ویروس کرونا یا کووید- ۱۹یک نوع سندرم تنفسی حاد با عامل ویروسی از خانواده کرونا ویروسها میباشد که همهی کشورهای جهان را در مدت زمان کوتاهی درگیر کرده است. با توجه به شیوع و میزان مرگ و میر بالای این بیماری و از سوی دیگر، احتمال اوج مجدد کووید- ۱۹خصوصاً به دلیل نبود درمان اختصاصی، آشنایی و بررسی اطلاعات مربوط به ویروس کووید- ۱۹اهمیت زیادی دارد

۱ پیش پردازش

پیش پردازش، یکی از مهمترین گامها در پروژههای داده کاوی است. رویکردهای مختلفی در زمینه ی مدیریت داده های گم شده و تبدیل داده ها به فرمتهای دیگر مورد استفاده قرار میگیرد و انتخاب دقیق این رویکردها تأثیر مستقیمی در کیفیت نتایج نهایی دارد؛ لذا همواره میبایست بهترین رویکرد را شناسایی و اعمال نمود.

۱-۱ تعداد داده های گم شده در هر ویژگی را مشخص کنید. سپس، با ذکر دلیل، رویکرد مورد استفاده خود را برای پر کردن داده های گم شده در هر ستون مشخص کرده و اقدام به تکمیل داده های گم شده کنید.

Import Libraries

In [76]:

```
import numpy as np
import pandas as pd
import requests

#for visualization
import matplotlib.pyplot as plt
from matplotlib import dates as mdates
import missingno as msno

from persiantools.jdatetime import JalaliDate
import datetime
import jalali_pandas
from tqdm import tqdm

pd.options.mode.chained_assignment = None  # default='warn'
```

Load CSV File

In [77]:

```
file = "CA1_Dataset.csv"
df = pd.read_csv(file)
df
```

Out[77]:

	iso_code	continent	location	date	total_cases	new_cases
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0
...	...	...	...	...	...	..
165631	ZWE	Africa	Zimbabwe	2022-02-26	235803.0	336.0
165632	ZWE	Africa	Zimbabwe	2022-02-27	235803.0	0.0
165633	ZWE	Africa	Zimbabwe	2022-02-28	236380.0	577.0
165634	ZWE	Africa	Zimbabwe	2022-03-01	236871.0	491.0
165635	ZWE	Africa	Zimbabwe	2022-03-02	237503.0	632.0

165636 rows × 67 columns

به ازای هر ویژگی تعداد مقادیر خالی را میشماریم

In [78]:

```
with pd.option_context("display.max_rows", df.shape[0]+1):
    print(df.isna().sum())
```

```
iso_code                                  0
continent                              9917
location                                  0
date                                      0
total_cases                            3030
new_cases                              3172
new_cases_smoothed                     5156
total_deaths                          20843
new_deaths                            20803
new_deaths_smoothed                   22902
total_cases_per_million                3785
new_cases_per_million                  3927
new_cases_smoothed_per_million         5905
total_deaths_per_million              21585
new_deaths_per_million                21545
new_deaths_smoothed_per_million       23638
reproduction_rate                     40569
icu_patients                         142246
icu_patients_per_million             142246
hosp_patients                        141072
hosp_patients_per_million            141072
weekly_icu_admissions                160232
weekly_icu_admissions_per_million    160232
weekly_hosp_admissions               154759
weekly_hosp_admissions_per_million   154759
new_tests                             98630
total_tests                           96692
total_tests_per_thousand              96692
new_tests_per_thousand                98630
new_tests_smoothed                    81978
new_tests_smoothed_per_thousand       81978
positive_rate                         87046
tests_per_case                        87609
tests_units                           79655
total_vaccinations                   120658
people_vaccinated                    122844
people_fully_vaccinated              125608
total_boosters                       148296
new_vaccinations                     128384
new_vaccinations_smoothed             81524
total_vaccinations_per_hundred       120658
people_vaccinated_per_hundred        122844
people_fully_vaccinated_per_hundred  125608
total_boosters_per_hundred           148296
new_vaccinations_smoothed_per_million 81524
```

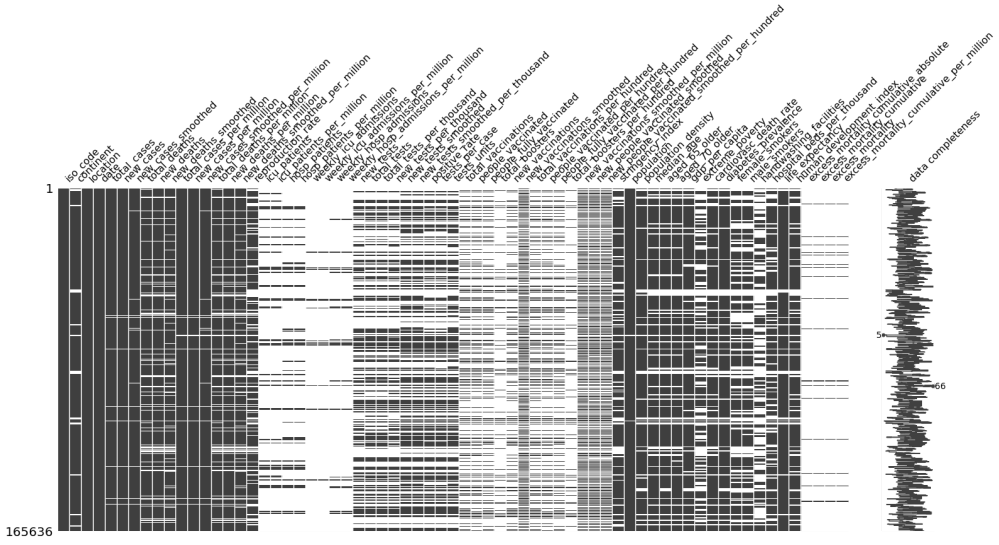
```
new_people_vaccinated_smoothed                  82815
new_people_vaccinated_smoothed_per_hundred      82815
stringency_index                                35774
population                                       1072
population_density                              18323
median_age                                      28378
aged_65_older                                   29866
aged_70_older                                   29114
gdp_per_capita                                  27708
extreme_poverty                                 74799
cardiovasc_death_rate                           29428
diabetes_prevalence                             22287
female_smokers                                  60027
male_smokers                                    61476
handwashing_facilities                          97352
hospital_beds_per_thousand                      42485
life_expectancy                                 11016
human_development_index                         29953
excess_mortality_cumulative_absolute           159940
excess_mortality_cumulative                    159940
excess_mortality                               159940
excess_mortality_cumulative_per_million        159940
dtype: int64
```

In [79]:

```
1 msno.matrix(df, labels=True)
```

Out[79]:

<AxesSubplot:>



ویژگی‌هایی که بیش از ۸۵٪ داده خالی دارند را حذف میکنیم. زیرا پر کردن آنها به ما داده پرت میدهد.

In [80]:

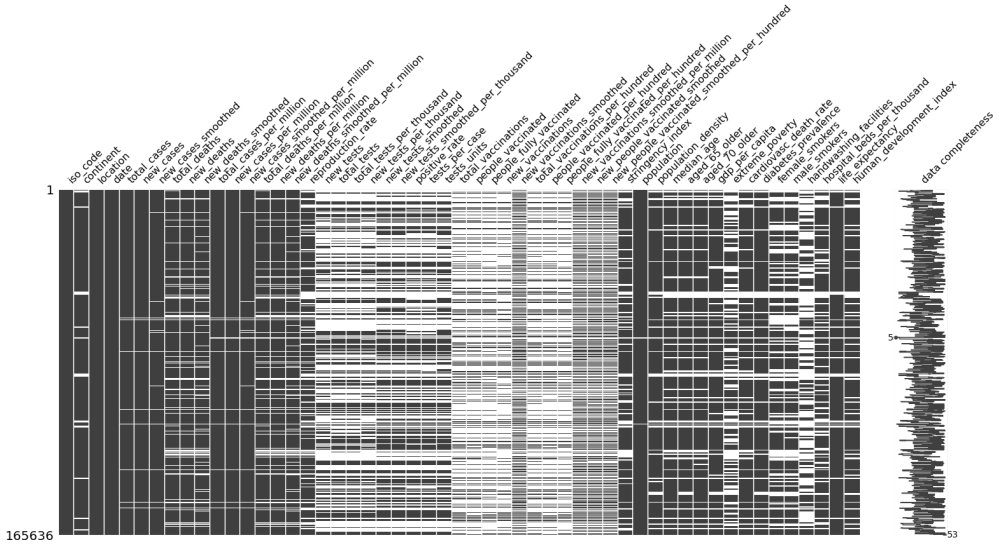
```
# Delete columns containing either 85% or more than 85% NaN Values
perc = 85.0
min_count =  int(((100-perc)/100)*df.shape[0] + 1)
mod_df = df.dropna( axis=1,
                thresh=min_count)

with pd.option_context("display.max_rows", mod_df.shape[0]+1):
    print(mod_df.isna().sum())

msno.matrix(mod_df, labels=True)
```

```
iso_code                                      0
continent                                  9917
location                                      0
date                                          0
total_cases                                3030
new_cases                                  3172
new_cases_smoothed                         5156
total_deaths                              20843
new_deaths                                20803
new_deaths_smoothed                       22902
total_cases_per_million                    3785
new_cases_per_million                      3927
new_cases_smoothed_per_million             5905
total_deaths_per_million                  21585
new_deaths_per_million                    21545
new_deaths_smoothed_per_million           23638
reproduction_rate                         40569
new_tests                                 98630
total_tests                               96692
total_tests_per_thousand                  96692
new_tests_per_thousand                    98630
new_tests_smoothed                        81978
new_tests_smoothed_per_thousand           81978
positive_rate                             87046
tests_per_case                            87609
tests_units                               79655
total_vaccinations                       120658
people_vaccinated                        122844
people_fully_vaccinated                  125608
new_vaccinations                         128384
new_vaccinations_smoothed                 81524
total_vaccinations_per_hundred           120658
people_vaccinated_per_hundred            122844
people_fully_vaccinated_per_hundred      125608
new_vaccinations_smoothed_per_million     81524
new_people_vaccinated_smoothed            82815
new_people_vaccinated_smoothed_per_hundred    82815
```

```
stringency_index                    35774
population                           1072
population_density                  18323
median_age                          28378
aged_65_older                       29866
aged_70_older                       29114
gdp_per_capita                      27708
extreme_poverty                     74799
cardiovasc_death_rate               29428
diabetes_prevalence                 22287
female_smokers                      60027
male_smokers                        61476
handwashing_facilities              97352
hospital_beds_per_thousand          42485
life_expectancy                     11016
human_development_index             29953
dtype: int64
```

Out[80]:

<AxesSubplot:>



In [81]:

```
1 mod_df.interpolate(method="linear", limit_direction="forward")
```

Out[81]:

	iso_code	continent	location	date	total_cases	new_cases
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0
...	...	...	...	...	...	..
165631	ZWE	Africa	Zimbabwe	2022-02-26	235803.0	336.0
165632	ZWE	Africa	Zimbabwe	2022-02-27	235803.0	0.0
165633	ZWE	Africa	Zimbabwe	2022-02-28	236380.0	577.0
165634	ZWE	Africa	Zimbabwe	2022-03-01	236871.0	491.0
165635	ZWE	Africa	Zimbabwe	2022-03-02	237503.0	632.0

165636 rows × 53 columns

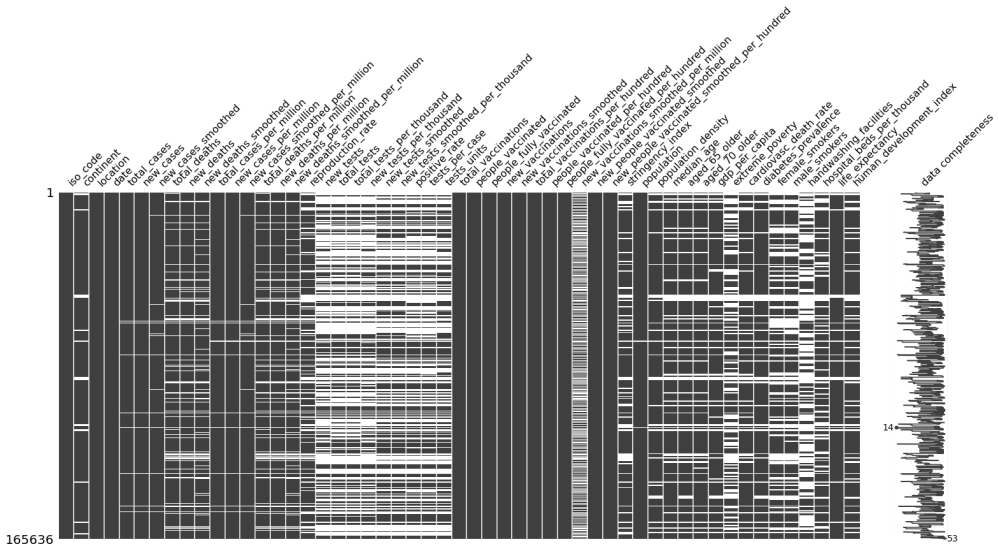
ویژگیهای مرتبط با واکسن را با روش backfill پر میکنیم زیرا این مقادیر در ابتدا تا کشف و توزیع واکسن صفر است.

In [82]:

```
mod_df["new_vaccinations"] = mod_df["new_vaccinations"].fillna(method="bfi
mod_df["total_vaccinations"]= mod_df["total_vaccinations"].fillna(method="
mod_df["people_vaccinated"] = mod_df["people_vaccinated"].fillna(method="b
mod_df["people_fully_vaccinated"]= mod_df["people_fully_vaccinated"].fillr
mod_df["new_vaccinations_smoothed"]= mod_df["new_vaccinations_smoothed"].f
mod_df["total_vaccinations_per_hundred"]= mod_df["total_vaccinations_per_h
mod_df["people_vaccinated_per_hundred"]= mod_df["people_vaccinated_per_hur
mod_df["people_fully_vaccinated_per_hundred"] = mod_df["people_fully_vacci
mod_df["new_people_vaccinated_smoothed"] = mod_df["new_people_vaccinated_s
mod_df["new_people_vaccinated_smoothed_per_hundred"]= mod_df["new_people_v



msno.matrix(mod_df, labels=True)
```

Out[82]:

<AxesSubplot:>



ستون‌های باقیمانده چون مقادیر آماری هستند مقادیر خالی آنها را به کمک میانگین‌گیری پر میکنیم. برای این کار ابتدا بر روی کشور میانگیری میکنیم و در مراحل بعد بر روی قاره و iso_code میانگین میگیریم و به جای مقادیر خالی باقیمانده میگذاریم.

In [83]:

```
# mod_df = mod_df.interpolate()
# mod_df.fillna(method ='bfill', inplace = True)
# mod_df.fillna(method ='ffill', inplace = True)


list_cols = list(mod_df.columns.values)
list_cols.remove('iso_code')
list_cols.remove('location')
list_cols.remove('date')
list_cols.remove('continent')
list_cols.remove('tests_units')

mean1 = mod_df.groupby('location').mean()
mean2 = mod_df.groupby('continent').mean()
mean3 = mod_df.groupby('iso_code').mean()


for col in tqdm(list_cols):
        mod_df[col] = mod_df.apply(
            lambda row: mean1[col][row["location"]] if pd.isna(row[col]) e
            axis=1
        )

for col in tqdm(list_cols):
        mod_df[col] = mod_df.apply(
            lambda row: mean2[col][row["continent"]] if pd.isna(row[col])
            axis=1
        )

for col in tqdm(list_cols):
        mod_df[col] = mod_df.apply(
            lambda row: mean3[col][row["iso_code"]] if pd.isna(row[col]) e
            axis=1
        )
```

```
100%|██████████████████████████████████████████████████
                    | 48/48 [01:58<00:00,  2.48s/it]
100%|██████████████████████████████████████████████████
                    | 48/48 [01:59<00:00,  2.48s/it]
100%|██████████████████████████████████████████████████
                    | 48/48 [01:49<00:00,  2.29s/it]
```

In [84]:

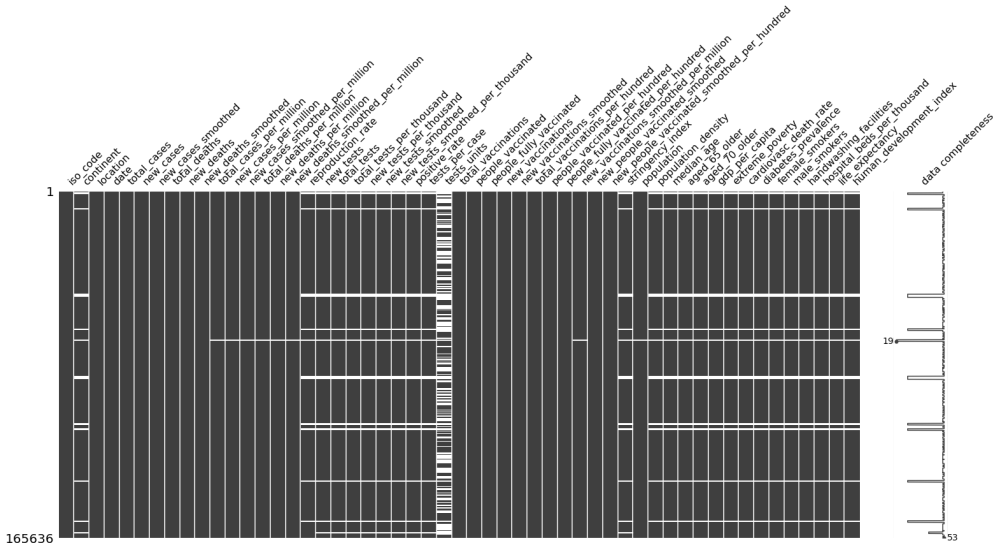
```
with pd.option_context("display.max_rows", mod_df.shape[0]+1):
    print(mod_df.isna().sum())
msno.matrix(mod_df, labels=True)
```

```
iso_code                                      0
continent                                  9917
location                                      0
date                                          0
total_cases                                   0
new_cases                                     0
new_cases_smoothed                            0
total_deaths                                  0
new_deaths                                    0
new_deaths_smoothed                           0
total_cases_per_million                     755
new_cases_per_million                       755
new_cases_smoothed_per_million              755
total_deaths_per_million                    755
new_deaths_per_million                      755
new_deaths_smoothed_per_million             755
reproduction_rate                          9146
new_tests                                  9917
total_tests                                9917
total_tests_per_thousand                   9917
new_tests_per_thousand                     9917
new_tests_smoothed                         9917
new_tests_smoothed_per_thousand            9917
positive_rate                              9917
tests_per_case                             9917
tests_units                               79655
total_vaccinations                            0
people_vaccinated                             0
people_fully_vaccinated                       0
new_vaccinations                              0
new_vaccinations_smoothed                     0
total_vaccinations_per_hundred                0
people_vaccinated_per_hundred                 0
people_fully_vaccinated_per_hundred           0
new_vaccinations_smoothed_per_million       755
new_people_vaccinated_smoothed                0
new_people_vaccinated_smoothed_per_hundred    0
stringency_index                           9917
population                                  755
population_density                         9146
median_age                                 9146
aged_65_older                              9146
aged_70_older                              9146
gdp_per_capita                             9146
```

```
extreme_poverty                          9146
cardiovasc_death_rate                    9146
diabetes_prevalence                      9146
female_smokers                           9146
male_smokers                             9146
handwashing_facilities                   9146
hospital_beds_per_thousand               9146
life_expectancy                          9146
human_development_index                  9146
dtype: int64
```

Out[84]:

```
<AxesSubplot:>
```



مقادیر ستون test_unit پس از این مراحل همچنان داده خالی زادی دارد و تصمیم بر حذف آن میگیریم.

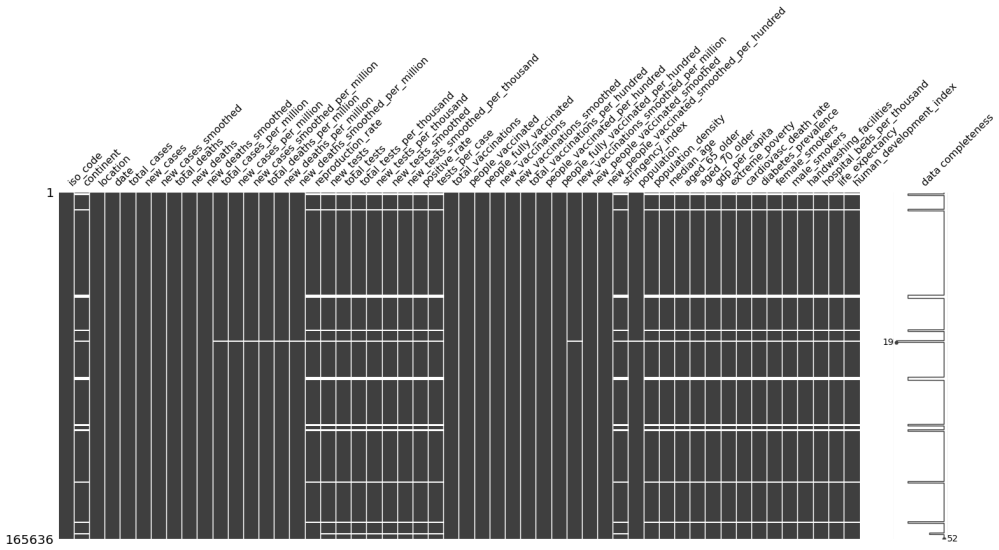
In [85]:

```
mod_df = mod_df.drop('tests_units', 1)
msno.matrix(mod_df, labels=True)
```

```
C:\Users\mamat\AppData\Local\Temp/ipykernel_15300/2207112408.py:
1: FutureWarning: In a future version of pandas all arguments of
DataFrame.drop except for the argument 'labels' will be keyword-
only
  mod_df = mod_df.drop('tests_units', 1)
```

Out[85]:

<AxesSubplot:>



۱-۲ دیتافریم دیگری درست نمایید که در آن، تعداد کیس‌های جدید، تعداد واکسینه‌های جدید، تعداد فوتیها و جمعیت برای هر کشور به صورت تجمیع شده محاسبه شده باشد. (محاسبه‌ی جمع داده‌ها از ابتدا تا آخرین تاریخ موجود در مجموعه داده‌ها برای هر کشور)

In [87]:

```
df1 = mod_df[["location", "new_cases", "new_vaccinations", "new_deaths", '
df1
```

Out[87]:

location	new_cases	new_vaccinations	new_deaths	population
Afghanistan	1.745540e+05	2.974970e+06	7.917389e+03	2.939855e+10
Africa	1.123052e+07	5.818190e+08	2.486680e+05	1.028741e+12
Albania	2.767058e+05	3.002796e+06	3.546175e+03	2.117352e+09
Algeria	2.650790e+05	8.412034e+06	6.994856e+03	3.288245e+10
Andorra	3.824900e+04	1.605815e+06	1.552475e+02	5.654577e+07
...	...	...	...	...
Wallis and Futuna	4.540000e+02	0.000000e+00	1.008646e+01	5.547000e+06
World	4.390117e+08	1.122516e+10	5.946817e+06	6.071599e+12
Yemen	1.178904e+04	7.335200e+04	2.198542e+03	2.109952e+10
Zambia	3.132030e+05	3.583561e+06	4.039750e+03	1.352827e+10
Zimbabwe	2.378426e+05	8.920041e+06	5.418800e+03	1.076072e+10

238 rows × 4 columns

۱-۳ ستون جدیدی با اسم تاریخ شمسی ایجاد کنید و برای ایجاد آن، تاریخ میلادی را به شمسی تبدیل نمایید

In [88]:

```
from persiantools.jdatetime import JalaliDate
import datetime
import jalali_pandas

#for our local dataset
mod_df.date = pd.to_datetime(mod_df.date)
mod_df["shamsi_date"] = mod_df.date.jalali.to_jalali()
mod_df
```

Out[88]:

	iso_code	continent	location	date	total_cases	new_cases
0	AFG	Asia	Afghanistan	2020-02-24	5.0	5.0
1	AFG	Asia	Afghanistan	2020-02-25	5.0	0.0
2	AFG	Asia	Afghanistan	2020-02-26	5.0	0.0
3	AFG	Asia	Afghanistan	2020-02-27	5.0	0.0
4	AFG	Asia	Afghanistan	2020-02-28	5.0	0.0
...	...	...	...	...	...	..
165631	ZWE	Africa	Zimbabwe	2022-02-26	235803.0	336.0
165632	ZWE	Africa	Zimbabwe	2022-02-27	235803.0	0.0
165633	ZWE	Africa	Zimbabwe	2022-02-28	236380.0	577.0
165634	ZWE	Africa	Zimbabwe	2022-03-01	236871.0	491.0
165635	ZWE	Africa	Zimbabwe	2022-03-02	237503.0	632.0

165636 rows × 53 columns

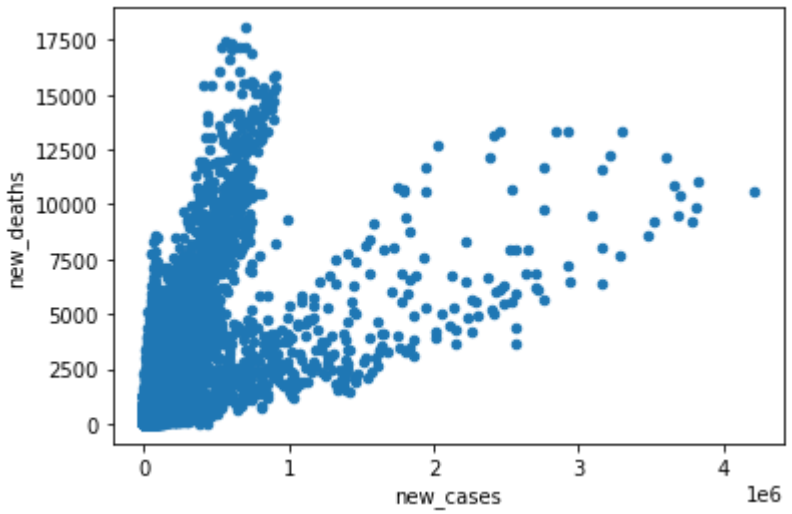
۴-۱ با توجه به تعداد بالای ویژگیها، آیا میتوان از تعداد ویژگیها کاست؟ (از معیار correlationمیتوانید استفاده کنید.)

In [119]:

```
1 mod_df.plot.scatter(x='new_cases', y='new_deaths')
```

Out[119]:

<AxesSubplot:xlabel='new_cases', ylabel='new_deaths'>

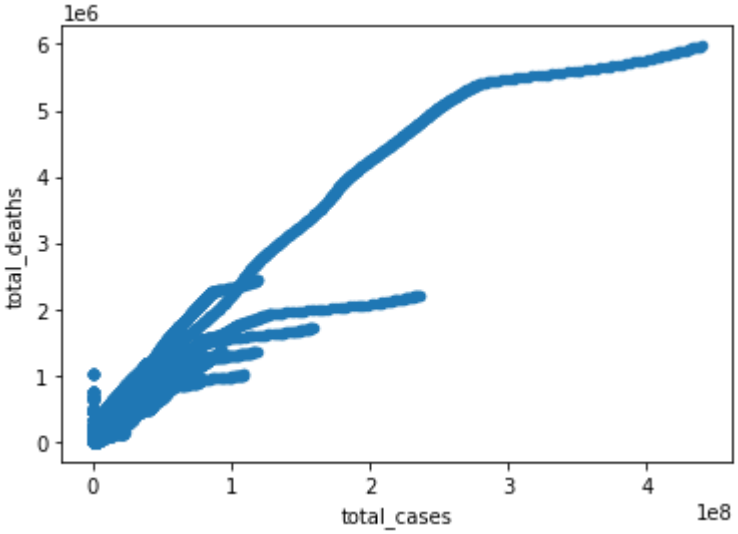


In [117]:

```
1 mod_df.plot.scatter(x='total_cases', y='total_deaths')
```

Out[117]:

<AxesSubplot:xlabel='total_cases', ylabel='total_deaths'>



بله با توجه به نمودارهای scatter-plot بین ستون ها(فیچرها) میتوان در آنها همبستگی ها را شناسایی و آنهایی که با یکدیگر همبستگی دارند را بدست آورد و حذف نمود. باید توجه داشت که این همبستگی معنادار باشد و متغیر confounding بین دو ویژگی نباشد که باعث این همبستگی شده باشد.

شکل‌های بدست آمده به دلیل تخمین بودن داده‌ها بعضا ظاهر مناسبی ندارند ولی همبستگی میان آنها قابل تشخیص است.

۱-۵ دیتافریم جدیدی درست نمایید که در آن صرفاً اطلاعات مربوط به کشور ایران قرار داده شده باشد

In [91]:

```
iran_df = mod_df.loc[mod_df['location'] == "Iran"]
iran_df
```

Out[91]:

	iso_code	continent	location	date	total_cases	new_cases
71639	IRN	Asia	Iran	2020-02-19	2.0	2.0
71640	IRN	Asia	Iran	2020-02-20	5.0	3.0
71641	IRN	Asia	Iran	2020-02-21	18.0	13.0
71642	IRN	Asia	Iran	2020-02-22	28.0	10.0
71643	IRN	Asia	Iran	2020-02-23	43.0	15.0
...	...	...	...	...	...	...
72377	IRN	Asia	Iran	2022-02-26	7030943.0	7039.0
72378	IRN	Asia	Iran	2022-02-27	7040467.0	9524.0
72379	IRN	Asia	Iran	2022-02-28	7051429.0	10962.0
72380	IRN	Asia	Iran	2022-03-01	7060741.0	9312.0
72381	IRN	Asia	Iran	2022-03-02	7066975.0	6234.0

743 rows × 53 columns

۱-۶ در دیتافریم ایران، ستونی ایجاد نمایید که در آن، ماه به عنوان یک ویژگی مستقل در نظر گرفته شده است

In [92]:

```
iran_df['gregorian_month'] = pd.to_datetime(iran_df['date']).dt.month
iran_df["shamsi_month"] = iran_df.shamsi_date.jalali.month

iran_df
```

Out[92]:

nd	life_expectancy	human_development_index	shamsi_date	gregorian_mo
1.5	76.68	0.783	1398-11-30 00:00:00	
1.5	76.68	0.783	1398-12-01 00:00:00	
1.5	76.68	0.783	1398-12-02 00:00:00	
1.5	76.68	0.783	1398-12-03 00:00:00	
1.5	76.68	0.783	1398-12-04 00:00:00	
...	...	...	...	
1.5	76.68	0.783	1400-12-07 00:00:00	
1.5	76.68	0.783	1400-12-08 00:00:00	
1.5	76.68	0.783	1400-12-09 00:00:00	
1.5	76.68	0.783	1400-12-10 00:00:00	
1.5	76.68	0.783	1400-12-11 00:00:00	

۱-۷ دیتافریم جدیدی ایجاد نمایید که مجموعه داده ایران را بر اساس ماه در سال ۲۰۲۱تجمیع کند

In [93]:

```
iran2021_df = iran_df.loc[(iran_df['location'] == "Iran") & (iran_df['date
iran2021_df.groupby("gregorian_month").first()
```

Out[93]:

l_beds_per_thousand	life_expectancy	human_development_index	shamsi_da
1.5	76.68	0.783	1399-1 00:(
1.5	76.68	0.783	1399-1 00:(
1.5	76.68	0.783	1399-1 00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(
1.5	76.68	0.783	1400-(00:(

۲ نمایش دادگان

یکی از مواردی که در داده‌کاوی بسیار مورد استفاده قرار میگیرد، مصورسازی داده‌ها میباشد که به کمک آن میتوان درکی از مجموعه داده‌ی مورد نظر به دست آورد و همچنین، تحلیلهای کاملی بر اساس نمودارهای به دست آمده، ارائه نمود.

۲-۱ کدام کشورها بهترین و کدام کشورها بدترین عملکرد در مهار ویروس کرونا را داشته اند؟ با یک نمودار مناسب این مساله را بررسی نمایید و برداشت خود را از نتایج ذکر نمایید. (منظور از عملکرد، تعداد فوتی نسبت به کل جمعیت است.)

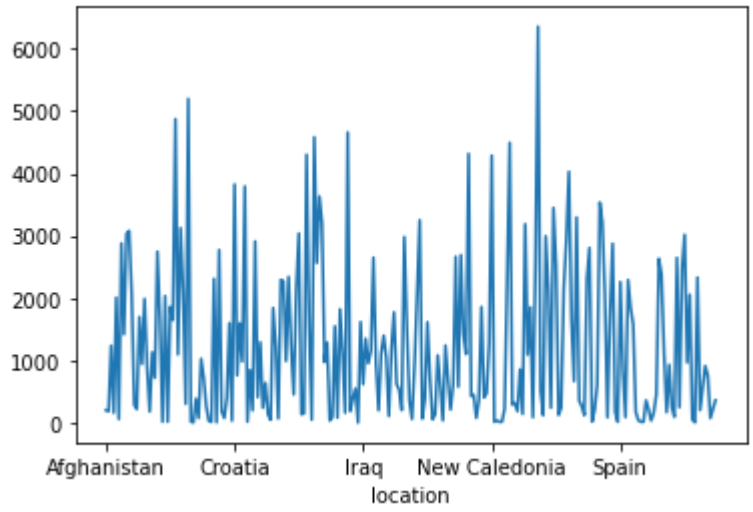
بهترین/بدترین کشورها با توجه به مرگ و میر نسبت به جمعیت آنها سنجیده میشوند برای اینکار مرگ و میر در میلیون نفر جمعیت را برای هر کشور جمع میزنیم و مقایسه میکنیم که نتایج در زیر مشاهده میشود

In [94]:

```
1 mod_df.groupby('location')["new_deaths_per_million"].sum().plot()
```

Out[94]:

<AxesSubplot:xlabel='location'>



In [95]:

```
1 mod_df.groupby('location').sum().nsmallest(10, 'new_deaths_per_million')
```

Out[95]:

location	total_cases	new_cases	new_cases_smoothed	total_deaths
International	537473.0	721.000000	635.912610	1.094343e+0
Burundi	5707819.0	38127.000000	38414.181336	9.674151e+0
China	67919335.0	109850.433594	109912.280708	3.374058e+0
Vanuatu	1962.0	19.000000	18.232864	4.780000e+0
New Zealand	3806317.0	167013.153425	106841.942820	2.053636e+0
Chad	2409357.0	7257.000000	7311.881381	9.611775e+0
Niger	2881815.0	8775.307584	8896.783443	1.079621e+0
South Sudan	5320426.0	16989.000000	17110.430081	6.346731e+0
Tajikistan	8652869.0	17786.000000	17834.917654	6.277812e+0
Tanzania	4909841.0	33620.000000	33900.405401	1.293000e+0

10 rows × 48 columns

In [96]:

```
1 mod_df.groupby('location').sum().nlargest(10, 'new_deaths_per_million')
```

Out[96]:

location	total_cases	new_cases	new_cases_smoothed	total_deaths
Peru	1.011461e+09	3.518721e+06	3.543039e+06	9.056462e+07
Bulgaria	2.272544e+08	1.096194e+06	1.098723e+06	8.975619e+06
Bosnia and Herzegovina	1.024036e+08	3.715530e+05	3.739922e+05	4.493398e+06
Hungary	3.857990e+08	1.795580e+06	1.800467e+06	1.283320e+07
Gibraltar	2.723664e+06	1.569660e+04	1.598372e+04	5.997362e+04
North Macedonia	7.956426e+07	2.981950e+05	2.994287e+05	2.742471e+06
Montenegro	5.521087e+07	2.305120e+05	2.320334e+05	7.889604e+05
Georgia	2.544243e+08	1.616159e+06	1.614808e+06	3.544215e+06
New Caledonia	2.803950e+06	5.558084e+04	5.468645e+04	1.721684e+05
Saint Pierre and Miquelon	5.880800e+04	1.090565e+03	1.096466e+03	6.970000e+02

10 rows × 48 columns

۲-۲ میخواهیم تاثیر واکسیناسیون بر تعداد فوتی‌ها را بررسی کنیم. برای این کار فرض کنید الزام است که اطلاعات ۵کشور را بررسی کنیم. شما کدام کشورها را برای مقایسه انتخاب میکنید؟ با یک نمودار مناسب این مساله را بررسی نمایید و برداشت خود را از نتایج ذکر نمایید.

۵ کشور انتخابی را کشورهایی با تعداد واکسینه بالا در نظر میگیریم زیرا هرچه آماره‌های بیشتری داشته باشیم، نتایج را بهتر میتوانیم مورد تحلیل و بررسی قرار دهیم.

In [97]:

```
1 mod_df.groupby('location').sum().nlargest(20, 'new_vaccinations')
```

Out[97]:

location	total_cases	new_cases	new_cases_smoothed	total_deaths
World	9.749264e+10	4.390117e+08	4.380398e+08	1.988572e+0
Asia	2.795110e+10	1.178114e+08	1.168446e+08	4.073148e+0
Upper middle income	2.951138e+10	1.182120e+08	1.182281e+08	8.046135e+0
China	6.791934e+07	1.098504e+05	1.099123e+05	3.374058e+0
Lower middle income	2.216595e+10	8.258872e+07	8.266932e+07	3.931400e+0
High income	4.534286e+10	2.363931e+08	2.353120e+08	7.971662e+0
India	1.274592e+10	4.294516e+07	4.326238e+07	1.785145e+0
Europe	2.808516e+10	1.588156e+08	1.581429e+08	5.747552e+0
North America	2.282960e+10	9.319513e+07	9.369690e+07	5.053764e+0
European Union	1.850176e+10	1.100334e+08	1.095869e+08	3.678955e+0
South America	1.529715e+10	5.422262e+07	5.446028e+07	4.715735e+0
Africa	3.175293e+09	1.123052e+07	1.130212e+07	8.123984e+0
United States	1.936437e+10	7.924650e+07	7.970140e+07	3.426091e+0
Brazil	8.654431e+09	2.881973e+07	2.906807e+07	2.417534e+0
Pakistan	5.009209e+08	1.511754e+06	1.521661e+06	1.127242e+0
Bangladesh	5.684844e+08	1.958523e+06	1.972327e+06	9.391999e+0
Indonesia	1.330201e+09	5.630096e+06	5.573406e+06	4.213669e+0
Japan	5.698234e+08	5.150496e+06	4.993874e+06	6.569827e+0
Russia	3.274351e+09	1.635387e+07	1.616838e+07	8.635976e+0

location	total_cases	new_cases	new_cases_smoothed	total_deaths
Turkey	2.842550e+09	1.340146e+07	1.325831e+07	2.581149e+0

20 rows × 48 columns

In [98]:

```
isChina = mod_df['location']=='China'
df_china = mod_df[isChina]
df_china
```

Out[98]:

	iso_code	continent	location	date	total_cases	new_cases
31401	CHN	Asia	China	2020-01-22	547.0	142.477865
31402	CHN	Asia	China	2020-01-23	639.0	92.000000
31403	CHN	Asia	China	2020-01-24	916.0	277.000000
31404	CHN	Asia	China	2020-01-25	1399.0	483.000000
31405	CHN	Asia	China	2020-01-26	2062.0	663.000000
...	...	...	...	...	...	...
32167	CHN	Asia	China	2022-02-26	109092.0	239.000000
32168	CHN	Asia	China	2022-02-27	109326.0	234.000000
32169	CHN	Asia	China	2022-02-28	109526.0	200.000000
32170	CHN	Asia	China	2022-03-01	109750.0	224.000000
32171	CHN	Asia	China	2022-03-02	109964.0	214.000000

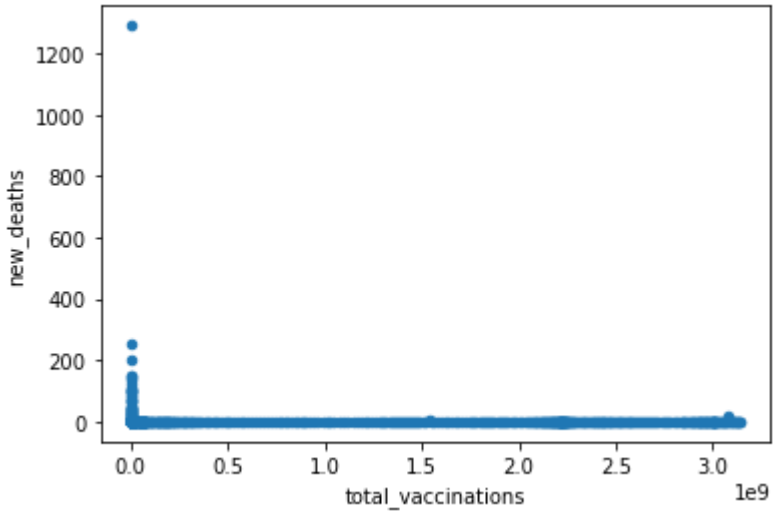
771 rows × 53 columns

In [122]:

```
1 df_china.plot.scatter(x="total_vaccinations", y="new_deaths")
```

Out[122]:

<AxesSubplot:xlabel='total_vaccinations', ylabel='new_deaths'>

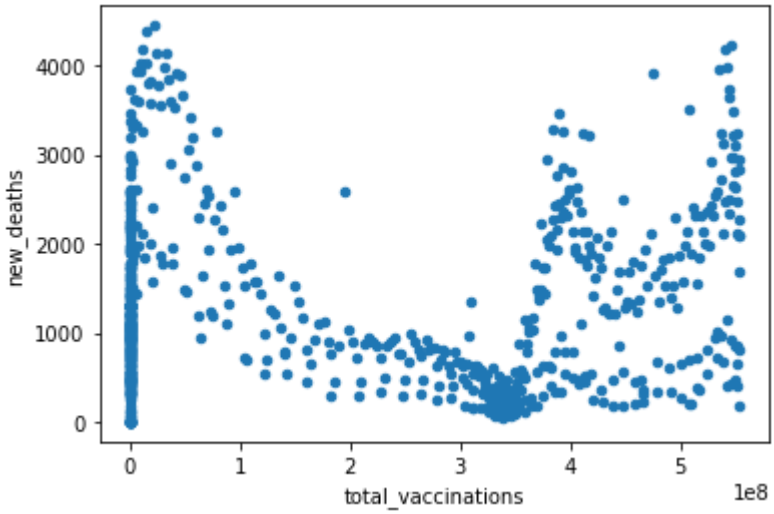


In [125]:

```
isUS = mod_df['location']=='United States'
df_us = mod_df[isUS]
df_us.plot.scatter(x="total_vaccinations", y="new_deaths")
```

Out[125]:

<AxesSubplot:xlabel='total_vaccinations', ylabel='new_deaths'>

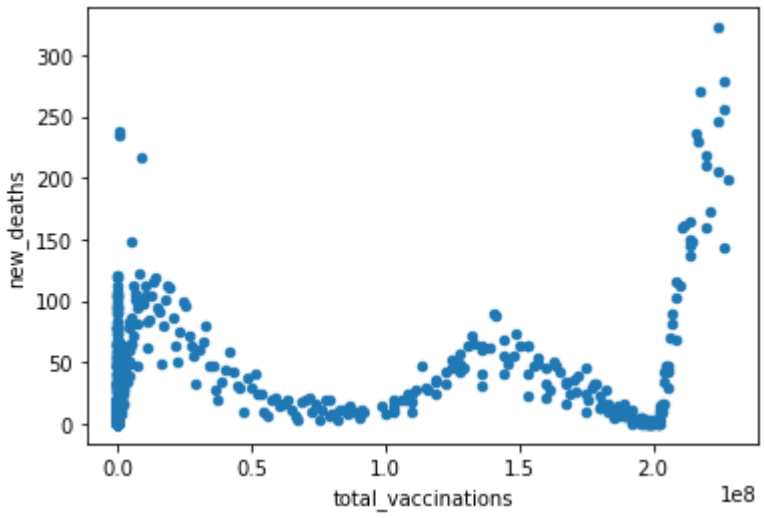


In [124]:

```
isJpn = mod_df['location']=='Japan'
df_Jpn = mod_df[isJpn]
df_Jpn.plot.scatter(x="total_vaccinations", y="new_deaths")
```

Out[124]:

```
<AxesSubplot:xlabel='total_vaccinations', ylabel='new_deaths'>
```



با توجه به تصاویر در کشور چین تاثیر واکسن بسیار زیاد بوده البته این تاثیر بعلت سخت گیری‌های شدید این کشور نیز هست. در مقایسه نمودار کشورهای آمریکا و ژاپن را میبینیم که تاثیر واکسن بر آنها بصورت متناوب بوده و در بازه‌هایی همبستگی منفی(کاهش آمار کرونا) بین آنها وجود دارد و در بازه‌هایی همبستگی مثبت(افزایش آمار کرونا)

۳-۲ قصد داریم سرعت واکسیناسیون در کشورهای مختلف را بررسی کنیم. برای این کار فرض کنید الزام ست که اطلاعات ۵کشور را ارزیابی کنیم. شما کدام کشورها را برای مقایسه انتخاب میکنید؟ با یک نمودار مناسب این مساله را بررسی نمایید و برداشت خود را از نتایج ذکر نمایید

برای انتخاب کشورهای برتر برای سرعت واکسیناسیون، کشورهایی با آمار بالای تعداد افراد واکسینه شده در ۱۰۰ نفر را مورد بررسی قرار میدهیم.

In [100]:

```
1 mod_df.groupby('location').sum().nlargest(5, 'people_vaccinated_per_hundre
```

Out[100]:

location	total_cases	new_cases	new_cases_smoothed	total_deaths
Gibraltar	2723664.0	15696.595041	15983.719615	5.997362e+04
China	67919335.0	109850.433594	109912.280708	3.374058e+06
Saudi Arabia	278340103.0	746066.000000	750521.229974	4.359211e+06
United Arab Emirates	286274793.0	880970.000000	886244.333029	9.212383e+05
British Virgin Islands	839359.0	6085.000000	6128.742511	9.957738e+03

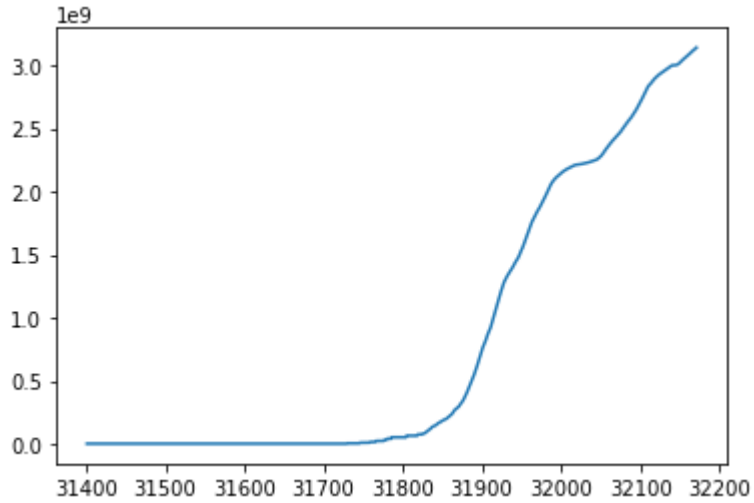
5 rows × 48 columns

In [101]:

```
1 df_china["total_vaccinations"].plot()
```

Out[101]:

<AxesSubplot:>

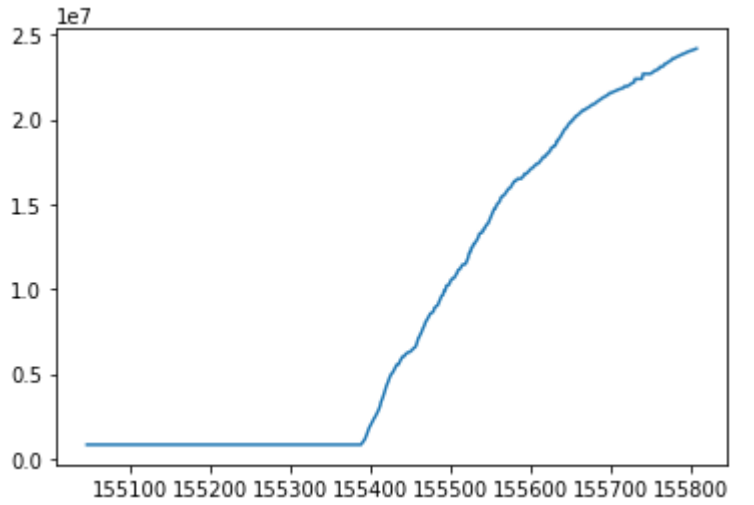


In [128]:

```
1 isUA = mod_df['location']=='United Arab Emirates'
2 df_UA = mod_df[isUA]
3 df_UA["total_vaccinations"].plot()
```

Out[128]:

<AxesSubplot:>

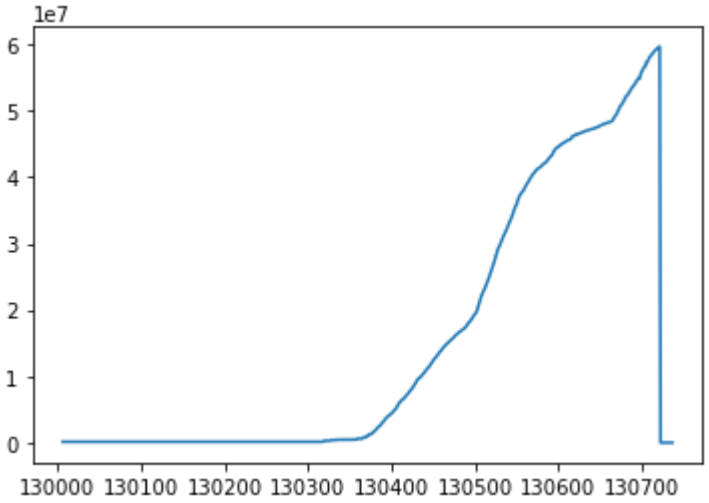


In [129]:

```
isSA = mod_df['location']=='Saudi Arabia'
df_SA = mod_df[isSA]
df_SA["total_vaccinations"].plot()
```

Out[129]:

<AxesSubplot:>



۲-۴ روند سختگیری در حوزه‌ی کرونا در ایران را در طول زمان بررسی کنید، توجه نمایید برای پاسخگویی به این سوال براساس تحلیل خود میتوانید از ویژگی یا ویژگیهای دلخواه استفاده نمایید، تحلیل خود را بیان نمایید.

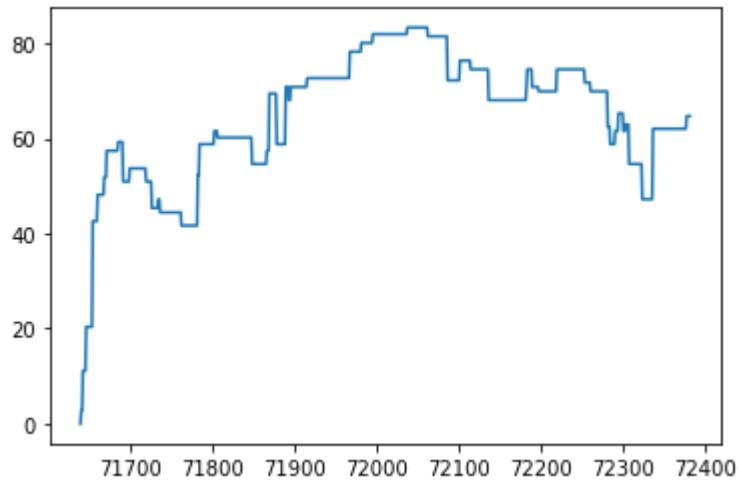
به صورت کلی سخت گیریها در ابتدا کم و به مرور زیاد شده و در ادامه پس از مدتی کمتر شده است (ولی از حالت ابتدایی همچنان بالاتر) همچنین بازه‌هایی وجود دارند که بسیار پایین آمده که احتمالا مربوط به زمانهایی است که در کرونا سهل انگاری انجام میشد

In [102]:

```
1 iran_df["stringency_index"].plot()
```

Out[102]:

<AxesSubplot:>



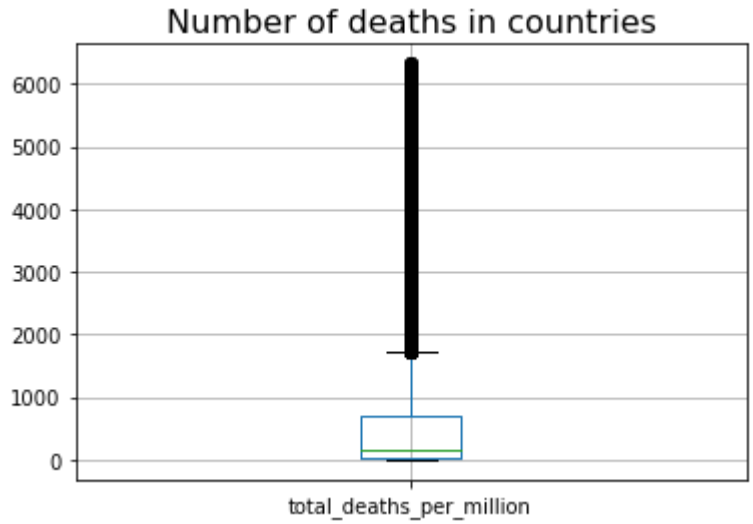
۲-۵ با استفاده از دیتافریم تجمیع شده ای که ایجاد کردید، برای ویژگی تعداد فوتیهای هر کشور نمودار BoxPlotرسم کنید و کشورهای پرت را شناسایی کنید و رویکرد مناسبی برای آنها اتخاذ نمایید. با توجه به مقدار میانه و میانگین، چولگی نمودار به کدام سمت میباشد؟

In [103]:

```
boxplot = mod_df.boxplot(column=['total_deaths_per_million'])
plt.title('Number of deaths in countries', fontsize=16)
boxplot
```

Out[103]:

```
<AxesSubplot:title={'center':'Number of deaths in countries'}>
```



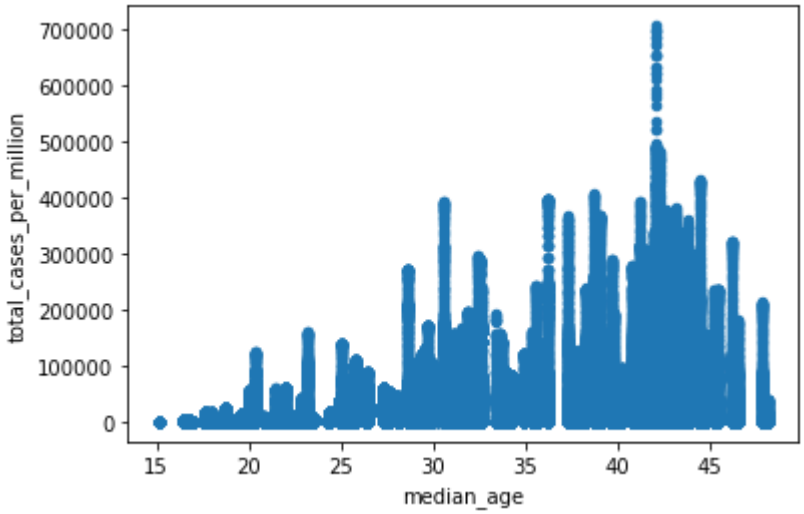
۶-۲ تاثیر ویژگیهای تراکم جمعیت، میانگین سنی، وجود امکانات بهداشتی، تعداد تخت بیمارستانها و شاخص پیشرفت انسانی را بر تعداد فوتیها و تعداد کیسهای جدید با رسم نمودار مناسب بررسی کنید

In [104]:

```
mod_df.plot.scatter(x='median_age',
                    y='total_cases_per_million')
```

Out[104]:

```
<AxesSubplot:xlabel='median_age', ylabel='total_cases_per_millio
n'>
```



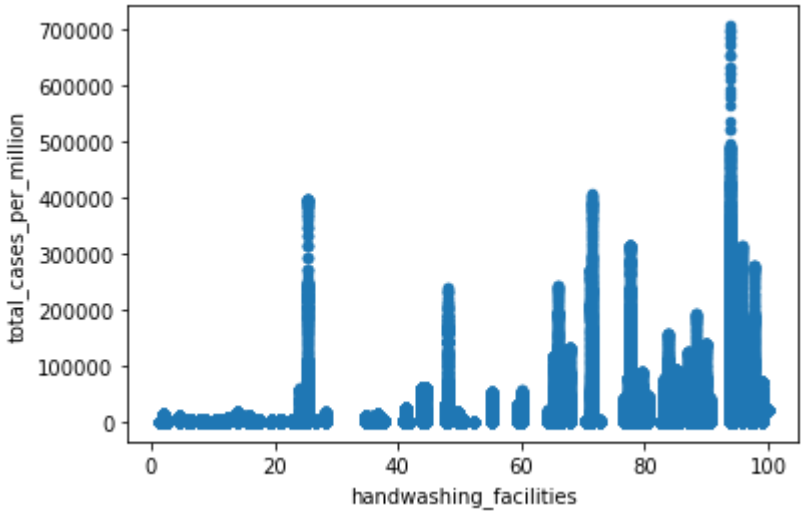
با توجه به نمودار هرچه میانگین سنی بالا میرود تعداد مبتلایان نیز افزایش پیدا میکند.

In [105]:

```
1
2 mod_df.plot.scatter(x='handwashing_facilities',
3                     y='total_cases_per_million')
```

Out[105]:

```
<AxesSubplot:xlabel='handwashing_facilities', ylabel='total_case
s_per_million'>
```



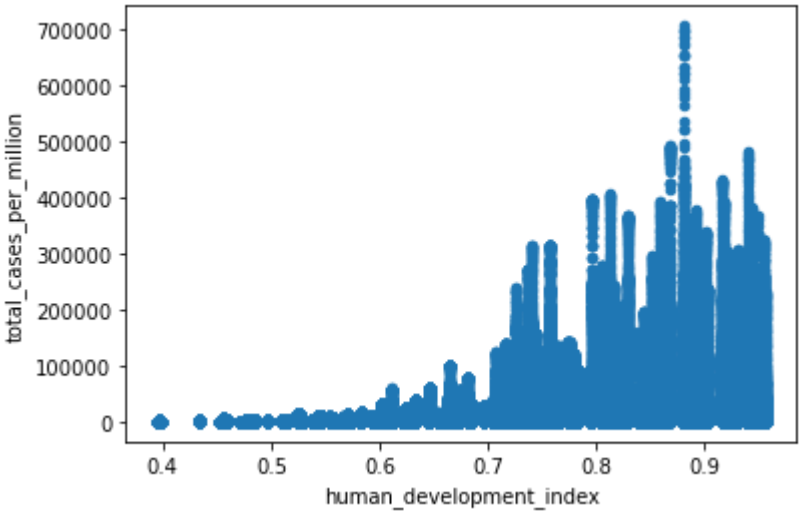
آمار در این ویژگی دقت بالایی ندارد ولی میتوان کفت به صورت میانگین کشورهایی که آمار بالاتری داشته اند امکانات بهداشتی بیشتری فراهم کرده‌اند

In [106]:

```
1
2 mod_df.plot.scatter(x='human_development_index',
3                     y='total_cases_per_million')
```

Out[106]:

```
<AxesSubplot:xlabel='human_development_index', ylabel='total_cas
es_per_million'>
```



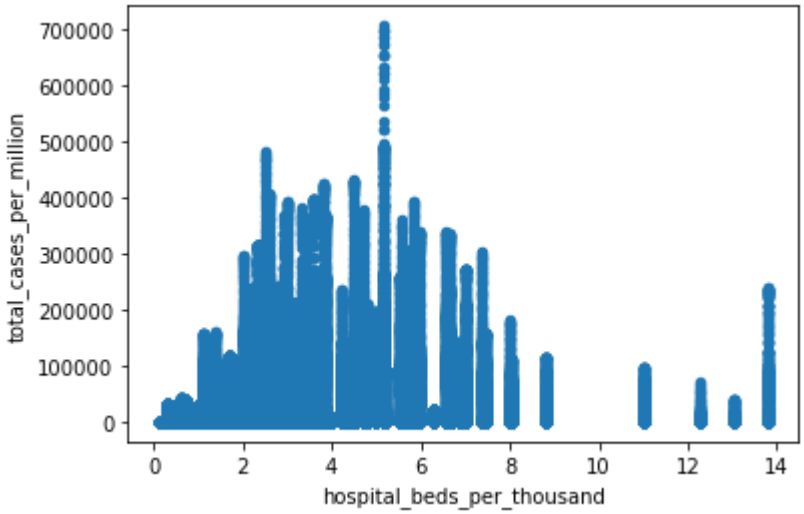
با توجه به نمودار تعداد مبتلایان در کشورهای توسعه یافته بیشتر از کشورهای توسعه نیافته است.

In [107]:

```
1
2 mod_df.plot.scatter(x='hospital_beds_per_thousand',
3                     y='total_cases_per_million')
```

Out[107]:

```
<AxesSubplot:xlabel='hospital_beds_per_thousand', ylabel='total_
cases_per_million'>
```



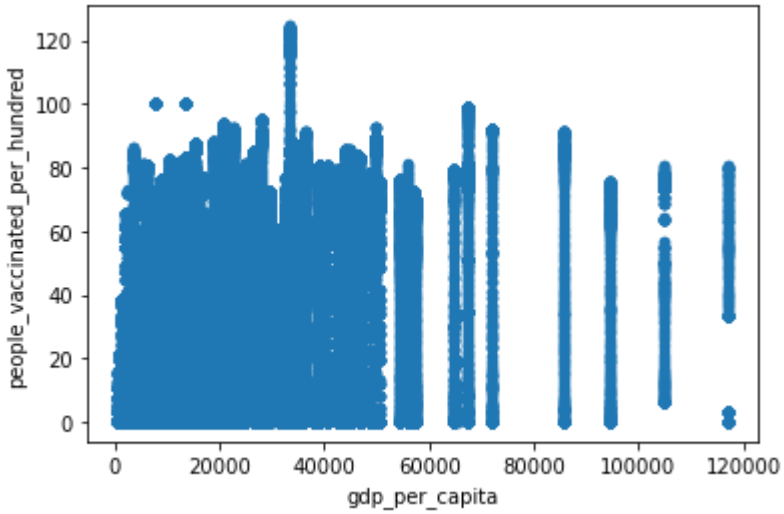
۷-۲ رابطه بین وضعیت اقتصادی کشورها و تعداد افراد واکسینه شده را بررسی کنید و تحلیل خود را بیان نمایید

In [108]:

```
mod_df.plot.scatter(x='gdp_per_capita',
                    y='people_vaccinated_per_hundred')
```

Out[108]:

```
<AxesSubplot:xlabel='gdp_per_capita', ylabel='people_vaccinated_
per_hundred'>
```



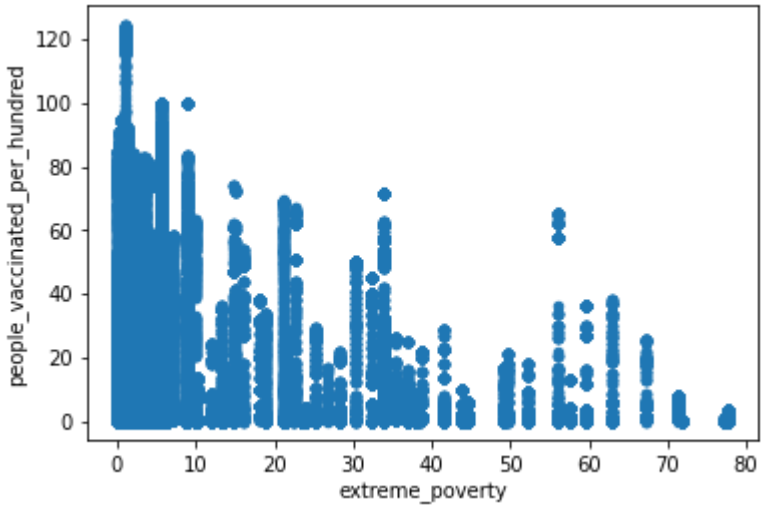
تولید ناخالص داخلی معیار پولی ارزش بازار همه کالاها و خدمات نهایی تولید شده در یک دوره زمانی خاص توسط کشورها است. با توجه به نمودار میتوان به صورت میانگین گفت که این معیار تاثیر به سزایی در تعداد افراد واکسینه شده در کشور ندارد.

In [109]:

```
mod_df.plot.scatter(x='extreme_poverty',
                    y='people_vaccinated_per_hundred')
```

Out[109]:

```
<AxesSubplot:xlabel='extreme_poverty', ylabel='people_vaccinated
_per_hundred'>
```



با توجه به نمودار میتوان کفت هرچه فقر مطلق در کشوری بیشتر باشد تعداد افراد واکسینه شده در آن نیز کمتر است.

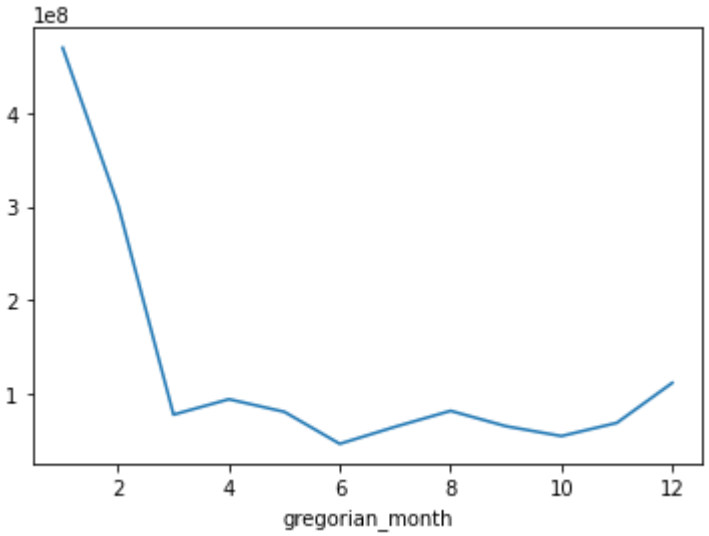
```
<'span style='font-family:B Nazanin;font-size:16px>

۸-۲ در سال 2021توزیع تعداد مبتلایان به تفکیک ماه را بررسی نمایید و تحلیل خود را ذکر نمایید
```

In [116]:

```
mod_df['gregorian_month'] = pd.to_datetime(mod_df['date']).dt.month
mod2021_df = mod_df.loc[mod_df["date"] > datetime.datetime(2021, 1,1)]
tmp = mod2021_df.groupby("gregorian_month").sum()
tmp["new_cases"].plot()
```

Out[116]:

<AxesSubplot:xlabel='gregorian_month'>



آمار کیس‌های جدید کرونا در ابتدای شیوع بسیار بالا بوده و سپس کاهش یافته و پس از آن بصورت متناوب دچار پیک ها و کاهش‌هایی شده است.

In [ ]:

```
1
```