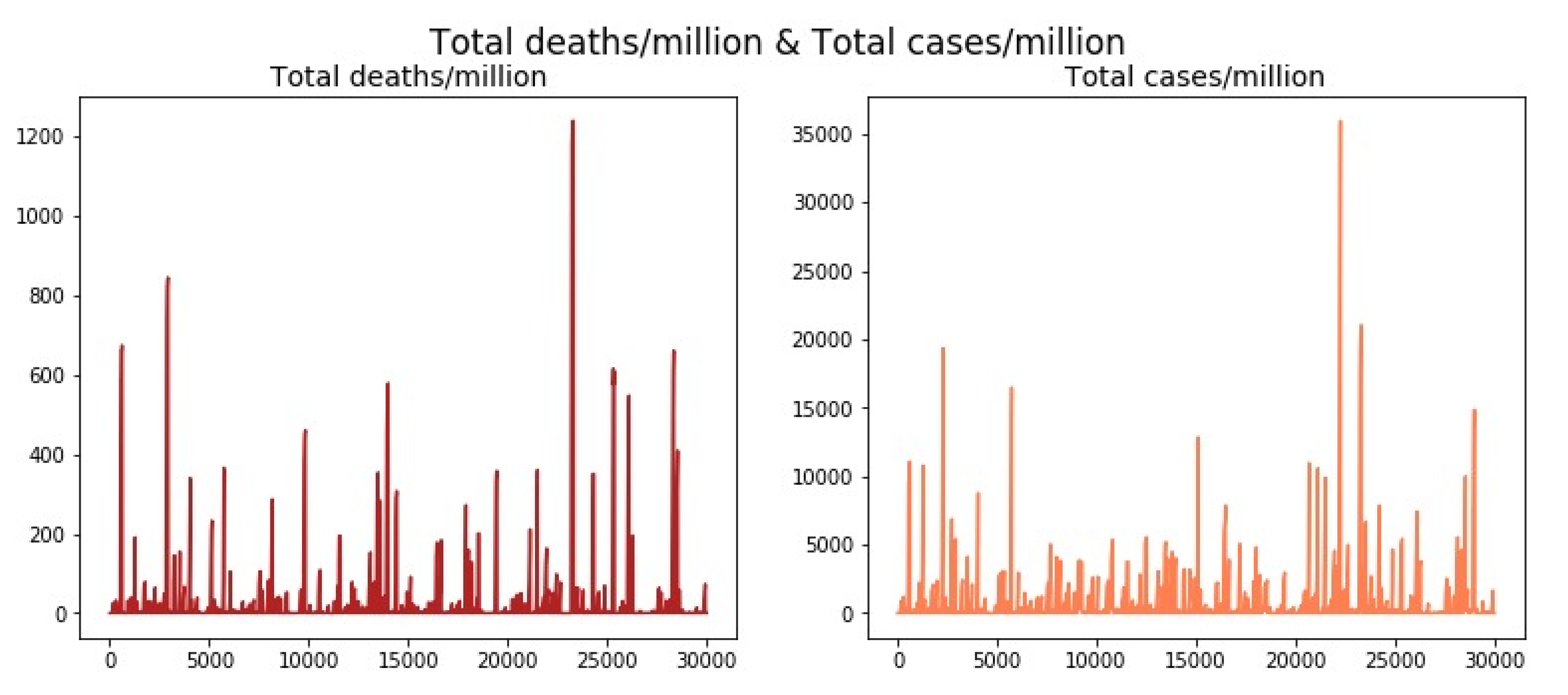


In [17]:
```
#cleared scatter plot
covid.plot.scatter(x="total_cases_per_million", y="total_deaths_per_million", s=20, color="red", fontsize=15, marker="*")
plt.rcParams['figure.figsize']=(10,5)
plt.xlabel('total_cases_per_million',fontsize=15)
plt.ylabel('total_deaths_per_million',fontsize=15)
plt.title("Total cases per million vs Total deaths per million", fontsize=15)
plt.xlim(10000,8000)
plt.ylim(0,1200)
plt.grid()
```



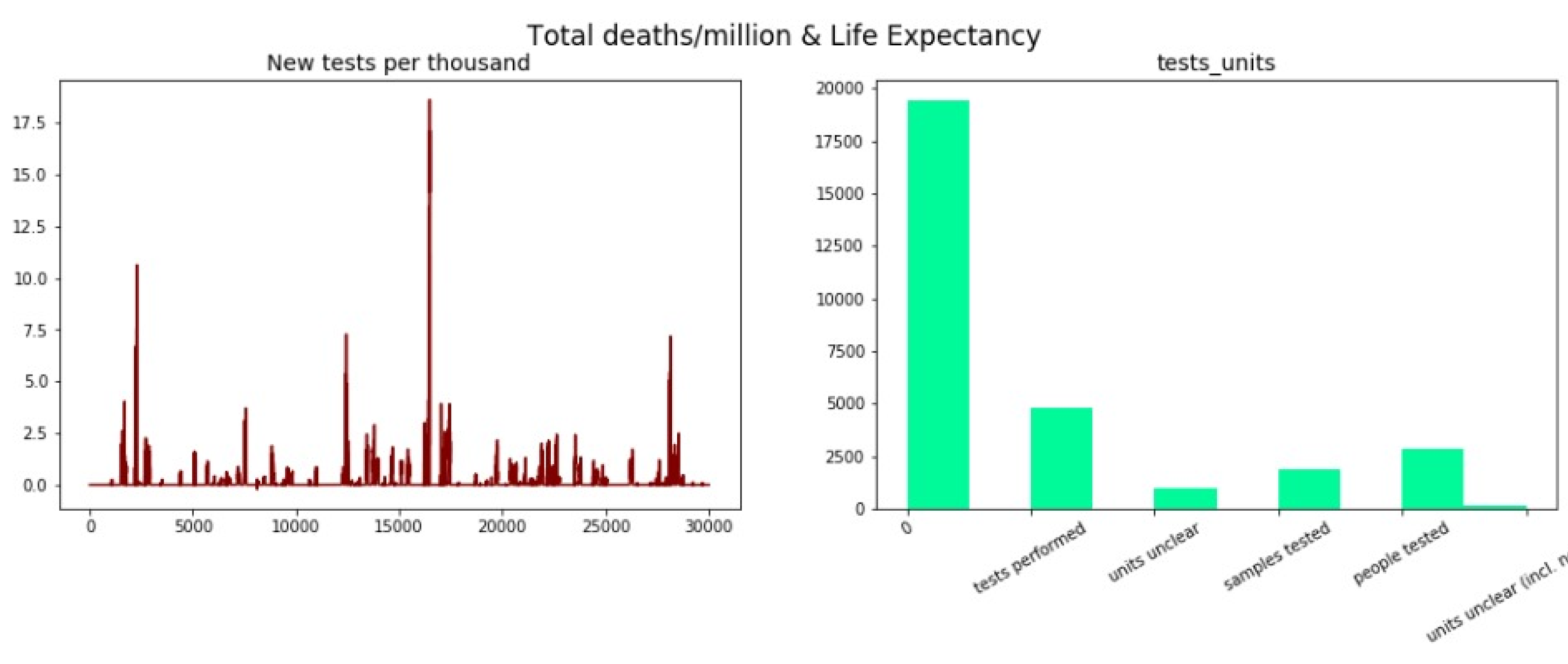
In [18]:
```
#subplot between total_deaths_per_million and total_cases_per_million
plt.figure(figsize=(13,5))
plt.subplot(1,2,1)
plt.plot(covid["total_deaths_per_million"],color="firebrick")
plt.title("Total deaths/million",fontsize=14)
plt.subplot(1,2,2)
plt.plot(covid["total_cases_per_million"],color="coral")
plt.title("Total cases/million",fontsize=14)
plt.suptitle("Total deaths/million & Total cases/million", fontsize=17)
```

Out[18]: Text(0.5, 0.98, 'Total deaths/million & Total cases/million')

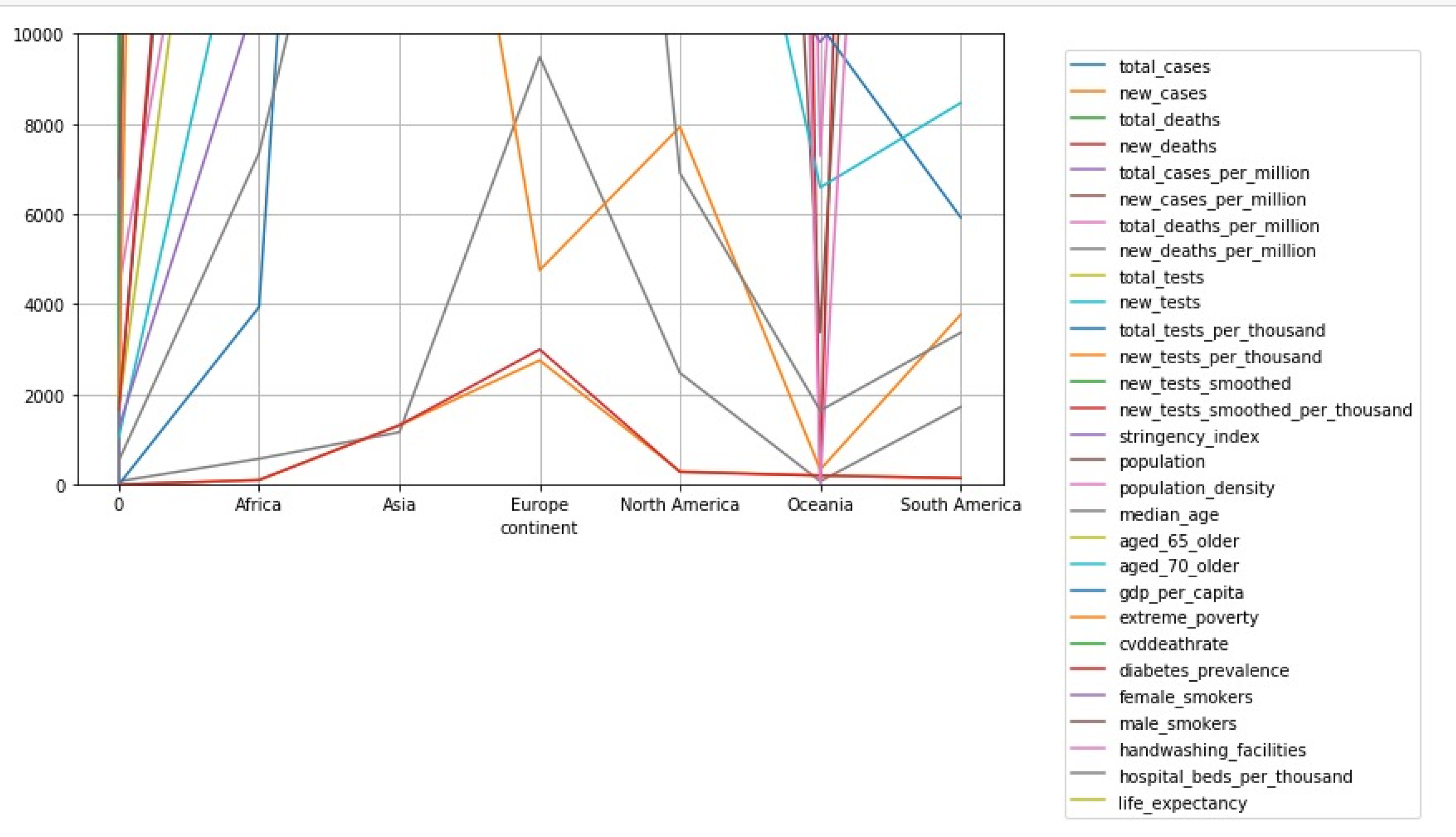


In [19]:
```
#Subplot between new_tests_per_thousand and tests_units
plt.figure(figsize=(17,5))
plt.subplot(1,2,1)
plt.plot(covid["new_tests_per_thousand"],color="maroon")
plt.title("New tests per thousand",fontsize=14)
plt.subplot(1,2,2)
plt.hist(covid["tests_units"],color="mediumspringgreen",width=0.5)
plt.xticks(rotation=30)
plt.title("tests_units",fontsize=14)
plt.suptitle("Total deaths/million & Life Expectancy", fontsize=17)
```

Out[19]: Text(0.5, 0.98, 'Total deaths/million & Life Expectancy')



In [20]:
```
#linear plot
covid.groupby('continent').sum().plot(fontsize=10)
plt.rcParams['figure.figsize']=(10,10)
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=1)
plt.ylim(0,10000)
plt.grid()
```



In [21]:
```
#horizontal plot
plt.figure(figsize=(10,7))
covid["total_tests"].describe().plot(kind="barh", label="total_tests",color="red",fontsize=15)
covid["new_tests"].describe().plot(kind="barh",label="new_tests",color="orange")
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', borderaxespad=1)
plt.xticks(rotation=30)
plt.xlim(0.0,400000)
```

Out[21]: (0.0, 400000)