Лабораторная работа 1 Елизаров Олег

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
data = pd.read_csv('archive/full_grouped.csv')
data
```

Out[2]:

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered
0	2020-01-22	Afghanistan	0	0	0	0	0	0	0
1	2020-01-22	Albania	0	0	0	0	0	0	0
2	2020-01-22	Algeria	0	0	0	0	0	0	0
3	2020-01-22	Andorra	0	0	0	0	0	0	0
4	2020-01-22	Angola	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...
35151	2020-07-27	West Bank and Gaza	10621	78	3752	6791	152	2	0
35152	2020-07-27	Western Sahara	10	1	8	1	0	0	0
35153	2020-07-27	Yemen	1691	483	833	375	10	4	36
35154	2020-07-27	Zambia	4552	140	2815	1597	71	1	465
35155	2020-07-27	Zimbabwe	2704	36	542	2126	192	2	24

35156 rows × 10 columns

In [3]:

```
data.shape
```

Out[3]:

```
(35156, 10)
```

In [4]:

```
data.isnull().sum()
```

Out[4]:

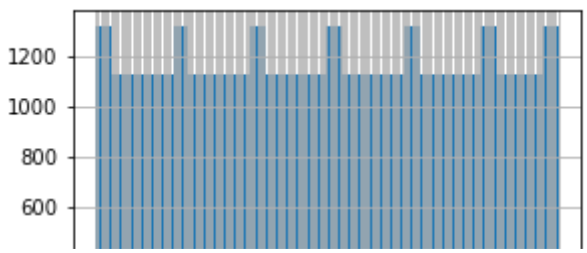
```
Date              0
Country/Region    0
Confirmed         0
Deaths            0
Recovered         0
Active            0
New cases         0
New deaths        0
New recovered     0
WHO Region        0
dtype: int64
```

In [5]:

```
data.dtypes
types = []
for col in data:
    if data[col].dtype== "int64":
        types.append(col)
```

In [6]:

```
# возьмем часть кода из лекции
def diagnostic_plots(data,col):
    # stars.hist(bins=30)
    fig, ax = plt.subplots(figsize=(10,7))
    # гистограмма
    plt.subplot(2, 2, 1)
    data[col].hist(bins=30)
for col in data:
    diagnostic_plots(data,col)
```



In [7]:

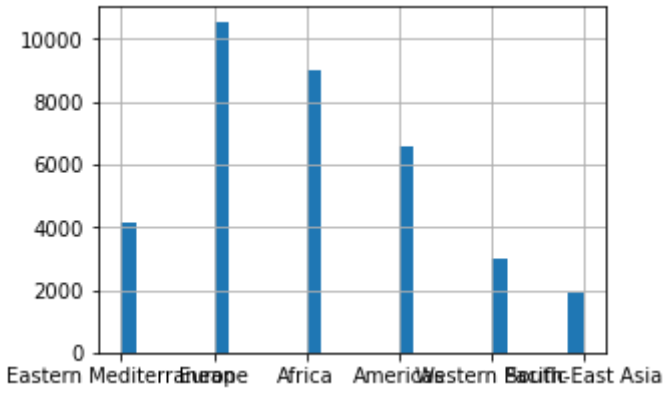
```
for col in data:
    print(col,len(data[col].unique()))

# Видим, что было ошибкой делать гистограммы для всех колонок
```

```
Date 188
Country/Region 187
Confirmed 10732
Deaths 3598
Recovered 7649
Active 8440
New cases 2800
New deaths 715
New recovered 2276
WHO Region 6
```

In [8]:

```
diagnostic_plots(data,"WHO Region")    # 1
```



Давайте теперь возьмем суммарные и средние заболевания по странам и регионам

In [9]:

```
data.groupby("WHO Region").mean().sort_values(by="Confirmed")
```

Out[9]:

WHO Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths
Africa	2414.874446	48.756427	1240.439938	1125.678081	80.179521	1.354499
Western Pacific	8768.088763	309.983378	6270.595080	2187.510306	97.034242	2.736702
Eastern Mediterranean	17911.724371	465.190764	11617.674807	5828.858801	360.457930	9.269584
Europe	23639.797967	1830.455927	11702.324753	10107.017287	315.057751	20.055471
South-East Asia	29318.279255	775.603191	15973.578191	12569.097872	976.221277	21.994149
Americas	61133.920061	2942.141641	23870.736170	34321.042249	1343.838146	52.086930

In [10]:

```
data.groupby("WHO Region").sum().sort_values(by="Confirmed")
```

Out[10]:

WHO Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered
Africa	21791827	439978	11193730	10158119	723540	12223	440645
Western Pacific	26374411	932430	18861950	6580031	291879	8232	206742
South-East Asia	55118365	1458134	30030327	23629904	1835296	41349	1156933
Eastern Mediterranean	74082892	1924029	48050703	24108160	1490854	38339	1201400
Europe	248879793	19271040	123202075	106406678	3316928	211144	1993723
Americas	402261194	19359292	157069444	225832458	8842455	342732	4468616

In [11]:

```
data.groupby("Country/Region").mean().sort_values(by="Confirmed")
```

Out[11]:

Country/Region	Confirmed	Deaths	Recovered	Active	New cases	d
Western Sahara	4.792553e+00	0.335106	3.446809	1.010638	0.053191	0.0
Papua New Guinea	6.303191e+00	0.010638	3.696809	2.595745	0.329787	0.0
Holy See	7.212766e+00	0.000000	3.946809	3.265957	0.063830	0.0
Greenland	8.015957e+00	0.000000	7.297872	0.718085	0.074468	0.0
Saint Kitts and Nevis	9.425532e+00	0.000000	6.888298	2.537234	0.090426	0.0
...	...	...	...	...	...	
Spain	1.457662e+05	16133.138298	80285.015957	49348.042553	1504.398936	151.2
India	2.174652e+05	5913.994681	126509.148936	85042.090426	7872.728723	177.7
Russia	2.415341e+05	3294.601064	133619.404255	104620.095745	4344.042553	70.9
Brazil	4.761966e+05	20946.989362	289855.707447	165393.936170	12991.356383	466.0
US	1.193330e+06	58571.335106	299752.212766	835005.962766	22820.521277	787.2

187 rows × 7 columns

In [12]:

```
data.groupby("Country/Region").sum().sort_values(by="Confirmed")
```

Out[12]:

Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered
Western Sahara	901	63	648	190	10	1	8
Papua New Guinea	1185	2	695	488	62	0	11
Holy See	1356	0	742	614	12	0	12
Greenland	1507	0	1372	135	14	0	13
Saint Kitts and Nevis	1772	0	1295	477	17	0	15
...	...	...	...	...	...	...	...
Spain	27404045	3033030	15093583	9277432	282827	28432	150376
India	40883464	1111831	23783720	15987913	1480073	33408	951166
Russia	45408411	619385	25120448	19668578	816680	13334	602249
Brazil	89524967	3938034	54492873	31094060	2442375	87618	1846641
US	224345948	11011411	56353416	156981121	4290258	148011	1325804

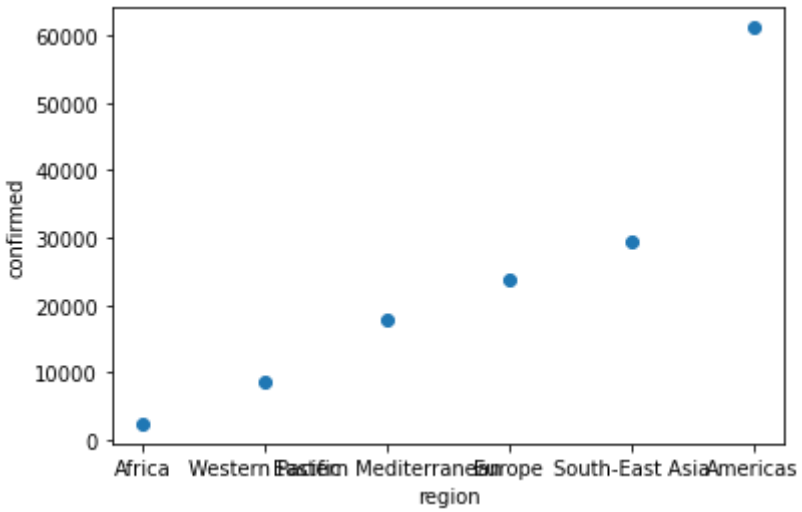
187 rows × 7 columns

In [13]:

```
graphData = data.groupby("WHO Region").mean().sort_values(by="Confirmed")
plt.xlabel('region')
plt.ylabel('confirmed')
plt.rcParams["figure.figsize"] = (10,10)
plt.scatter(x=graphData.index, y=graphData["Confirmed"]) #2
```

Out[13]:

```
<matplotlib.collections.PathCollection at 0x2375a42a640>
```



Тут можно увидеть регионы-лидеры по подтвержденным случаям ковида. Конечно, США вытягивает свой

регион в абсолютного чемпиона

In [14]:

```
graphData.index
```

Out[14]:

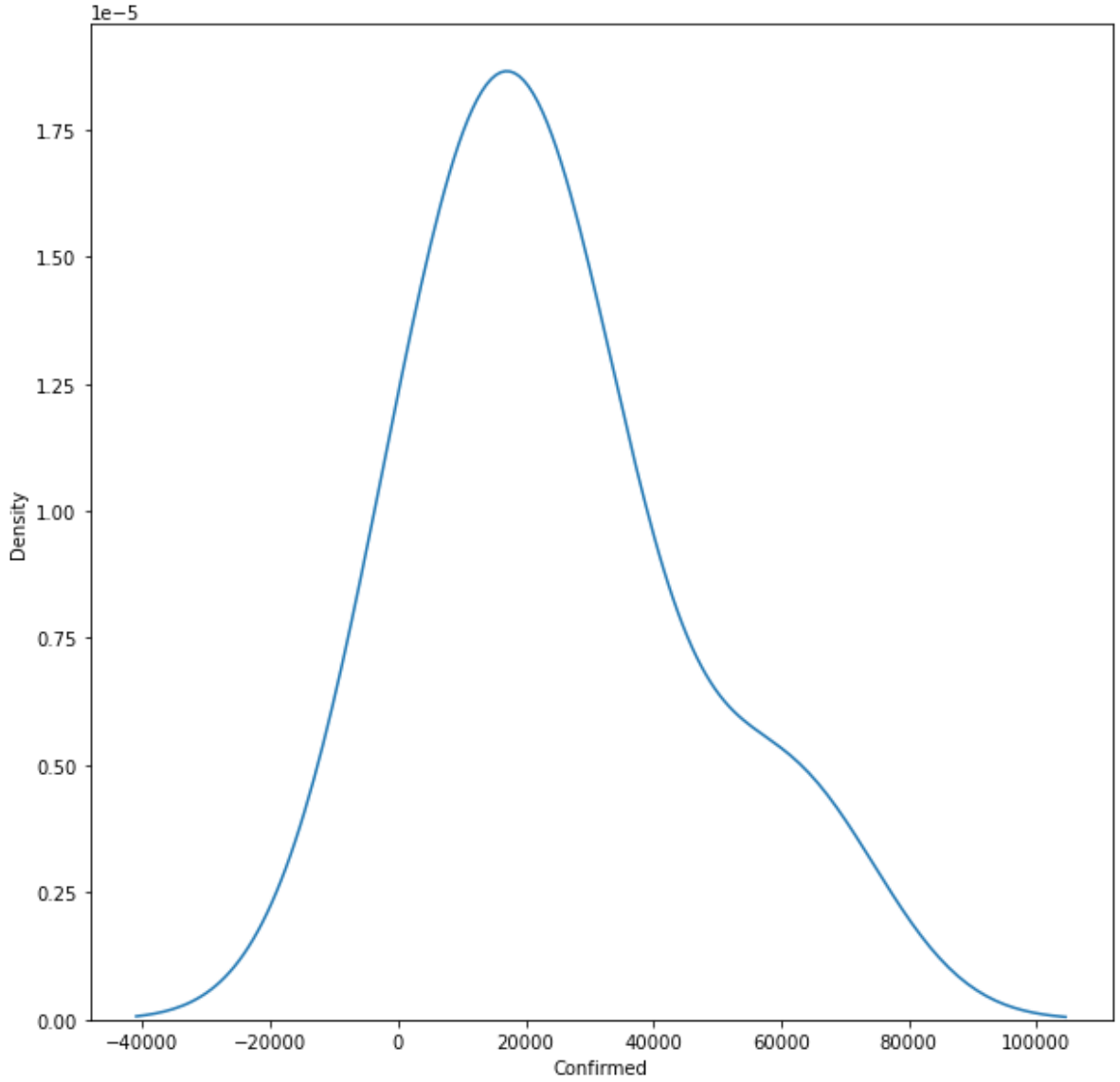
```
Index(['Africa', 'Western Pacific', 'Eastern Mediterranean', 'Europe',
       'South-East Asia', 'Americas'],
      dtype='object', name='WHO Region')
```

In [15]:

```
# Make default density plot
sns.kdeplot(graphData['Confirmed']) #3
```

Out[15]:

```
<AxesSubplot:xlabel='Confirmed', ylabel='Density'>
```

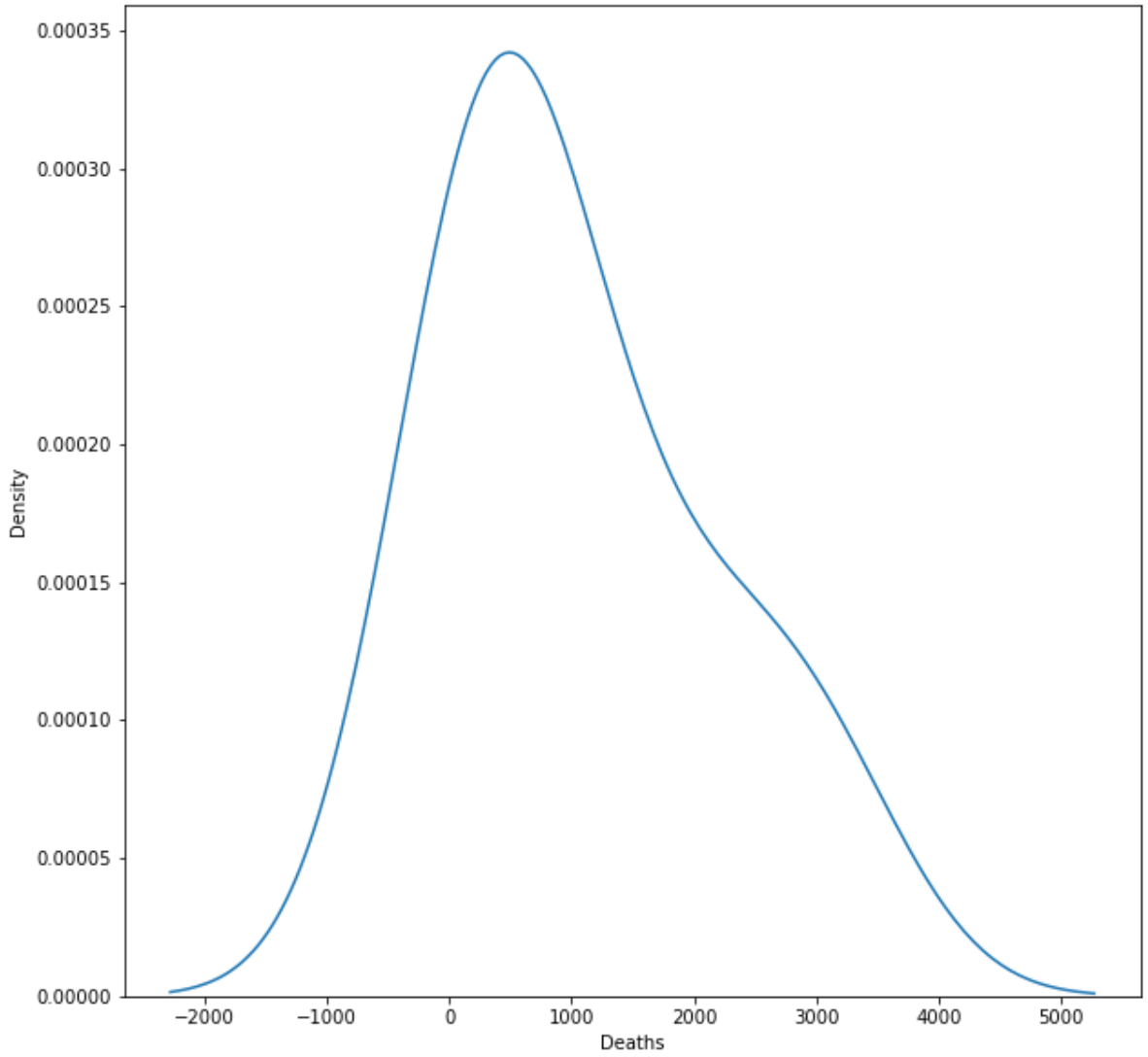


In [16]:

```
sns.kdeplot(graphData['Deaths'])
```

Out[16]:

```
<AxesSubplot:xlabel='Deaths', ylabel='Density'>
```



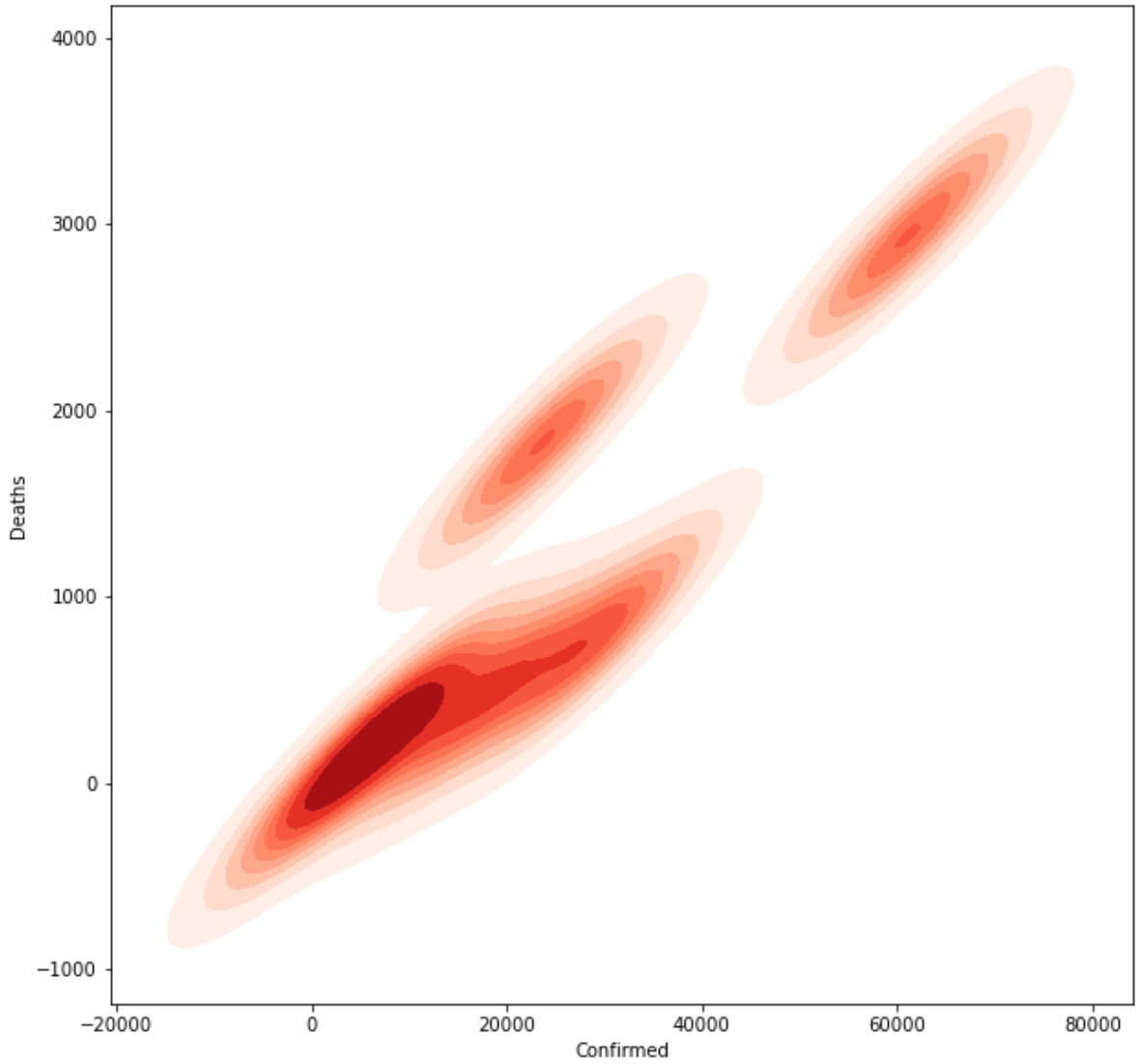
Можно ли верить полностью этим графикам ? Я бы был счастлив, если бы в мире было отрицательное число смертей, но ,увы это неточность графика.

In [17]:

```
sns.kdeplot(x=graphData["Confirmed"],y=graphData['Deaths'], cmap="Reds", shade=True, bw_adj
```

Out[17]:

```
<AxesSubplot:xlabel='Confirmed', ylabel='Deaths'>
```



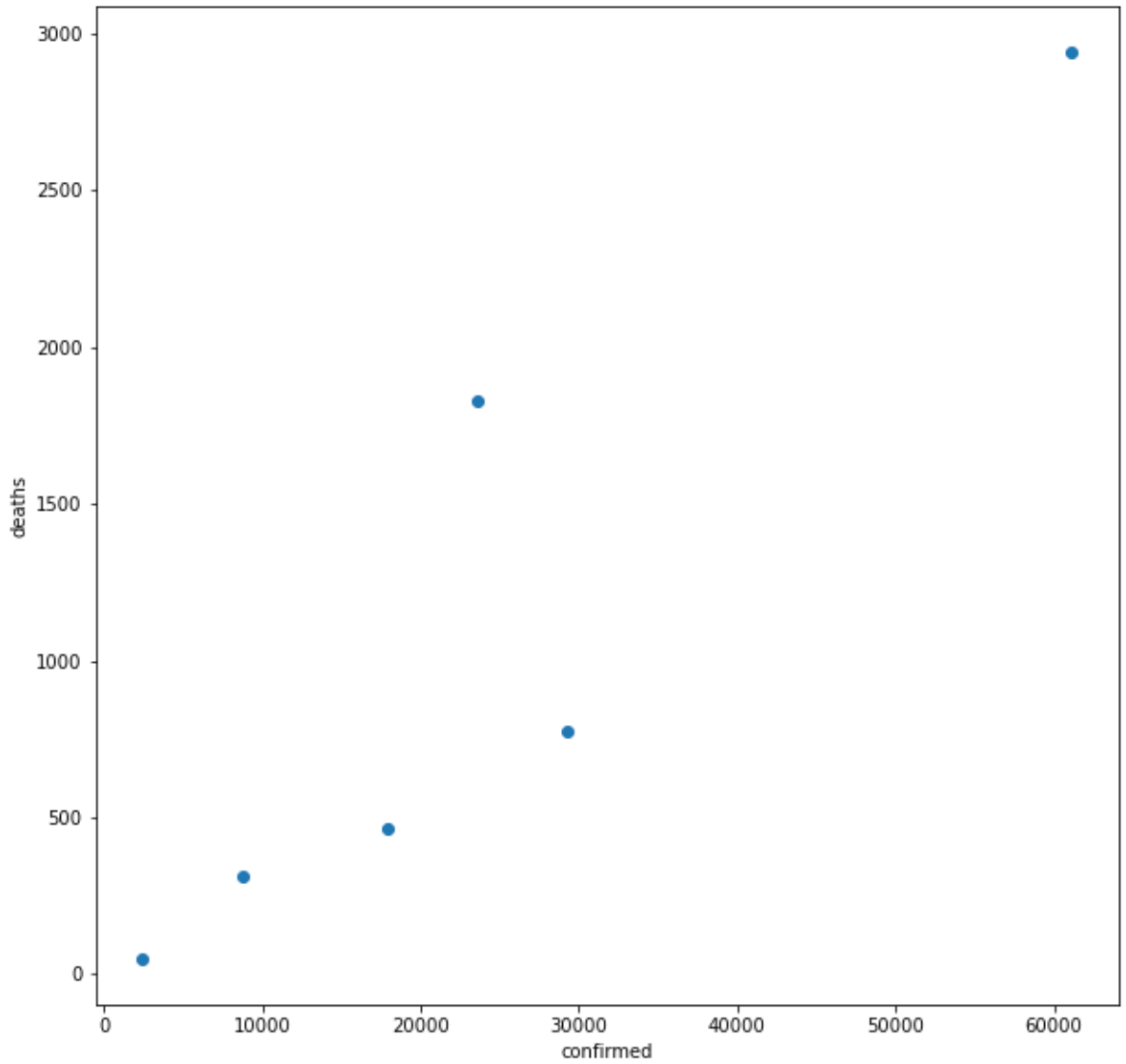
Я бы трактовал этот график так. Когда 0 заболеваний, тогда и 0 смертей При числе заболевших в 20000 - наиболее вероятное число погибших - 2000 При числе заболевших в 60000 - наиболее вероятное число погибших - 3000

In [18]:

```
plt.xlabel('confirmed')
plt.ylabel('deaths')
plt.rcParams["figure.figsize"] = (10,10)
plt.scatter(x=graphData["Confirmed"], y=graphData["Deaths"]) #5
```

Out[18]:

```
<matplotlib.collections.PathCollection at 0x2375b914850>
```

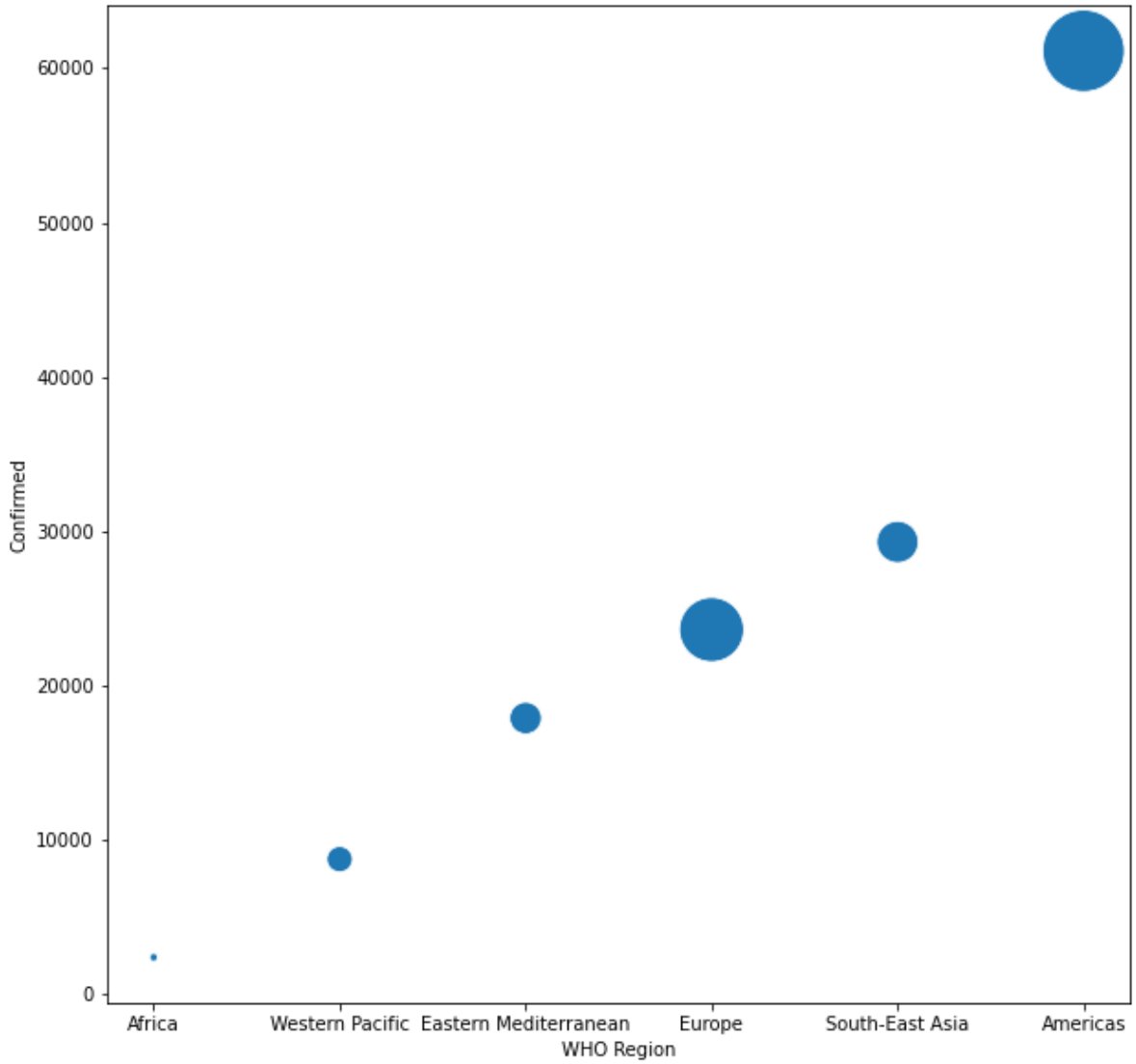


In [19]:

```
sns.scatterplot(data=graphData, x="WHO Region", y="Confirmed", size="Deaths", legend=False,
```

Out[19]:

```
<AxesSubplot:xlabel='WHO Region', ylabel='Confirmed'>
```



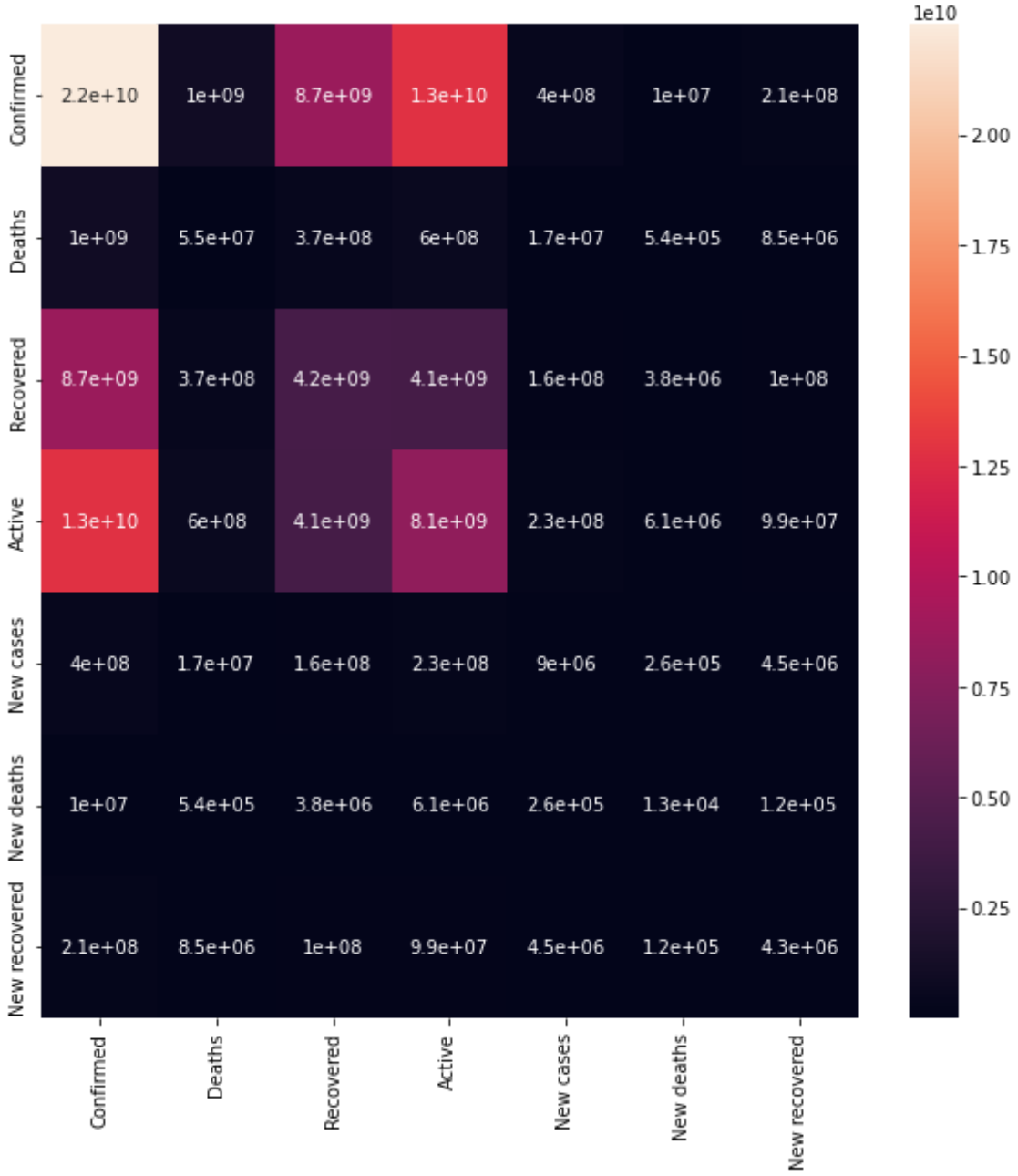
А тут мы можем увидеть интересную особенность Конечно, чем меньше заболевших - тем меньше смертность, но есть исключение. Европа имеет крайне большую смертность в сравнении в Юго-востоком Азии, хотя число заболевших больше. Причин может быть много,например, можно предположить , что в среднем жители Европы живут в более тепличных условиях,чем в Азии и их иммунитет менее развит. А иммунитет азиатов в состоянии противостоять ковиду, даже если человек заболе, но это лишь гипотеза.

In [32]:

```
sns.heatmap(data.cov(), annot = True) #7
# P.S. Была мысль, что могут быть сильные зависимости аттрибутов , но на деле не особо вышл
```

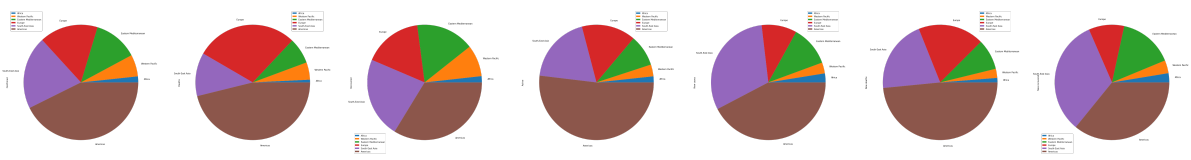
Out[32]:

<AxesSubplot:>



In [21]:

```
plot = graphData.plot.pie(subplots=True, figsize=(100, 100)) #8
```

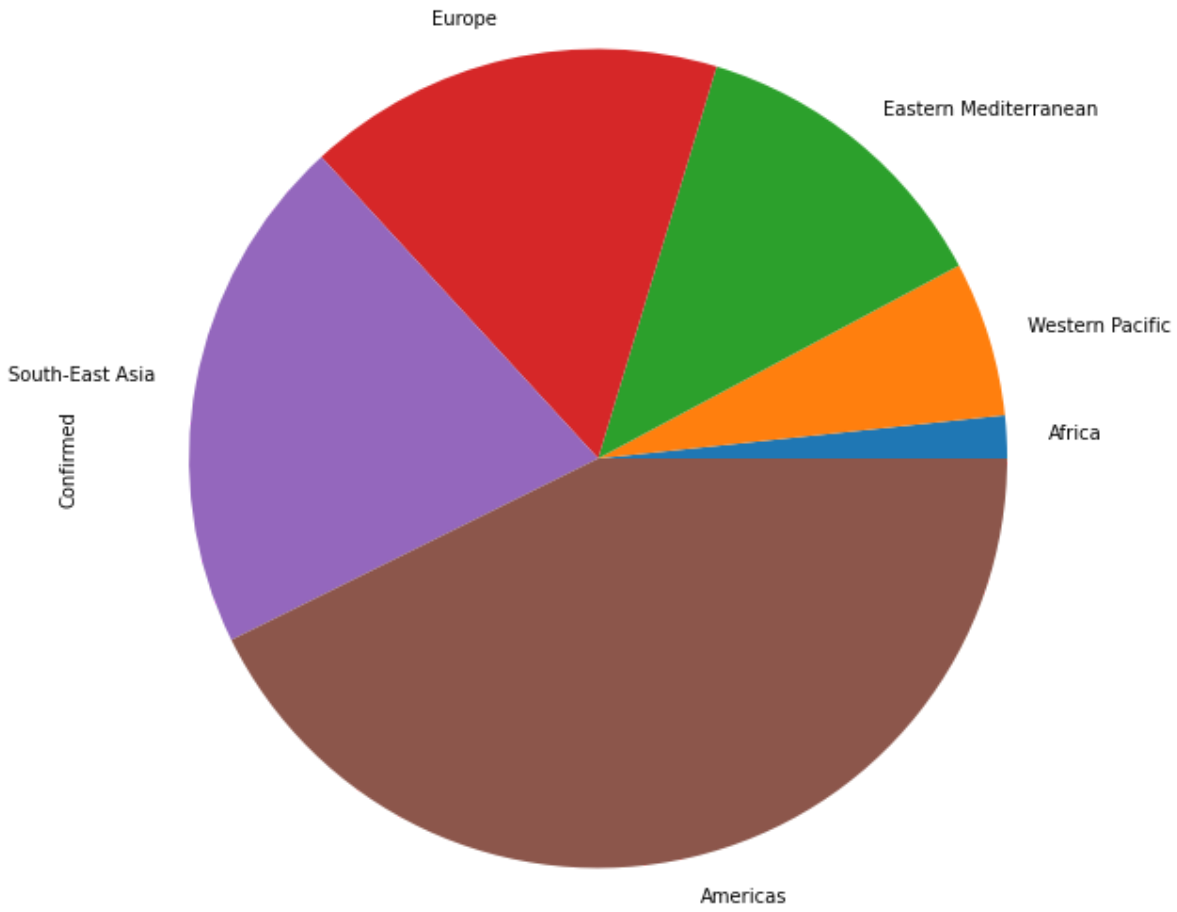


In [29]:

```
graphData["Confirmed"].plot.pie(subplots=True, figsize=(10, 10))
```

Out[29]:

```
array([<AxesSubplot:ylabel='Confirmed'>], dtype=object)
```

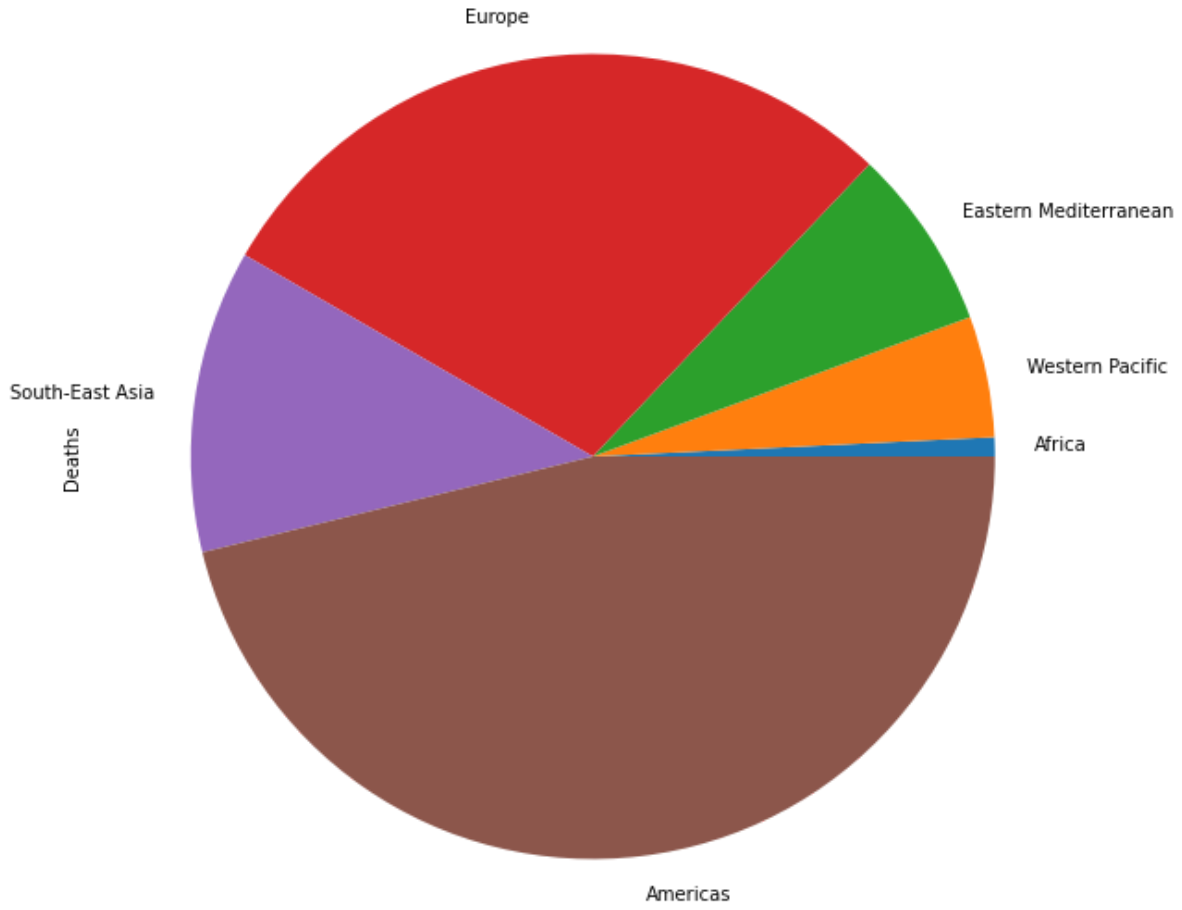


In [30]:

```
graphData["Deaths"].plot.pie(subplots=True, figsize=(10, 10))
```

Out[30]:

```
array([<AxesSubplot:ylabel='Deaths'>], dtype=object)
```



На пай-чартах еще раз можно для различных колонок посмотреть соотношение по регионам, тут опять лидирует регион, в котором США.