covid19

September 23, 2023

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import plotly.express as px
```

[2]:
```
dir(px)
```

[2]:
```
['Constant',
 'IdentityMap',
 'NO_COLOR',
 'Range',
 '__all__',
 '__builtins__',
 '__cached__',
 '__doc__',
 '__file__',
 '__loader__',
 '__name__',
 '__package__',
 '__path__',
 '__spec__',
 '_chart_types',
 '_core',
 '_doc',
 '_imshow',
 '_special_inputs',
 'absolute_import',
 'area',
 'bar',
 'bar_polar',
 'box',
 'choropleth',
 'choropleth_mapbox',
 'colors',
 'data',
 'defaults',
 'density_contour',
```

```
 'density_heatmap',
 'density_mapbox',
 'ecdf',
 'funnel',
 'funnel_area',
 'get_trendline_results',
 'histogram',
 'icicle',
 'imshow',
 'imshow_utils',
 'line',
 'line_3d',
 'line_geo',
 'line_mapbox',
 'line_polar',
 'line_ternary',
 'optional_imports',
 'parallel_categories',
 'parallel_coordinates',
 'pd',
 'pie',
 'scatter',
 'scatter_3d',
 'scatter_geo',
 'scatter_mapbox',
 'scatter_matrix',
 'scatter_polar',
 'scatter_ternary',
 'set_mapbox_access_token',
 'strip',
 'sunburst',
 'timeline',
 'treemap',
 'trendline_functions',
 'violin']
```

[3]:
```
import plotly
import plotly.graph_objs as go
from plotly import tools
from plotly.offline import init_notebook_mode,plot,iplot
```

[4]:
```
dir(plotly)
```

[4]:
```
['__version__',
 'colors',
 'data',
 'graph_objects',
```

```
 'graph_objs',
 'io',
 'offline',
 'tools',
 'utils']
```

[5]:
```
print(plotly.__version__)
```

```
5.16.1
```

[6]:
```
current_data = pd.read_csv('/home/samim/global/covid.csv')
current_data.head()
```

[6]:

	Date	Country	Confirmed	Recovered	Deaths
0	2020-01-22	Afghanistan	0	0	0
1	2020-01-23	Afghanistan	0	0	0
2	2020-01-24	Afghanistan	0	0	0
3	2020-01-25	Afghanistan	0	0	0
4	2020-01-26	Afghanistan	0	0	0

[7]:
```
current_data.tail(10)
```

[7]:

	Date	Country	Confirmed	Recovered	Deaths
161558	2022-04-07	Zimbabwe	246870	0	5455
161559	2022-04-08	Zimbabwe	246925	0	5457
161560	2022-04-09	Zimbabwe	246925	0	5457
161561	2022-04-10	Zimbabwe	246958	0	5457
161562	2022-04-11	Zimbabwe	247010	0	5460
161563	2022-04-12	Zimbabwe	247094	0	5460
161564	2022-04-13	Zimbabwe	247160	0	5460
161565	2022-04-14	Zimbabwe	247208	0	5462
161566	2022-04-15	Zimbabwe	247237	0	5462
161567	2022-04-16	Zimbabwe	247237	0	5462

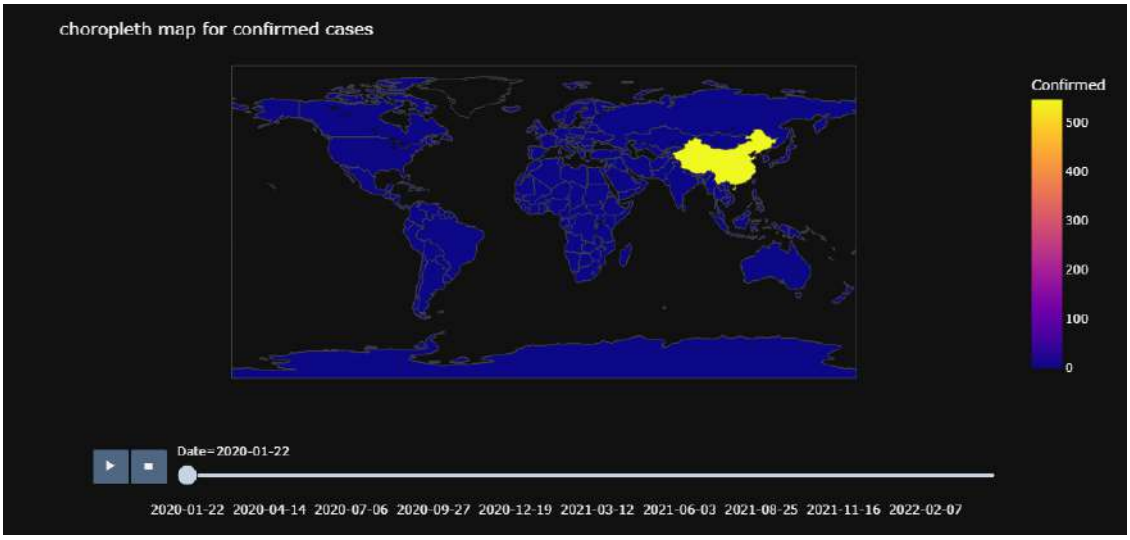
0.1 choropleth map: show interval data as colours. They are shaded in using one colour where the darker shades means higher number and lighter shades means lower number.

0.2 1st read the colour Legend/Key to know the shading behaviour.Then darker shades and Lower shades.

1 choropleth map for confirmed covid-19 cases ?

[8]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Confirmed',animation_frame='Date')
fig.update_layout(title='choropleth map for confirmed␣
 ↪cases',template='plotly_dark',width=1000,height=520,autosize=False,)
```

```
fig.show()
```



1.1 import plotly.io as pio

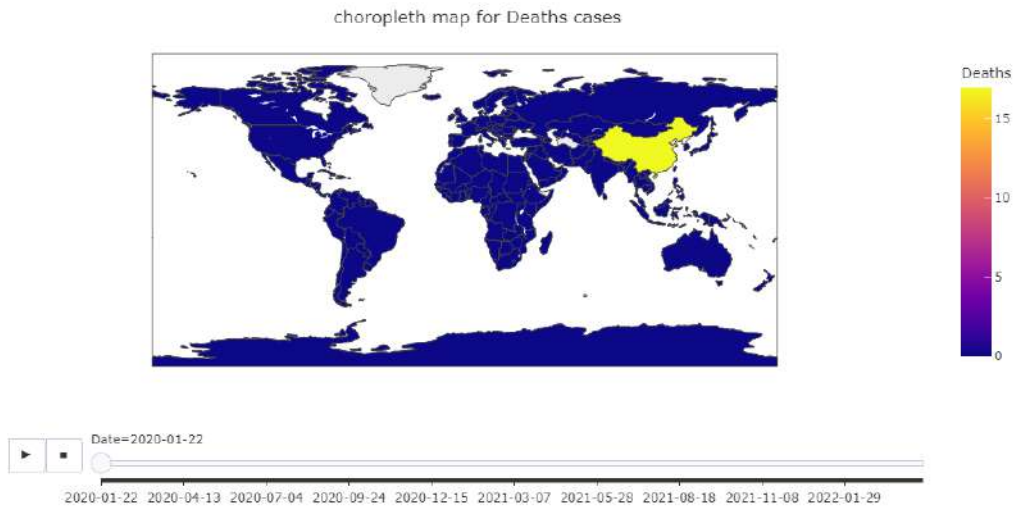
pio.templates

1.2

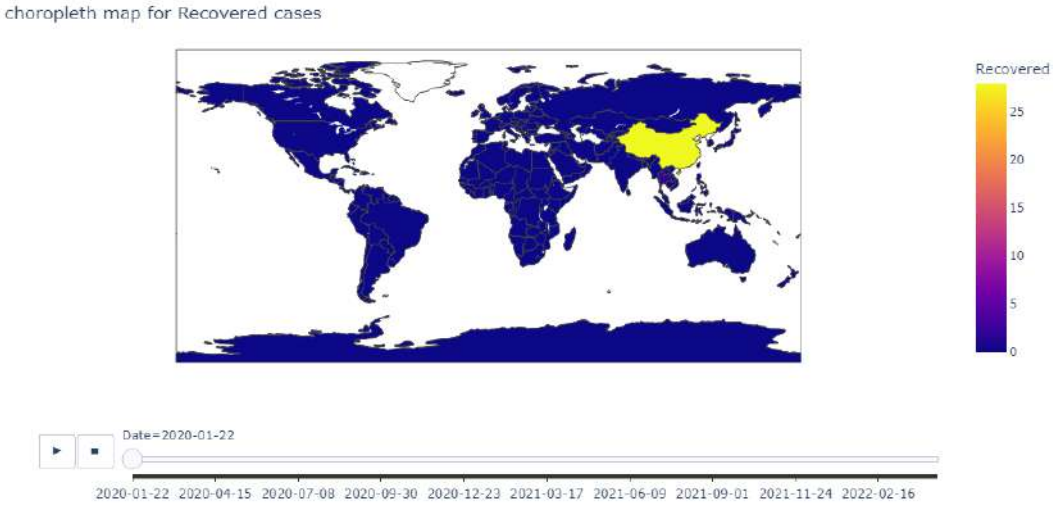
1.3 Templates configuration

```
Default template: 'plotly'
Available templates:
    ['ggplot2', 'seaborn', 'simple_white', 'plotly',
     'plotly_white', 'plotly_dark', 'presentation', 'xgridoff',
     'ygridoff', 'gridon', 'none']
```

[9]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Deaths',animation_frame='Date')
fig.update_layout(title='choropleth map for Deaths␣
 ↪cases',template='ggplot2',width=1200,height=520,autosize=False,)
fig.show()
```



[10]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Recovered',animation_frame='Date')
fig.update_layout(title='choropleth map for Recovered␣
 ↪cases',template='plotly_white',width=1100,height=520,autosize=False,)
fig.show()
```



2 Choropleth Map for a specific Continent ?

[11]:
```
help(px.choropleth)
```

```
Help on function choropleth in module plotly.express._chart_types:
```

```
choropleth(data_frame=None, lat=None, lon=None, locations=None,
locationmode=None, geojson=None, featureidkey=None, color=None, facet_row=None,
facet_col=None, facet_col_wrap=0, facet_row_spacing=None,
facet_col_spacing=None, hover_name=None, hover_data=None, custom_data=None,
animation_frame=None, animation_group=None, category_orders=None, labels=None,
color_discrete_sequence=None, color_discrete_map=None,
color_continuous_scale=None, range_color=None, color_continuous_midpoint=None,
projection=None, scope=None, center=None, fitbounds=None, basemap_visible=None,
title=None, template=None, width=None, height=None) ->
plotly.graph_objs._figure.Figure
        In a choropleth map, each row of `data_frame` is represented by a
        colored region mark on a map.

    Parameters
    ----------
    data_frame: DataFrame or array-like or dict
        This argument needs to be passed for column names (and not keyword
        names) to be used. Array-like and dict are transformed internally to a
        pandas DataFrame. Optional: if missing, a DataFrame gets constructed
        under the hood using the other arguments.
    lat: str or int or Series or array-like
        Either a name of a column in `data_frame`, or a pandas Series or
        array_like object. Values from this column or array_like are used to
        position marks according to latitude on a map.
    lon: str or int or Series or array-like
        Either a name of a column in `data_frame`, or a pandas Series or
        array_like object. Values from this column or array_like are used to
        position marks according to longitude on a map.
    locations: str or int or Series or array-like
        Either a name of a column in `data_frame`, or a pandas Series or
        array_like object. Values from this column or array_like are to be
        interpreted according to `locationmode` and mapped to
        longitude/latitude.
    locationmode: str
        One of 'ISO-3', 'USA-states', or 'country names' Determines the set of
        locations used to match entries in `locations` to regions on the map.
    geojson: GeoJSON-formatted dict
        Must contain a Polygon feature collection, with IDs, which are
        references from `locations`.
    featureidkey: str (default: `'id'`)
        Path to field in GeoJSON feature object with which to match the values
        passed in to `locations`.The most common alternative to the default is
        of the form `'properties.<key>`.
    color: str or int or Series or array-like
        Either a name of a column in `data_frame`, or a pandas Series or
        array_like object. Values from this column or array_like are used to
        assign color to marks.
```

```
facet_row: str or int or Series or array-like
    Either a name of a column in `data_frame`, or a pandas Series or
    array_like object. Values from this column or array_like are used to
    assign marks to facetted subplots in the vertical direction.
facet_col: str or int or Series or array-like
    Either a name of a column in `data_frame`, or a pandas Series or
    array_like object. Values from this column or array_like are used to
    assign marks to facetted subplots in the horizontal direction.
facet_col_wrap: int
    Maximum number of facet columns. Wraps the column variable at this
    width, so that the column facets span multiple rows. Ignored if 0, and
    forced to 0 if `facet_row` or a `marginal` is set.
facet_row_spacing: float between 0 and 1
    Spacing between facet rows, in paper units. Default is 0.03 or 0.0.7
    when facet_col_wrap is used.
facet_col_spacing: float between 0 and 1
    Spacing between facet columns, in paper units Default is 0.02.
hover_name: str or int or Series or array-like
    Either a name of a column in `data_frame`, or a pandas Series or
    array_like object. Values from this column or array_like appear in bold
    in the hover tooltip.
hover_data: str, or list of str or int, or Series or array-like, or dict
    Either a name or list of names of columns in `data_frame`, or pandas
    Series, or array_like objects or a dict with column names as keys, with
    values True (for default formatting) False (in order to remove this
    column from hover information), or a formatting string, for example
    ':.3f' or '|%a' or list-like data to appear in the hover tooltip or
    tuples with a bool or formatting string as first element, and list-like
    data to appear in hover as second element Values from these columns
    appear as extra data in the hover tooltip.
custom_data: str, or list of str or int, or Series or array-like
    Either name or list of names of columns in `data_frame`, or pandas
    Series, or array_like objects Values from these columns are extra data,
    to be used in widgets or Dash callbacks for example. This data is not
    user-visible but is included in events emitted by the figure (lasso
    selection etc.)
animation_frame: str or int or Series or array-like
    Either a name of a column in `data_frame`, or a pandas Series or
    array_like object. Values from this column or array_like are used to
    assign marks to animation frames.
animation_group: str or int or Series or array-like
    Either a name of a column in `data_frame`, or a pandas Series or
    array_like object. Values from this column or array_like are used to
    provide object-constancy across animation frames: rows with matching
    `animation_group`s will be treated as if they describe the same object
    in each frame.
category_orders: dict with str keys and list of str values (default `{}`)
    By default, in Python 3.6+, the order of categorical values in axes,
```

```
    legends and facets depends on the order in which these values are first
    encountered in `data_frame` (and no order is guaranteed by default in
    Python below 3.6). This parameter is used to force a specific ordering
    of values per column. The keys of this dict should correspond to column
    names, and the values should be lists of strings corresponding to the
    specific display order desired.
labels: dict with str keys and str values (default `{}`)
    By default, column names are used in the figure for axis titles, legend
    entries and hovers. This parameter allows this to be overridden. The
    keys of this dict should correspond to column names, and the values
    should correspond to the desired label to be displayed.
color_discrete_sequence: list of str
    Strings should define valid CSS-colors. When `color` is set and the
    values in the corresponding column are not numeric, values in that
    column are assigned colors by cycling through `color_discrete_sequence`
    in the order described in `category_orders`, unless the value of
    `color` is a key in `color_discrete_map`. Various useful color
    sequences are available in the `plotly.express.colors` submodules,
    specifically `plotly.express.colors.qualitative`.
color_discrete_map: dict with str keys and str values (default `{}`)
    String values should define valid CSS-colors Used to override
    `color_discrete_sequence` to assign a specific colors to marks
    corresponding with specific values. Keys in `color_discrete_map` should
    be values in the column denoted by `color`. Alternatively, if the
    values of `color` are valid colors, the string `'identity'` may be
    passed to cause them to be used directly.
color_continuous_scale: list of str
    Strings should define valid CSS-colors This list is used to build a
    continuous color scale when the column denoted by `color` contains
    numeric data. Various useful color scales are available in the
    `plotly.express.colors` submodules, specifically
    `plotly.express.colors.sequential`, `plotly.express.colors.diverging`
    and `plotly.express.colors.cyclical`.
range_color: list of two numbers
    If provided, overrides auto-scaling on the continuous color scale.
color_continuous_midpoint: number (default `None`)
    If set, computes the bounds of the continuous color scale to have the
    desired midpoint. Setting this value is recommended when using
    `plotly.express.colors.diverging` color scales as the inputs to
    `color_continuous_scale`.
projection: str
    One of `'equirectangular'`, `'mercator'`, `'orthographic'`, `'natural
    earth'`, `'kavrayskiy7'`, `'miller'`, `'robinson'`, `'eckert4'`,
    `'azimuthal equal area'`, `'azimuthal equidistant'`, `'conic equal
    area'`, `'conic conformal'`, `'conic equidistant'`, `'gnomonic'`,
    `'stereographic'`, `'mollweide'`, `'hammer'`, `'transverse mercator'`,
    `'albers usa'`, `'winkel tripel'`, `'aitoff'`, or `'sinusoidal'`Default
    depends on `scope`.
```

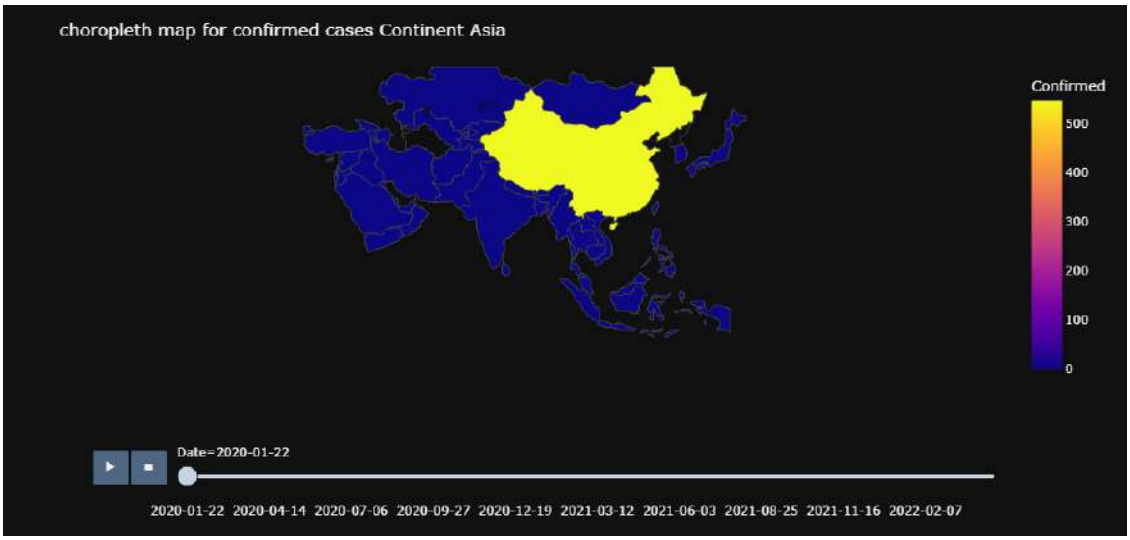
```
    scope: str (default `'world'`).
        One of `'world'`, `'usa'`, `'europe'`, `'asia'`, `'africa'`, `'north
        america'`, or `'south america'`Default is `'world'` unless `projection`
        is set to `'albers usa'`, which forces `'usa'`.
    center: dict
        Dict keys are `'lat'` and `'lon'` Sets the center point of the map.
    fitbounds: str (default `False`).
        One of `False`, `locations` or `geojson`.
    basemap_visible: bool
        Force the basemap visibility.
    title: str
        The figure title.
    template: str or dict or plotly.graph_objects.layout.Template instance
        The figure template name (must be a key in plotly.io.templates) or
        definition.
    width: int (default `None`)
        The figure width in pixels.
    height: int (default `None`)
        The figure height in pixels.

    Returns
    -------
        plotly.graph_objects.Figure
```

2.1 scope: str (default 'world').

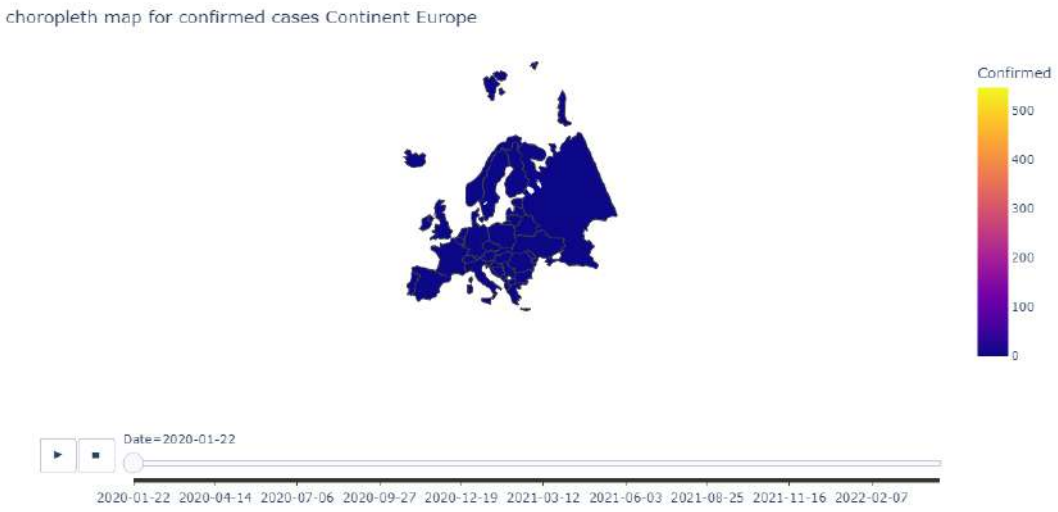
```
    One of `'world'`, `'usa'`, `'europe'`, `'asia'`, `'africa'`, `'north
    america'`, or `'south america'`Default is `'world'` unless `projection`
    is set to `'albers usa'`, which forces `'usa'`.
```

[12]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Confirmed',animation_frame='Date',scope='asia')
fig.update_layout(title='choropleth map for confirmed cases Continent␣
 ↪Asia',template='plotly_dark',width=1000,height=520,autosize=False,)
fig.show()
```



[13]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Confirmed',animation_frame='Date',scope='usa')
fig.update_layout(title='choropleth map for confirmed cases Continent␣
 ↪USA',template='plotly_white',width=1000,height=520,autosize=False,)
fig.show()
```

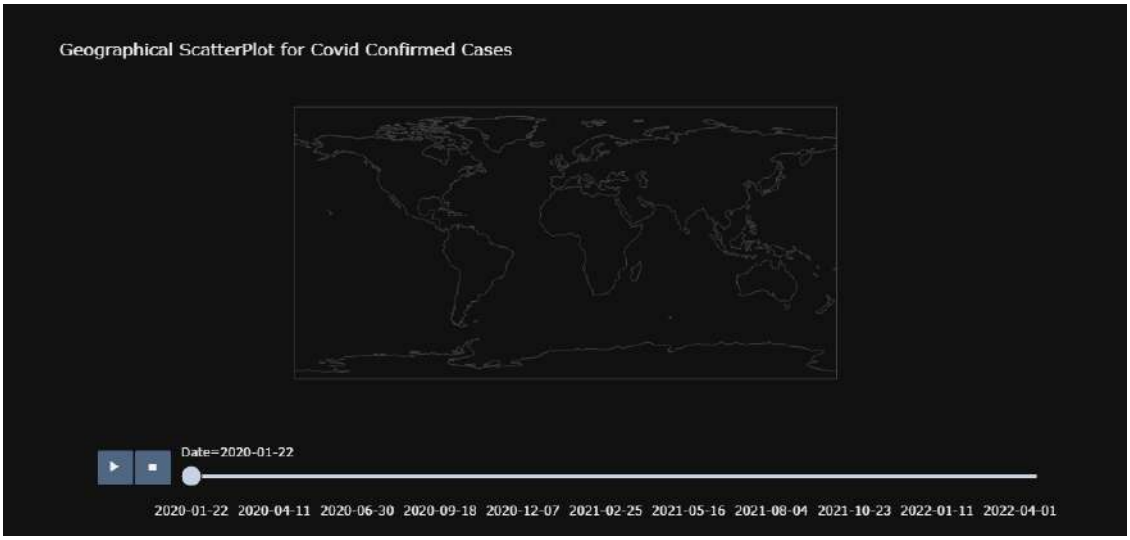
[14]:
```
fig = px.choropleth(current_data,locations='Country',locationmode='country␣
 ↪names',color='Confirmed',animation_frame='Date',scope='europe')
fig.update_layout(title='choropleth map for confirmed cases Continent␣
 ↪Europe',template='plotly_white',width=1000,height=520,autosize=False,)
fig.show()
```



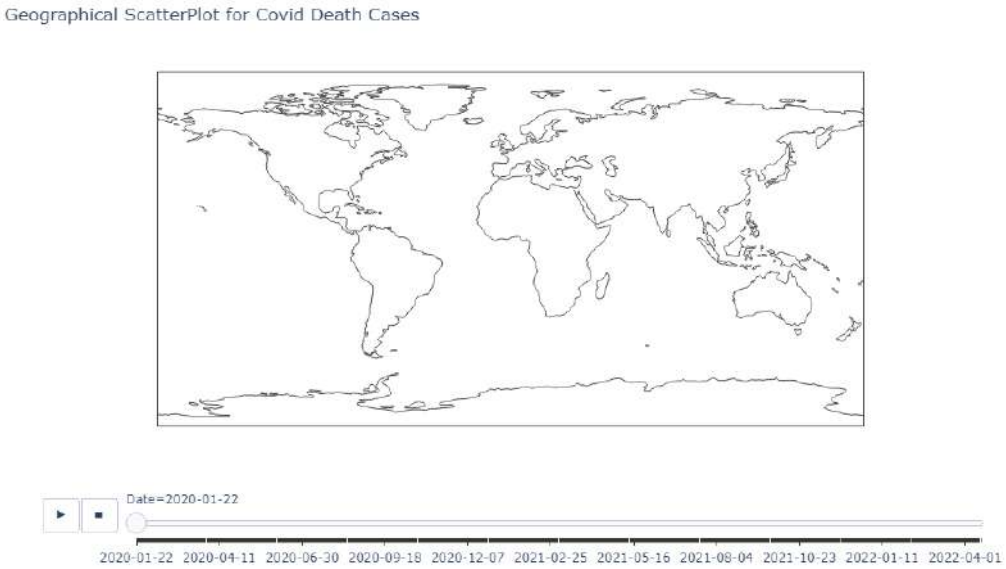
3 Geographical Scatterplot?

3.1 Data points (x,y) in form of (latitude,longitude).Providing (latitude,longitude) in form of location.

[15]:
```
fig = px.scatter_geo(current_data,locations='Country',locationmode='country␣
 ↪names',color='Confirmed',size='Confirmed',hover_name='Country',animation_frame='Date',title
 ↪ScatterPlot for Covid Confirmed Cases')
fig.update(layout_coloraxis_showscale=␣
 ↪False,layout_template='plotly_dark',layout_width=1100,layout_height=520,layout_autosize=Fal
fig.show()
```



[16]:
```
fig = px.scatter_geo(current_data,locations='Country',locationmode='country␣
↪names',color='Deaths',size='Deaths',hover_name='Country',animation_frame='Date',title='Geog
↪ScatterPlot for Covid Death Cases')
fig.update(layout_coloraxis_showscale=␣
↪False,layout_template='plotly_white',layout_width=1100,layout_height=600,layout_autosize=Fa
fig.show()
```



4 How geopy works?

[17]:
```
!pip install geopy
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: geopy in /home/samim/.local/lib/python3.10/site-
packages (2.4.0)
Requirement already satisfied: geographiclib<3,>=1.52 in
/home/samim/.local/lib/python3.10/site-packages (from geopy) (2.0)
```

[18]:
```
import geopy
from geopy.geocoders import Nominatim
geolocator = Nominatim(user_agent='app')
```

[19]:
```
location = geolocator.geocode('chhend , Rourkela, Odisha, 769015 , India')
```

[20]:
```
location.latitude
```

[20]:
```
22.2408446
```

[21]:
```
location.longitude
```

[21]:
```
84.8169946
```

5 Data Preparation for Spatial Analysis?

[22]:
```
df = current_data.copy()
df.head()
```

[22]:

	Date	Country	Confirmed	Recovered	Deaths
0	2020-01-22	Afghanistan	0	0	0
1	2020-01-23	Afghanistan	0	0	0
2	2020-01-24	Afghanistan	0	0	0
3	2020-01-25	Afghanistan	0	0	0
4	2020-01-26	Afghanistan	0	0	0

[23]:
```
df['Country']=='Afghanistan'
```

[23]:
```
0          True
1          True
2          True
3          True
4          True
          …
161563    False
161564    False
161565    False
161566    False
161567    False
Name: Country, Length: 161568, dtype: bool
```

[24]:
```
df[df['Country']=='Afghanistan']
```

[24]:

	Date	Country	Confirmed	Recovered	Deaths
0	2020-01-22	Afghanistan	0	0	0
1	2020-01-23	Afghanistan	0	0	0
2	2020-01-24	Afghanistan	0	0	0
3	2020-01-25	Afghanistan	0	0	0
4	2020-01-26	Afghanistan	0	0	0
..	…	…	…	…	…
811	2022-04-12	Afghanistan	178257	0	7676
812	2022-04-13	Afghanistan	178295	0	7676
813	2022-04-14	Afghanistan	178352	0	7676
814	2022-04-15	Afghanistan	178373	0	7676

```
815  2022-04-16  Afghanistan     178387          0    7676

[816 rows x 5 columns]
```

[25]:
```
df.groupby(['Country'])[['Confirmed','Recovered','Deaths']]
```

[25]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x7fc71cd0e710>
```

[26]:
```
df.groupby(['Country'])[['Confirmed','Recovered','Deaths']].max()
```

[26]:
```
                      Confirmed  Recovered  Deaths
Country
Afghanistan              178387      82586    7676
Albania                  274462     130314    3496
Algeria                  265739     118409    6874
Andorra                   40709      14380     155
Angola                    99194      39582    1900
…                           …          …       …
West Bank and Gaza       656617     312320    5656
Winter Olympics 2022        535          0       0
Yemen                     11817       4251    2148
Zambia                   318467     189658    3973
Zimbabwe                 247237      82994    5462

[198 rows x 3 columns]
```

[27]:
```
df.groupby(['Country'])[['Confirmed','Recovered','Deaths']].max().reset_index()
```

[27]:
```
                  Country  Confirmed  Recovered  Deaths
0             Afghanistan     178387      82586    7676
1                 Albania     274462     130314    3496
2                 Algeria     265739     118409    6874
3                 Andorra      40709      14380     155
4                  Angola      99194      39582    1900
..                    …          …          …       …
193    West Bank and Gaza     656617     312320    5656
194  Winter Olympics 2022        535          0       0
195                 Yemen      11817       4251    2148
196                Zambia     318467     189658    3973
197              Zimbabwe     247237      82994    5462

[198 rows x 4 columns]
```

[28]:
```
df2 = df.groupby(['Country'])[['Confirmed','Recovered','Deaths']].max().
 ↪reset_index()
df2.head(10)
```

[28]:

	Country	Confirmed	Recovered	Deaths
0	Afghanistan	178387	82586	7676
1	Albania	274462	130314	3496
2	Algeria	265739	118409	6874
3	Andorra	40709	14380	155
4	Angola	99194	39582	1900
5	Antarctica	11	0	0
6	Antigua and Barbuda	7535	1239	135
7	Argentina	9060495	4615834	128344
8	Armenia	422747	220438	8621
9	Australia	5384615	24203	6779

[29]:
```
import geopy
from geopy.geocoders import Nominatim
geolocator = Nominatim(user_agent='app')
```

[30]:
```
lat_lon = []
for location in df2['Country']:
    location = geolocator.geocode(location)

    if location is None:
        lat_lon.append(np.nan)
    else:
        geo = (location.latitude,location.longitude)
        lat_lon.append(geo)
        print(geo)
```

```
(33.7680065, 66.2385139)
(11.244803399999999, -72.51609706675976)
(28.0000272, 2.9999825)
(42.5407167, 1.5732033)
(-11.8775768, 17.5691241)
(-72.8438691, 0.0)
(17.2234721, -61.9554608)
(-34.9964963, -64.9672817)
(4.536307, -75.6723751)
(-24.7761086, 134.755)
(47.59397, 14.12456)
(40.3936294, 47.7872508)
(24.7736546, -78.0000547)
(26.0280409, 50.55316820508136)
(-0.2864982, 36.0514231)
(13.1500331, -59.5250305)
(53.4250605, 27.6971358)
(50.6402809, 4.6667145)
(16.8259793, -88.7600927)
(9.5293472, 2.2584408)
```

(33.0204804, 75.6458366)
(-17.0568696, -64.9912286)
(44.3053476, 17.5961467)
(-23.1681782, 24.5928742)
(-10.3333333, -53.2)
(4.4137155, 114.5653908)
(46.7889169, 23.6184909)
(12.0753083, -1.6880314)
(48.9370618, 72.8337335)
(-3.426449, 29.9324519)
(16.0000552, -24.0083947)
(3.9767059, -73.1493675)
(4.6125522, 13.1535811)
(61.0666922, -107.991707)
(7.0323598, 19.9981227)
(15.6134137, 19.0156172)
(-31.7613365, -71.3187697)
(25.491579899999998, -98.98111150455688)
(4.099917, -72.9088133)
(-12.2045176, 44.2832964)
(-0.7264327, 15.6419155)
(-2.9814344, 23.8222636)
(9.536456900000001, -84.17566257468567)
(7.9897371, -5.5679458)
(45.3658443, 15.6575209)
(23.0131338, -80.8328748)
(34.9174159, 32.889902651331866)
(49.7439047, 15.3381061)
(55.670249, 10.3333283)
(53.8953584, 27.5554078)
(11.8145966, 42.8453061)
(15.4113138, -61.3653618)
(19.0974031, -70.3028026)
(-1.3397668, -79.3666965)
(35.8681298, -90.9456751)
(13.8000382, -88.9140683)
(1.613172, 10.5170357)
(15.9500319, 37.9999668)
(58.7523778, 25.3319078)
(-26.5624806, 31.3991317)
(14.4823769, 121.0340788)
(-18.1239696, 179.0122737)
(63.2467777, 25.9209164)
(46.603354, 1.8883335)
(-0.8999695, 11.6899699)
(13.470062, -15.4900464)
(32.3293809, -83.1137366)
(40.4203479, -79.1166983)

(8.0300284, -1.0800271)
(43.2097838, -77.6930602)
(12.1360374, -61.6904045)
(15.5855545, -90.345759)
(10.7226226, -10.7083587)
(11.815215, -15.2351044)
(4.8417097, -58.6416891)
(19.1399952, -72.3570972)
(30.0341947, -95.8116685)
(15.2572432, -86.0755145)
(41.9767629, -72.7789842)
(64.9841821, -18.1059013)
(22.3511148, 78.6677428)
(-2.4833826, 117.8902853)
(-12.853783, -73.6051796)
(33.0955793, 44.1749775)
(52.865196, -7.9794599)
(39.3181528, -79.8109014)
(42.6384261, 12.674297)
(18.1850507, -77.3947693)
(36.5748441, 139.2394179)
(32.3635964, 35.561242)
(48.1012954, 66.7780818)
(1.4419683, 38.4313975)
(0.3448612, 173.6641773)
(-30.5159999, 151.6652595)
(42.5869578, 20.9021231)
(29.3796532, 47.9734174)
(42.4858224, 74.714583)
(40.499847, 37.670719)
(56.8406494, 24.7537645)
(40.375713, -76.4626118)
(-29.6039267, 28.3350193)
(5.7499721, -9.3658524)
(26.8234472, 18.1236723)
(47.1416307, 9.5531527)
(55.3500003, 23.7499997)
(49.6112768, 6.129799)
(52.4424926, 4.8298607)
(-18.9249604, 46.4416422)
(-13.2687204, 33.9301963)
(4.5693754, 102.2656823)
(3.7203503, 73.2244152)
(16.3700359, -2.2900239)
(35.8885993, 14.4476911)
(8.230816999999998, 167.7953223704529)
(25.21534785, 55.16782550577598)
(-20.2759451, 57.5703566)

(19.4326296, -99.1331785)
(8.6062347, 151.832744331612)
(47.2879608, 28.5670941)
(43.7323492, 7.4276832)
(43.9382593, -79.2235563)
(-12.6998188, -38.3260762)
(28.3347722, -10.371337908392647)
(-19.302233, 34.9144977)
(-23.2335499, 17.3231107)
(-6.4955532, 110.8440229)
(52.2434979, 5.6343227)
(-41.5000831, 172.8344077)
(12.6090157, -85.2936911)
(17.7356214, 9.3238432)
(9.6000359, 7.9999721)
(41.6171214, 21.7168387)
(64.5731537, 11.52803643954819)
(42.2679001, 26.9246399)
(30.3308401, 71.247499)
(42.5717989, 2.9600905)
(8.559559, -81.1308434)
(-5.6816069, 144.2489081)
(-23.3165935, -58.1693445)
(-6.8699697, -75.0458515)
(12.7503486, 122.7312101)
(52.215933, 19.134422)
(39.6621648, -8.1353519)
(39.1074426, 47.5061085)
(45.9852129, 24.6859225)
(40.2338211, -84.4096729)
(-1.9646631, 30.0644358)
(17.250512, -62.6725973)
(13.8250489, -60.975036)
(12.90447, -61.2765569)
(-13.7693895, -172.12005)
(43.9458623, 12.458306)
(0.9713095, 7.02255)
(25.6242618, 42.3528328)
(14.4750607, -14.4529612)
(44.024322850000004, 21.07657433209902)
(-4.6574977, 55.4540146)
(8.6400349, -11.8400269)
(1.357107, 103.8194992)
(48.7411522, 19.4528646)
(46.1199444, 14.8153333)
(-8.7053941, 159.1070693851845)
(8.3676771, 49.083416)
(-28.8166236, 24.991639)

```
(7.8699431, 29.6667897)
(39.3260685, -4.8379791)
(7.5554942, 80.7137847)
(10.9, 6.5)
(50.4484727, 30.259556)
(4.1413025, -56.0771187)
(44.133435, -70.822678)
(46.7985624, 8.2319736)
(35.8323648, 38.5414697)
(23.5983227, 120.83537694479215)
(38.6281733, 70.8156541)
(-6.5247123, 35.7878438)
(13.03876215, 101.70017611907599)
(-8.7443169, 126.063482)
(8.7800265, 1.0199765)
(-19.9160819, -175.202642)
(10.7466905, -61.0840075)
(36.8002068, 10.1857757)
(38.9597594, 34.9249653)
(50.5663266, 13.820670539566608)
(1.5333554, 32.2166578)
(49.4871968, 31.2718321)
(24.0002488, 53.9994829)
(54.7023545, -3.2765753)
(-32.8755548, -56.0201525)
(41.32373, 63.9528098)
(-16.5255069, 168.1069154)
(8.0018709, -66.1109318)
(15.9266657, 107.9650855)
(31.9049661, 35.2023413)
(45.74693565, 126.69649301779177)
(16.3471243, 47.8915271)
(-14.5189121, 27.5589884)
(-18.4554963, 29.7468414)
```

[31]:
```
df2['geo_loc'] = lat_lon
df2.head()
```

[31]:

	Country	Confirmed	Recovered	Deaths	\
0	Afghanistan	178387	82586	7676	
1	Albania	274462	130314	3496	
2	Algeria	265739	118409	6874	
3	Andorra	40709	14380	155	
4	Angola	99194	39582	1900	

	geo_loc
0	(33.7680065, 66.2385139)

```
1  (11.244803399999999, -72.51609706675976)
2                    (28.0000272, 2.9999825)
3                    (42.5407167, 1.5732033)
4                  (-11.8775768, 17.5691241)
```

[32]:
```
type(df2['geo_loc'])
```

[32]:
```
pandas.core.series.Series
```

[33]:
```
df2['geo_loc'][0]
```

[33]:
```
(33.7680065, 66.2385139)
```

[34]:
```
type(df2['geo_loc'][0])
```

[34]:
```
tuple
```

[35]:
```
lat,lon = zip(*np.array(df2['geo_loc']))
```

[36]:
```
type(lat)
```

[36]:
```
tuple
```

[37]:
```
type(lon)
```

[37]:
```
tuple
```

[38]:
```
lat
```

[38]:
```
(33.7680065,
 11.244803399999999,
 28.0000272,
 42.5407167,
 -11.8775768,
 -72.8438691,
 17.2234721,
 -34.9964963,
 4.536307,
 -24.7761086,
 47.59397,
 40.3936294,
 24.7736546,
 26.0280409,
 -0.2864982,
 13.1500331,
 53.4250605,
 50.6402809,
```

```
16.8259793,
9.5293472,
33.0204804,
-17.0568696,
44.3053476,
-23.1681782,
-10.3333333,
4.4137155,
46.7889169,
12.0753083,
48.9370618,
-3.426449,
16.0000552,
3.9767059,
4.6125522,
61.0666922,
7.0323598,
15.6134137,
-31.7613365,
25.491579899999998,
4.099917,
-12.2045176,
-0.7264327,
-2.9814344,
9.536456900000001,
7.9897371,
45.3658443,
23.0131338,
34.9174159,
49.7439047,
55.670249,
53.8953584,
11.8145966,
15.4113138,
19.0974031,
-1.3397668,
35.8681298,
13.8000382,
1.613172,
15.9500319,
58.7523778,
-26.5624806,
14.4823769,
-18.1239696,
63.2467777,
46.603354,
-0.8999695,
```

```
13.470062,
32.3293809,
40.4203479,
8.0300284,
43.2097838,
12.1360374,
15.5855545,
10.7226226,
11.815215,
4.8417097,
19.1399952,
30.0341947,
15.2572432,
41.9767629,
64.9841821,
22.3511148,
-2.4833826,
-12.853783,
33.0955793,
52.865196,
39.3181528,
42.6384261,
18.1850507,
36.5748441,
32.3635964,
48.1012954,
1.4419683,
0.3448612,
-30.5159999,
42.5869578,
29.3796532,
42.4858224,
40.499847,
56.8406494,
40.375713,
-29.6039267,
5.7499721,
26.8234472,
47.1416307,
55.3500003,
49.6112768,
52.4424926,
-18.9249604,
-13.2687204,
4.5693754,
3.7203503,
16.3700359,
```

```
35.8885993,
8.230816999999998,
25.21534785,
-20.2759451,
19.4326296,
8.6062347,
47.2879608,
43.7323492,
43.9382593,
-12.6998188,
28.3347722,
-19.302233,
-23.2335499,
-6.4955532,
52.2434979,
-41.5000831,
12.6090157,
17.7356214,
9.6000359,
41.6171214,
64.5731537,
42.2679001,
30.3308401,
42.5717989,
8.559559,
-5.6816069,
-23.3165935,
-6.8699697,
12.7503486,
52.215933,
39.6621648,
39.1074426,
45.9852129,
40.2338211,
-1.9646631,
17.250512,
13.8250489,
12.90447,
-13.7693895,
43.9458623,
0.9713095,
25.6242618,
14.4750607,
44.024322850000004,
-4.6574977,
8.6400349,
1.357107,
```

```
 48.7411522,
 46.1199444,
 -8.7053941,
 8.3676771,
 -28.8166236,
 7.8699431,
 39.3260685,
 7.5554942,
 10.9,
 50.4484727,
 4.1413025,
 44.133435,
 46.7985624,
 35.8323648,
 23.5983227,
 38.6281733,
 -6.5247123,
 13.03876215,
 -8.7443169,
 8.7800265,
 -19.9160819,
 10.7466905,
 36.8002068,
 38.9597594,
 50.5663266,
 1.5333554,
 49.4871968,
 24.0002488,
 54.7023545,
 -32.8755548,
 41.32373,
 -16.5255069,
 8.0018709,
 15.9266657,
 31.9049661,
 45.74693565,
 16.3471243,
 -14.5189121,
 -18.4554963)
```

[39]:
```
df2['lat'] = lat
df2['lon'] = lon
df2.head()
```

[39]:

	Country	Confirmed	Recovered	Deaths	\
0	Afghanistan	178387	82586	7676	
1	Albania	274462	130314	3496	

```
2      Algeria     265739     118409    6874
3      Andorra      40709      14380     155
4       Angola      99194      39582    1900

                                    geo_loc        lat        lon
0                  (33.7680065, 66.2385139)  33.768006  66.238514
1  (11.244803399999999, -72.51609706675976)  11.244803 -72.516097
2                   (28.0000272, 2.9999825)  28.000027   2.999983
3                   (42.5407167, 1.5732033)  42.540717   1.573203
4                 (-11.8775768, 17.5691241) -11.877577  17.569124
```

[40]:
```
df2.drop('geo_loc',axis=1,inplace=True)

df2.head()
```

[40]:
```
       Country  Confirmed  Recovered  Deaths        lat        lon
0  Afghanistan     178387      82586    7676  33.768006  66.238514
1      Albania     274462     130314    3496  11.244803 -72.516097
2      Algeria     265739     118409    6874  28.000027   2.999983
3      Andorra      40709      14380     155  42.540717   1.573203
4       Angola      99194      39582    1900 -11.877577  17.569124
```

6 Idea behind Tileset, Raster and Vector data?

6.1 Tileset is a collection of Raster or Vector data broken into a uniform grid of square tiles.

6.2 Raster is consists of matrix of cells or pixels organized into rows and columns(or a grid) , where each cell contains a value representing information.

6.3 Vector is a data structure used to store spital data . it comprised of lines or arcs , defined by begining and end points.

[41]:
```
!pip install rasterio
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: rasterio in
/home/samim/.local/lib/python3.10/site-packages (1.3.8)
Requirement already satisfied: cligj>=0.5 in /usr/local/lib/python3.10/dist-
packages (from rasterio) (0.7.2)
Requirement already satisfied: affine in /home/samim/.local/lib/python3.10/site-
packages (from rasterio) (2.4.0)
Requirement already satisfied: numpy>=1.18 in /usr/local/lib/python3.10/dist-
packages (from rasterio) (1.25.2)
Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages
(from rasterio) (59.6.0)
Requirement already satisfied: snuggs>=1.4.1 in
```

```
/home/samim/.local/lib/python3.10/site-packages (from rasterio) (1.4.7)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-
packages (from rasterio) (2023.7.22)
Requirement already satisfied: click-plugins in /usr/local/lib/python3.10/dist-
packages (from rasterio) (1.1.1)
Requirement already satisfied: attrs in /usr/local/lib/python3.10/dist-packages
(from rasterio) (23.1.0)
Requirement already satisfied: click>=4.0 in /usr/local/lib/python3.10/dist-
packages (from rasterio) (8.1.7)
Requirement already satisfied: pyparsing>=2.1.6 in /usr/lib/python3/dist-
packages (from snuggs>=1.4.1->rasterio) (2.4.7)
```

[42]:
```
from scipy.interpolate import griddata
import rasterio
```

[43]:
```
#define raster resolution
rRes = 50
```

7 Marker use cases?

7.1 A Marker identifies a location in a map.

7.2 How to plot Marker on a map? you need to create a basemap on which your markers will be placed. Then add your Markers onto it.

[44]:
```
!pip install folium
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: folium in /usr/local/lib/python3.10/dist-packages
(0.14.0)
Requirement already satisfied: jinja2>=2.9 in /usr/local/lib/python3.10/dist-
packages (from folium) (3.1.2)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-
packages (from folium) (2.31.0)
Requirement already satisfied: branca>=0.6.0 in /usr/local/lib/python3.10/dist-
packages (from folium) (0.6.0)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages
(from folium) (1.25.2)
Requirement already satisfied: MarkupSafe>=2.0 in
/usr/local/lib/python3.10/dist-packages (from jinja2>=2.9->folium) (2.1.3)
Requirement already satisfied: charset-normalizer<4,>=2 in
/usr/local/lib/python3.10/dist-packages (from requests->folium) (3.2.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-
packages (from requests->folium) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/usr/local/lib/python3.10/dist-packages (from requests->folium) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.10/dist-packages (from requests->folium) (2023.7.22)
```

[45]:
```
import folium
basemap = folium.Map(location=[54,15],zoom_start=2,tiles='openstreetmap')
basemap
```

[45]:
```
<folium.folium.Map at 0x7fc71ccd3670>
```

7.3 shift + tab to view command attributes.

[46]:
```
for id,row in df2.iterrows():
    print(id)
    print(row)
```

```
0
Country      Afghanistan
Confirmed         178387
Recovered          82586
Deaths              7676
lat             33.768006
lon             66.238514
Name: 0, dtype: object
1
Country        Albania
Confirmed       274462
Recovered       130314
Deaths            3496
lat           11.244803
lon          -72.516097
Name: 1, dtype: object
2
Country        Algeria
Confirmed       265739
Recovered       118409
Deaths            6874
lat           28.000027
lon            2.999983
Name: 2, dtype: object
3
Country        Andorra
Confirmed        40709
Recovered        14380
Deaths             155
lat           42.540717
lon            1.573203
Name: 3, dtype: object
4
Country         Angola
Confirmed        99194
```

```
Recovered        39582
Deaths            1900
lat          -11.877577
lon           17.569124
Name: 4, dtype: object
5
Country      Antarctica
Confirmed            11
Recovered             0
Deaths                0
lat           -72.843869
lon                  0.0
Name: 5, dtype: object
6
Country      Antigua and Barbuda
Confirmed                   7535
Recovered                   1239
Deaths                       135
lat                    17.223472
lon                   -61.955461
Name: 6, dtype: object
7
Country      Argentina
Confirmed      9060495
Recovered      4615834
Deaths          128344
lat          -34.996496
lon          -64.967282
Name: 7, dtype: object
8
Country        Armenia
Confirmed       422747
Recovered       220438
Deaths            8621
lat           4.536307
lon          -75.672375
Name: 8, dtype: object
9
Country      Australia
Confirmed      5384615
Recovered        24203
Deaths            6779
lat          -24.776109
lon            134.755
Name: 9, dtype: object
10
Country       Austria
Confirmed     4045809
```

```
Recovered      644388
Deaths          16407
lat          47.59397
lon          14.12456
Name: 10, dtype: object
11
Country      Azerbaijan
Confirmed        792349
Recovered        333694
Deaths             9705
lat           40.393629
lon           47.787251
Name: 11, dtype: object
12
Country        Bahamas
Confirmed        33391
Recovered        12702
Deaths             789
lat          24.773655
lon         -78.000055
Name: 12, dtype: object
13
Country        Bahrain
Confirmed       562759
Recovered       267220
Deaths            1473
lat          26.028041
lon          50.553168
Name: 13, dtype: object
14
Country      Bangladesh
Confirmed       1952275
Recovered       1141157
Deaths            29124
lat           -0.286498
lon           36.051423
Name: 14, dtype: object
15
Country       Barbados
Confirmed        64348
Recovered         4251
Deaths             383
lat          13.150033
lon          -59.52503
Name: 15, dtype: object
16
Country        Belarus
Confirmed       974046
```

```
Recovered       443417
Deaths            6899
lat           53.425061
lon           27.697136
Name: 16, dtype: object
17
Country        Belgium
Confirmed      3972963
Recovered        31130
Deaths           31165
lat           50.640281
lon            4.666715
Name: 17, dtype: object
18
Country         Belize
Confirmed        57331
Recovered        13543
Deaths             672
lat           16.825979
lon          -88.760093
Name: 18, dtype: object
19
Country         Benin
Confirmed       26952
Recovered        8136
Deaths            163
lat           9.529347
lon           2.258441
Name: 19, dtype: object
20
Country         Bhutan
Confirmed        51800
Recovered         2418
Deaths              16
lat            33.02048
lon           75.645837
Name: 20, dtype: object
21
Country        Bolivia
Confirmed       903888
Recovered       411830
Deaths           21903
lat           -17.05687
lon          -64.991229
Name: 21, dtype: object
22
Country      Bosnia and Herzegovina
Confirmed                    376437
```

```
Recovered                   189710
Deaths                       15749
lat                      44.305348
lon                      17.596147
Name: 22, dtype: object
23
Country       Botswana
Confirmed       305859
Recovered        96964
Deaths            2688
lat          -23.168178
lon           24.592874
Name: 23, dtype: object
24
Country         Brazil
Confirmed     30250077
Recovered     17771228
Deaths          662185
lat          -10.333333
lon              -53.2
Name: 24, dtype: object
25
Country          Brunei
Confirmed        139847
Recovered           280
Deaths              217
lat            4.413716
lon          114.565391
Name: 25, dtype: object
26
Country       Bulgaria
Confirmed      1149225
Recovered       398721
Deaths           36782
lat           46.788917
lon           23.618491
Name: 26, dtype: object
27
Country      Burkina Faso
Confirmed           20865
Recovered           13385
Deaths                383
lat              12.075308
lon              -1.688031
Name: 27, dtype: object
28
Country          Burma
Confirmed       612545
```

```
Recovered       225849
Deaths           19434
lat          48.937062
lon          72.833733
Name: 28, dtype: object
29
Country        Burundi
Confirmed        38722
Recovered          773
Deaths              38
lat          -3.426449
lon          29.932452
Name: 29, dtype: object
30
Country      Cabo Verde
Confirmed         55988
Recovered         33173
Deaths              401
lat           16.000055
lon          -24.008395
Name: 30, dtype: object
31
Country       Cambodia
Confirmed       136044
Recovered        72803
Deaths            3055
lat           3.976706
lon         -73.149367
Name: 31, dtype: object
32
Country       Cameroon
Confirmed       119780
Recovered        35261
Deaths            1927
lat           4.612552
lon          13.153581
Name: 32, dtype: object
33
Country          Canada
Confirmed       3633948
Recovered       1405971
Deaths            38363
lat           61.066692
lon         -107.991707
Name: 33, dtype: object
34
Country      Central African Republic
Confirmed                       14649
```

```
Recovered                        6859
Deaths                            113
lat                          7.03236
lon                        19.998123
Name: 34, dtype: object
35
Country           Chad
Confirmed         7378
Recovered         4796
Deaths             192
lat          15.613414
lon          19.015617
Name: 35, dtype: object
36
Country          Chile
Confirmed      3528626
Recovered      1575377
Deaths           57231
lat         -31.761336
lon          -71.31877
Name: 36, dtype: object
37
Country          China
Confirmed      1760211
Recovered        99228
Deaths           13748
lat            25.49158
lon          -98.981112
Name: 37, dtype: object
38
Country       Colombia
Confirmed      6089540
Recovered      4615354
Deaths          139745
lat            4.099917
lon          -72.908813
Name: 38, dtype: object
39
Country        Comoros
Confirmed         8100
Recovered         3873
Deaths             160
lat          -12.204518
lon           44.283296
Name: 39, dtype: object
40
Country      Congo (Brazzaville)
Confirmed                  24079
```

```
Recovered                  12421
Deaths                       385
lat                    -0.726433
lon                    15.641915
Name: 40, dtype: object
41
Country      Congo (Kinshasa)
Confirmed               87023
Recovered               30043
Deaths                   1337
lat                 -2.981434
lon                 23.822264
Name: 41, dtype: object
42
Country      Costa Rica
Confirmed        844892
Recovered        334759
Deaths             8357
lat            9.536457
lon          -84.175663
Name: 42, dtype: object
43
Country      Cote d'Ivoire
Confirmed            81857
Recovered            49642
Deaths                 797
lat                7.989737
lon               -5.567946
Name: 43, dtype: object
44
Country        Croatia
Confirmed      1113135
Recovered       354830
Deaths           15725
lat           45.365844
lon           15.657521
Name: 44, dtype: object
45
Country           Cuba
Confirmed      1099444
Recovered       373354
Deaths            8520
lat           23.013134
lon          -80.832875
Name: 45, dtype: object
46
Country         Cyprus
Confirmed       464366
```

```
Recovered        39061
Deaths             993
lat          34.917416
lon          32.889903
Name: 46, dtype: object
47
Country        Czechia
Confirmed      3881644
Recovered      1641321
Deaths           39980
lat          49.743905
lon          15.338106
Name: 47, dtype: object
48
Country        Denmark
Confirmed      3143644
Recovered       307374
Deaths            6034
lat          55.670249
lon          10.333328
Name: 48, dtype: object
49
Country      Diamond Princess
Confirmed                712
Recovered                699
Deaths                    13
lat                53.895358
lon                27.555408
Name: 49, dtype: object
50
Country       Djibouti
Confirmed        15598
Recovered        11491
Deaths             189
lat          11.814597
lon          42.845306
Name: 50, dtype: object
51
Country       Dominica
Confirmed        11988
Recovered          209
Deaths              63
lat          15.411314
lon         -61.365362
Name: 51, dtype: object
52
Country      Dominican Republic
Confirmed                578733
```

```
Recovered                324861
Deaths                     4375
lat                   19.097403
lon                  -70.302803
Name: 52, dtype: object
53
Country        Ecuador
Confirmed       865585
Recovered       443880
Deaths           35513
lat           -1.339767
lon          -79.366697
Name: 53, dtype: object
54
Country          Egypt
Confirmed       511977
Recovered       232179
Deaths           24522
lat            35.86813
lon          -90.945675
Name: 54, dtype: object
55
Country      El Salvador
Confirmed         162089
Recovered          76670
Deaths              4125
lat            13.800038
lon           -88.914068
Name: 55, dtype: object
56
Country      Equatorial Guinea
Confirmed                16001
Recovered                 8709
Deaths                     183
lat                   1.613172
lon                  10.517036
Name: 56, dtype: object
57
Country        Eritrea
Confirmed         9733
Recovered         6475
Deaths             103
lat           15.950032
lon           37.999967
Name: 57, dtype: object
58
Country        Estonia
Confirmed       567183
```

```
Recovered       129183
Deaths            2511
lat           58.752378
lon           25.331908
Name: 58, dtype: object
59
Country       Eswatini
Confirmed        70098
Recovered        22127
Deaths            1395
lat          -26.562481
lon           31.399132
Name: 59, dtype: object
60
Country        Ethiopia
Confirmed        470259
Recovered        264008
Deaths             7509
lat           14.482377
lon          121.034079
Name: 60, dtype: object
61
Country            Fiji
Confirmed         64509
Recovered         10848
Deaths              862
lat           -18.12397
lon          179.012274
Name: 61, dtype: object
62
Country        Finland
Confirmed       949583
Recovered        46000
Deaths            3517
lat           63.246778
lon           25.920916
Name: 62, dtype: object
63
Country         France
Confirmed     27874269
Recovered       415111
Deaths          145159
lat           46.603354
lon            1.888334
Name: 63, dtype: object
64
Country         Gabon
Confirmed       47594
```

```
Recovered       25228
Deaths            303
lat         -0.899969
lon          11.68997
Name: 64, dtype: object
65
Country         Gambia
Confirmed        11994
Recovered         7310
Deaths             365
lat          13.470062
lon         -15.490046
Name: 65, dtype: object
66
Country        Georgia
Confirmed      1652929
Recovered       390827
Deaths           16789
lat          32.329381
lon         -83.113737
Name: 66, dtype: object
67
Country        Germany
Confirmed     23416663
Recovered      3659260
Deaths          132942
lat          40.420348
lon         -79.116698
Name: 67, dtype: object
68
Country         Ghana
Confirmed      161101
Recovered       98633
Deaths           1445
lat          8.030028
lon         -1.080027
Name: 68, dtype: object
69
Country         Greece
Confirmed      3232496
Recovered        93764
Deaths           28537
lat          43.209784
lon          -77.69306
Name: 69, dtype: object
70
Country        Grenada
Confirmed        14165
```

```
Recovered          161
Deaths             219
lat          12.136037
lon         -61.690404
Name: 70, dtype: object
71
Country      Guatemala
Confirmed       837492
Recovered       331374
Deaths           17427
lat          15.585555
lon         -90.345759
Name: 71, dtype: object
72
Country         Guinea
Confirmed        36502
Recovered        24463
Deaths             440
lat          10.722623
lon         -10.708359
Name: 72, dtype: object
73
Country      Guinea-Bissau
Confirmed             8176
Recovered             4027
Deaths                 170
lat              11.815215
lon             -15.235104
Name: 73, dtype: object
74
Country         Guyana
Confirmed        63368
Recovered        22327
Deaths            1228
lat            4.84171
lon         -58.641689
Name: 74, dtype: object
75
Country          Haiti
Confirmed        30594
Recovered        12961
Deaths             876
lat          19.139995
lon         -72.357097
Name: 75, dtype: object
76
Country       Holy See
Confirmed           29
```

```
Recovered           27
Deaths               0
lat          30.034195
lon         -95.811668
Name: 76, dtype: object
77
Country       Honduras
Confirmed       421268
Recovered       102384
Deaths           10971
lat          15.257243
lon         -86.075514
Name: 77, dtype: object
78
Country        Hungary
Confirmed      1879480
Recovered       749773
Deaths           45865
lat          41.976763
lon         -72.778984
Name: 78, dtype: object
79
Country        Iceland
Confirmed       183974
Recovered         6993
Deaths             110
lat          64.984182
lon         -18.105901
Name: 79, dtype: object
80
Country          India
Confirmed     43042097
Recovered     30974748
Deaths          521751
lat          22.351115
lon          78.667743
Name: 80, dtype: object
81
Country       Indonesia
Confirmed       6039266
Recovered       2907920
Deaths           155844
lat           -2.483383
lon          117.890285
Name: 81, dtype: object
82
Country           Iran
Confirmed      7205064
```

```
Recovered      3444798
Deaths          140800
lat          -12.853783
lon           -73.60518
Name: 82, dtype: object
83
Country           Iraq
Confirmed      2323040
Recovered      1494760
Deaths           25198
lat           33.095579
lon           44.174977
Name: 83, dtype: object
84
Country        Ireland
Confirmed      1498834
Recovered        23364
Deaths            6932
lat           52.865196
lon            -7.97946
Name: 84, dtype: object
85
Country         Israel
Confirmed      4029066
Recovered       854888
Deaths           10612
lat           39.318153
lon          -79.810901
Name: 85, dtype: object
86
Country          Italy
Confirmed     15659835
Recovered      4144608
Deaths          161602
lat           42.638426
lon           12.674297
Name: 86, dtype: object
87
Country        Jamaica
Confirmed       129203
Recovered        47101
Deaths            2926
lat           18.185051
lon          -77.394769
Name: 87, dtype: object
88
Country           Japan
Confirmed       7332261
```

```
Recovered        852451
Deaths            28998
lat           36.574844
lon          139.239418
Name: 88, dtype: object
89
Country         Jordan
Confirmed      1694957
Recovered       752624
Deaths           14055
lat          32.363596
lon          35.561242
Name: 89, dtype: object
90
Country      Kazakhstan
Confirmed       1394190
Recovered        555079
Deaths            19013
lat           48.101295
lon           66.778082
Name: 90, dtype: object
91
Country          Kenya
Confirmed       323609
Recovered       191188
Deaths            5649
lat           1.441968
lon          38.431398
Name: 91, dtype: object
92
Country        Kiribati
Confirmed          3071
Recovered             0
Deaths               13
lat            0.344861
lon          173.664177
Name: 92, dtype: object
93
Country      Korea, South
Confirmed        16305752
Recovered          180719
Deaths              21092
lat               -30.516
lon            151.665259
Name: 93, dtype: object
94
Country         Kosovo
Confirmed       227838
```

```
Recovered       105688
Deaths            3138
lat           42.586958
lon           20.902123
Name: 94, dtype: object
95
Country         Kuwait
Confirmed       630888
Recovered       388880
Deaths            2555
lat           29.379653
lon           47.973417
Name: 95, dtype: object
96
Country      Kyrgyzstan
Confirmed        200980
Recovered        150852
Deaths             2991
lat            42.485822
lon            74.714583
Name: 96, dtype: object
97
Country           Laos
Confirmed       199868
Recovered         3804
Deaths             714
lat           40.499847
lon           37.670719
Name: 97, dtype: object
98
Country         Latvia
Confirmed       812877
Recovered       135690
Deaths            5714
lat           56.840649
lon           24.753764
Name: 98, dtype: object
99
Country        Lebanon
Confirmed      1095518
Recovered       537653
Deaths           10358
lat           40.375713
lon          -76.462612
Name: 99, dtype: object
100
Country        Lesotho
Confirmed        32968
```

```
Recovered         6664
Deaths             697
lat          -29.603927
lon           28.335019
Name: 100, dtype: object
101
Country       Liberia
Confirmed        7402
Recovered        2715
Deaths            294
lat          5.749972
lon         -9.365852
Name: 101, dtype: object
102
Country          Libya
Confirmed       501834
Recovered       195639
Deaths            6429
lat          26.823447
lon          18.123672
Name: 102, dtype: object
103
Country      Liechtenstein
Confirmed            16975
Recovered             3011
Deaths                  84
lat              47.141631
lon               9.553153
Name: 103, dtype: object
104
Country      Lithuania
Confirmed      1048704
Recovered       269840
Deaths            9011
lat              55.35
lon              23.75
Name: 104, dtype: object
105
Country      Luxembourg
Confirmed        229311
Recovered         72306
Deaths             1054
lat           49.611277
lon            6.129799
Name: 105, dtype: object
106
Country      MS Zaandam
Confirmed             9
```

```
Recovered             7
Deaths                2
lat           52.442493
lon            4.829861
Name: 106, dtype: object
107
Country      Madagascar
Confirmed         64089
Recovered         41177
Deaths             1390
lat           -18.92496
lon           46.441642
Name: 107, dtype: object
108
Country          Malawi
Confirmed         85727
Recovered         39841
Deaths             2631
lat           -13.26872
lon           33.930196
Name: 108, dtype: object
109
Country         Malaysia
Confirmed        4382402
Recovered         962731
Deaths             35409
lat            4.569375
lon          102.265682
Name: 109, dtype: object
110
Country        Maldives
Confirmed        178320
Recovered         75095
Deaths              298
lat             3.72035
lon           73.224415
Name: 110, dtype: object
111
Country            Mali
Confirmed         30651
Recovered         13962
Deaths              729
lat           16.370036
lon           -2.290024
Name: 111, dtype: object
112
Country           Malta
Confirmed         88558
```

```
Recovered        32438
Deaths             673
lat          35.888599
lon          14.447691
Name: 112, dtype: object
113
Country      Marshall Islands
Confirmed                   7
Recovered                   4
Deaths                      0
lat                  8.230817
lon                167.795322
Name: 113, dtype: object
114
Country      Mauritania
Confirmed         58710
Recovered         22859
Deaths              982
lat           25.215348
lon           55.167826
Name: 114, dtype: object
115
Country      Mauritius
Confirmed       218229
Recovered         1854
Deaths             990
lat         -20.275945
lon          57.570357
Name: 115, dtype: object
116
Country         Mexico
Confirmed      5726668
Recovered      2270427
Deaths          323938
lat           19.43263
lon         -99.133178
Name: 116, dtype: object
117
Country      Micronesia
Confirmed             1
Recovered             1
Deaths                0
lat            8.606235
lon          151.832744
Name: 117, dtype: object
118
Country        Moldova
Confirmed       516316
```

```
Recovered       252421
Deaths           11481
lat           47.287961
lon           28.567094
Name: 118, dtype: object
119
Country         Monaco
Confirmed        11341
Recovered         2759
Deaths              56
lat           43.732349
lon            7.427683
Name: 119, dtype: object
120
Country       Mongolia
Confirmed       920119
Recovered       164829
Deaths            2177
lat           43.938259
lon          -79.223556
Name: 120, dtype: object
121
Country      Montenegro
Confirmed        234137
Recovered         99152
Deaths             2709
lat           -12.699819
lon           -38.326076
Name: 121, dtype: object
122
Country        Morocco
Confirmed      1164345
Recovered       582692
Deaths           16062
lat           28.334772
lon          -10.371338
Name: 122, dtype: object
123
Country      Mozambique
Confirmed        225323
Recovered        100912
Deaths             2200
lat           -19.302233
lon            34.914498
Name: 123, dtype: object
124
Country        Namibia
Confirmed       158074
```

```
Recovered        96568
Deaths            4022
lat           -23.23355
lon           17.323111
Name: 124, dtype: object
125
Country           Nepal
Confirmed        978654
Recovered        661651
Deaths            11951
lat            -6.495553
lon           110.844023
Name: 125, dtype: object
126
Country      Netherlands
Confirmed        8194946
Recovered          28771
Deaths             22780
lat            52.243498
lon             5.634323
Name: 126, dtype: object
127
Country      New Zealand
Confirmed         828808
Recovered           2824
Deaths               554
lat            -41.500083
lon            172.834408
Name: 127, dtype: object
128
Country      Nicaragua
Confirmed        18491
Recovered         4225
Deaths             232
lat           12.609016
lon          -85.293691
Name: 128, dtype: object
129
Country          Niger
Confirmed         8871
Recovered         5351
Deaths             308
lat           17.735621
lon            9.323843
Name: 129, dtype: object
130
Country       Nigeria
Confirmed      255663
```

```
Recovered      165208
Deaths           3143
lat           9.600036
lon           7.999972
Name: 130, dtype: object
131
Country      North Macedonia
Confirmed             308516
Recovered             150440
Deaths                  9261
lat                41.617121
lon                21.716839
Name: 131, dtype: object
132
Country          Norway
Confirmed       1419507
Recovered         17998
Deaths             2783
lat           64.573154
lon           11.528036
Name: 132, dtype: object
133
Country           Oman
Confirmed       388795
Recovered       281724
Deaths            4257
lat            42.2679
lon           26.92464
Name: 133, dtype: object
134
Country       Pakistan
Confirmed      1527248
Recovered       952616
Deaths           30363
lat            30.33084
lon           71.247499
Name: 134, dtype: object
135
Country          Palau
Confirmed         4190
Recovered            0
Deaths               6
lat           42.571799
lon            2.960091
Name: 135, dtype: object
136
Country         Panama
Confirmed       768470
```

```
Recovered       420113
Deaths            8178
lat           8.559559
lon         -81.130843
Name: 136, dtype: object
137
Country      Papua New Guinea
Confirmed               43660
Recovered               17384
Deaths                    649
lat                  -5.681607
lon                 144.248908
Name: 137, dtype: object
138
Country       Paraguay
Confirmed       648446
Recovered       423964
Deaths           18734
lat          -23.316593
lon          -58.169345
Name: 138, dtype: object
139
Country           Peru
Confirmed      3555139
Recovered      2086086
Deaths          212619
lat           -6.86997
lon          -75.045851
Name: 139, dtype: object
140
Country      Philippines
Confirmed        3682847
Recovered        1528422
Deaths             59964
lat            12.750349
lon            122.73121
Name: 140, dtype: object
141
Country         Poland
Confirmed      5984940
Recovered      2653981
Deaths          115838
lat          52.215933
lon          19.134422
Name: 141, dtype: object
142
Country       Portugal
Confirmed      3719485
```

```
Recovered       912620
Deaths           21993
lat           39.662165
lon           -8.135352
Name: 142, dtype: object
143
Country           Qatar
Confirmed        363443
Recovered        224285
Deaths              677
lat           39.107443
lon           47.506109
Name: 143, dtype: object
144
Country         Romania
Confirmed       2881322
Recovered       1048072
Deaths            65331
lat           45.985213
lon           24.685923
Name: 144, dtype: object
145
Country          Russia
Confirmed      17801103
Recovered       5609682
Deaths           365774
lat           40.233821
lon          -84.409673
Name: 145, dtype: object
146
Country          Rwanda
Confirmed        129760
Recovered         44911
Deaths             1459
lat           -1.964663
lon           30.064436
Name: 146, dtype: object
147
Country      Saint Kitts and Nevis
Confirmed                     5557
Recovered                      549
Deaths                          43
lat                      17.250512
lon                     -62.672597
Name: 147, dtype: object
148
Country      Saint Lucia
Confirmed          23094
```

```
Recovered           5398
Deaths               367
lat            13.825049
lon           -60.975036
Name: 148, dtype: object
149
Country      Saint Vincent and the Grenadines
Confirmed                                9447
Recovered                                2233
Deaths                                    106
lat                                 12.90447
lon                                -61.276557
Name: 149, dtype: object
150
Country          Samoa
Confirmed         4793
Recovered            3
Deaths              10
lat           -13.76939
lon          -172.12005
Name: 150, dtype: object
151
Country      San Marino
Confirmed         15874
Recovered          5009
Deaths              114
lat            43.945862
lon            12.458306
Name: 151, dtype: object
152
Country      Sao Tome and Principe
Confirmed                     5948
Recovered                     2365
Deaths                          73
lat                        0.97131
lon                        7.02255
Name: 152, dtype: object
153
Country      Saudi Arabia
Confirmed          752479
Recovered          507374
Deaths               9068
lat             25.624262
lon             42.352833
Name: 153, dtype: object
154
Country        Senegal
Confirmed        85967
```

```
Recovered        48812
Deaths            1988
lat           14.475061
lon          -14.452961
Name: 154, dtype: object
155
Country         Serbia
Confirmed      1995351
Recovered        15564
Deaths           15923
lat           44.024323
lon           21.076574
Name: 155, dtype: object
156
Country      Seychelles
Confirmed         41660
Recovered         17874
Deaths              164
lat            -4.657498
lon            55.454015
Name: 156, dtype: object
157
Country      Sierra Leone
Confirmed            7681
Recovered            4287
Deaths                125
lat                8.640035
lon              -11.840027
Name: 157, dtype: object
158
Country       Singapore
Confirmed       1157251
Recovered         63357
Deaths             1313
lat              1.357107
lon            103.819499
Name: 158, dtype: object
159
Country       Slovakia
Confirmed      2505968
Recovered       255300
Deaths           19721
lat           48.741152
lon           19.452865
Name: 159, dtype: object
160
Country       Slovenia
Confirmed       996832
```

```
Recovered       253972
Deaths            6556
lat           46.119944
lon           14.815333
Name: 160, dtype: object
161
Country      Solomon Islands
Confirmed              12437
Recovered                 20
Deaths                   139
lat                 -8.705394
lon                159.107069
Name: 161, dtype: object
162
Country        Somalia
Confirmed        26471
Recovered         7661
Deaths            1361
lat           8.367677
lon          49.083416
Name: 162, dtype: object
163
Country      South Africa
Confirmed         3740398
Recovered         2258603
Deaths             100144
lat            -28.816624
lon             24.991639
Name: 163, dtype: object
164
Country      South Sudan
Confirmed          17369
Recovered          10514
Deaths               138
lat              7.869943
lon              29.66679
Name: 164, dtype: object
165
Country          Spain
Confirmed     11627487
Recovered       150376
Deaths          103104
lat           39.326068
lon           -4.837979
Name: 165, dtype: object
166
Country      Sri Lanka
Confirmed       662827
```

```
Recovered       284524
Deaths           16495
lat           7.555494
lon          80.713785
Name: 166, dtype: object
167
Country      Sudan
Confirmed    62057
Recovered    30647
Deaths        4929
lat           10.9
lon            6.5
Name: 167, dtype: object
168
Country      Summer Olympics 2020
Confirmed                     865
Recovered                       0
Deaths                          0
lat                     50.448473
lon                     30.259556
Name: 168, dtype: object
169
Country       Suriname
Confirmed        79276
Recovered        21978
Deaths            1325
lat           4.141303
lon         -56.077119
Name: 169, dtype: object
170
Country         Sweden
Confirmed      2495996
Recovered            0
Deaths           18605
lat           44.133435
lon          -70.822678
Name: 170, dtype: object
171
Country      Switzerland
Confirmed        3568616
Recovered         317600
Deaths             13647
lat            46.798562
lon             8.231974
Name: 171, dtype: object
172
Country          Syria
Confirmed        55769
```

```
Recovered        22019
Deaths            3149
lat           35.832365
lon            38.54147
Name: 172, dtype: object
173
Country         Taiwan*
Confirmed         33205
Recovered         12957
Deaths              854
lat            23.598323
lon           120.835377
Name: 173, dtype: object
174
Country      Tajikistan
Confirmed         17786
Recovered         14867
Deaths              125
lat            38.628173
lon            70.815654
Name: 174, dtype: object
175
Country       Tanzania
Confirmed        33851
Recovered          183
Deaths             803
lat           -6.524712
lon           35.787844
Name: 175, dtype: object
176
Country        Thailand
Confirmed       4029959
Recovered         26873
Deaths            26882
lat            13.038762
lon           101.700176
Name: 176, dtype: object
177
Country      Timor-Leste
Confirmed          22853
Recovered          10025
Deaths               130
lat             -8.744317
lon            126.063482
Name: 177, dtype: object
178
Country          Togo
Confirmed       36962
```

```
Recovered       14654
Deaths            273
lat          8.780026
lon          1.019977
Name: 178, dtype: object
179
Country           Tonga
Confirmed          8922
Recovered             0
Deaths               11
lat          -19.916082
lon         -175.202642
Name: 179, dtype: object
180
Country      Trinidad and Tobago
Confirmed                 142076
Recovered                  32454
Deaths                      3800
lat                    10.74669
lon                   -61.084007
Name: 180, dtype: object
181
Country        Tunisia
Confirmed      1038668
Recovered       530545
Deaths           28509
lat          36.800207
lon          10.185776
Name: 181, dtype: object
182
Country         Turkey
Confirmed     14991669
Recovered      5478185
Deaths           98551
lat          38.959759
lon          34.924965
Name: 182, dtype: object
183
Country             US
Confirmed     80625120
Recovered      6298082
Deaths          988609
lat          50.566327
lon          13.820671
Name: 183, dtype: object
184
Country         Uganda
Confirmed       164051
```

```
Recovered        86826
Deaths            3597
lat           1.533355
lon          32.216658
Name: 184, dtype: object
185
Country        Ukraine
Confirmed      5040518
Recovered      2258433
Deaths          112459
lat           49.487197
lon           31.271832
Name: 185, dtype: object
186
Country      United Arab Emirates
Confirmed                  895264
Recovered                  664130
Deaths                       2302
lat                     24.000249
lon                     53.999483
Name: 186, dtype: object
187
Country      United Kingdom
Confirmed          21916961
Recovered             24693
Deaths               172014
lat                54.702354
lon                -3.276575
Name: 187, dtype: object
188
Country        Uruguay
Confirmed       895592
Recovered       374203
Deaths            7193
lat          -32.875555
lon          -56.020153
Name: 188, dtype: object
189
Country      Uzbekistan
Confirmed        238252
Recovered        126377
Deaths             1637
lat             41.32373
lon             63.95281
Name: 189, dtype: object
190
Country         Vanuatu
Confirmed          6314
```

```
Recovered             3
Deaths                7
lat           -16.525507
lon           168.106915
Name: 190, dtype: object
191
Country      Venezuela
Confirmed       522034
Recovered       294607
Deaths            5701
lat            8.001871
lon          -66.110932
Name: 191, dtype: object
192
Country         Vietnam
Confirmed      10417887
Recovered         54332
Deaths            42934
lat            15.926666
lon           107.965086
Name: 192, dtype: object
193
Country      West Bank and Gaza
Confirmed                656617
Recovered                312320
Deaths                     5656
lat                   31.904966
lon                   35.202341
Name: 193, dtype: object
194
Country      Winter Olympics 2022
Confirmed                     535
Recovered                       0
Deaths                          0
lat                     45.746936
lon                    126.696493
Name: 194, dtype: object
195
Country          Yemen
Confirmed        11817
Recovered         4251
Deaths            2148
lat           16.347124
lon           47.891527
Name: 195, dtype: object
196
Country         Zambia
Confirmed       318467
```

```
Recovered       189658
Deaths            3973
lat          -14.518912
lon           27.558988
Name: 196, dtype: object
197
Country       Zimbabwe
Confirmed       247237
Recovered        82994
Deaths            5462
lat          -18.455496
lon           29.746841
Name: 197, dtype: object
```

[47]:
```
# id receive index & row receive rows.

for id,row in df2.iterrows():
    folium.Marker(location=[row['lat'],row['lon']],popup=row['Confirmed']).
 ↪add_to(basemap)

basemap
```

[47]:
```
<folium.folium.Map at 0x7fc71ccd3670>
```

8 folium Marker Cluster?

8.1 Marker clusters can be a good way to simply a map containing many markers. When the map is zoomed out nearby markers are combined together into a cluster, which is separated out when the map zoom level is closer.

[48]:
```
from folium.plugins import MarkerCluster
mc = MarkerCluster()
```

[49]:
```
import folium
m = folium.Map(location=[54,15],zoom_start=2,tiles='openstreetmap')

for id,row in df2.iterrows():
    mc.add_child(folium.
 ↪Marker(location=[row['lat'],row['lon']],popup=row['Confirmed']))

m.add_child(mc)
# save the map object as html
m.save("output.html")
m
```

[49]:
```
<folium.folium.Map at 0x7fc718e348e0>
```

9 Geographic Heatmap?

9.1 Geographic heat map are interactive way to identify where something occurs and demonstrate areas of high and low density.

9.2 In geospatial analysis, a heatmap is a graphical representation of data where values are depicted as colors on a map, with the color intensity varying according to the data's spatial distribution. Heatmaps are used to visualize the density, concentration, or magnitude of certain phenomena or attributes across a geographic area. They can provide valuable insights into patterns, trends, and spatial relationships within geospatial data.

9.2.1 Point Density: Heatmaps can represent the density of point data, such as the distribution of crime incidents, disease outbreaks, or customer locations. Hotspots, where the density is higher, are represented by more intense colors.

9.2.2 Interpolation: Heatmaps can be generated through interpolation techniques to estimate values at unobserved locations. For example, you can create a heatmap of temperature readings across a region using weather station data.

9.2.3 Spatial Patterns: Heatmaps can help identify spatial patterns in data, such as land cover classification or vegetation indices in remote sensing applications. Different colors may represent different land cover types or vegetation health levels.

9.2.4 Accessibility and Usage: In urban planning and transportation analysis, heatmaps can depict the accessibility of different areas or the usage of public transportation services. This information can inform decision-making processes.

9.2.5 Environmental Monitoring: In environmental studies, heatmaps can be used to visualize pollution levels, temperature variations, or other environmental factors over a geographical area.

9.2.6 Risk Assessment: Heatmaps can help assess risks, such as wildfire risk, flood risk, or earthquake intensity, by visualizing the likelihood or impact of events in different areas.

9.3 To create a heatmap in geospatial analysis, you typically need geospatial data (e.g., shapefiles, raster data) and tools or libraries capable of spatial analysis and visualization. You can use libraries like geopandas, rasterio, and visualization libraries such as matplotlib or seaborn in Python to generate heatmaps based on your specific data and analysis goals.

[50]:
```
from folium.plugins import HeatMap
df2.head()
```

[50]:

	Country	Confirmed	Recovered	Deaths	lat	lon
0	Afghanistan	178387	82586	7676	33.768006	66.238514
1	Albania	274462	130314	3496	11.244803	-72.516097
2	Algeria	265739	118409	6874	28.000027	2.999983
3	Andorra	40709	14380	155	42.540717	1.573203

```
4       Angola      99194      39582    1900 -11.877577  17.569124
```

[51]:
```
import folium
m = folium.Map(location=[54,15],zoom_start=2,tiles='openstreetmap')
```

[54]:
```
HeatMap(data=df2[['lat','lon','Confirmed']],radius=15).add_to(m)
m
```

[54]:
```
<folium.folium.Map at 0x7fc71dafc040>
```

[55]:
```
HeatMap(data=df2[['lat','lon','Deaths']],radius=15).add_to(m)
m
```

[55]:
```
<folium.folium.Map at 0x7fc71dafc040>
```

[ ]:
```

```