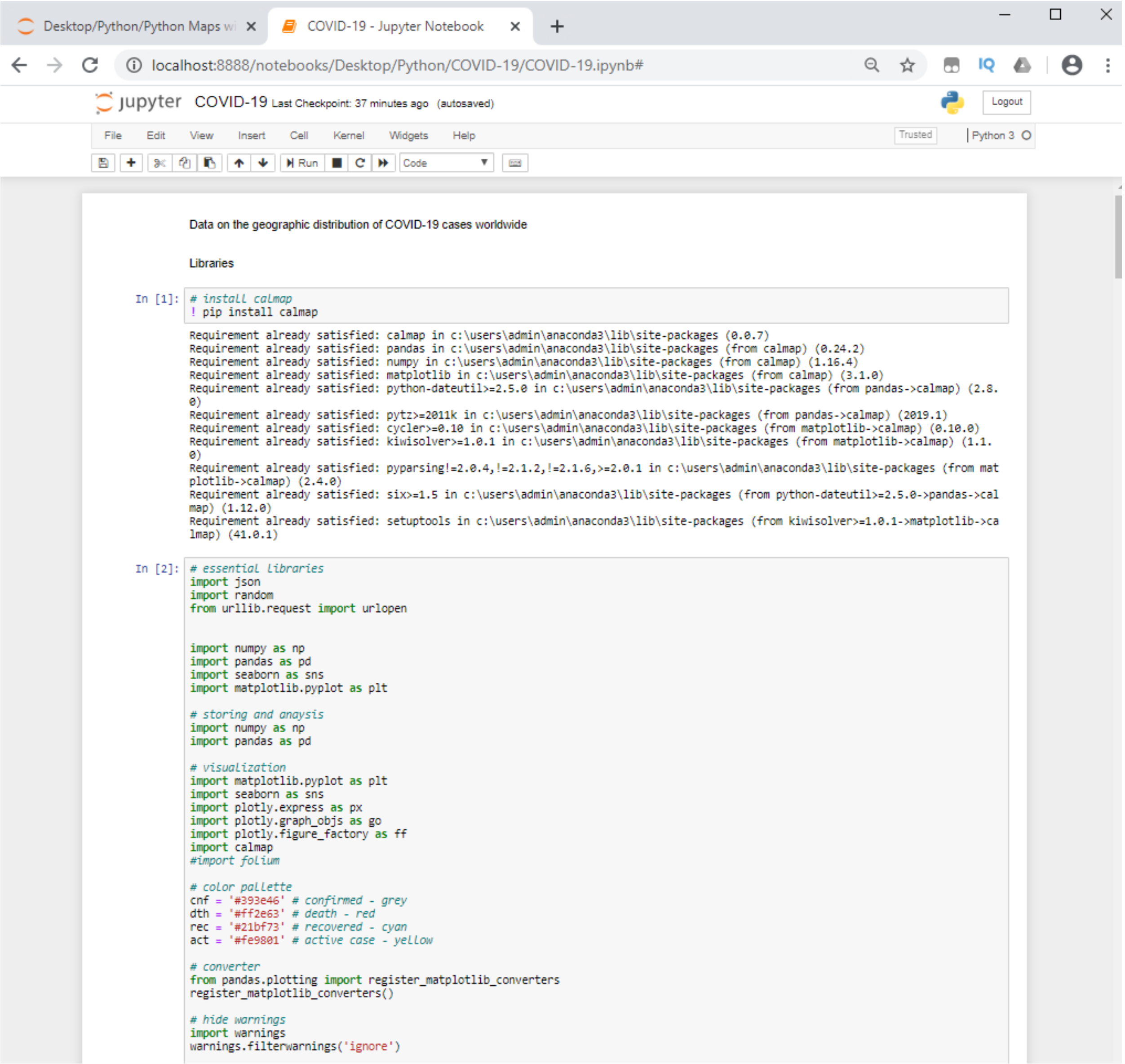

Desktop/Python/Python Maps w
COVID-19 - Jupyter Notebook
localhost:8888/notebooks/Desktop/Python/COVID-19/COVID-19.ipynb#
jupyter COVID-19 Last Checkpoint: 37 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 ○
Run
Code

Data on the geographic distribution of COVID-19 cases worldwide

Libraries

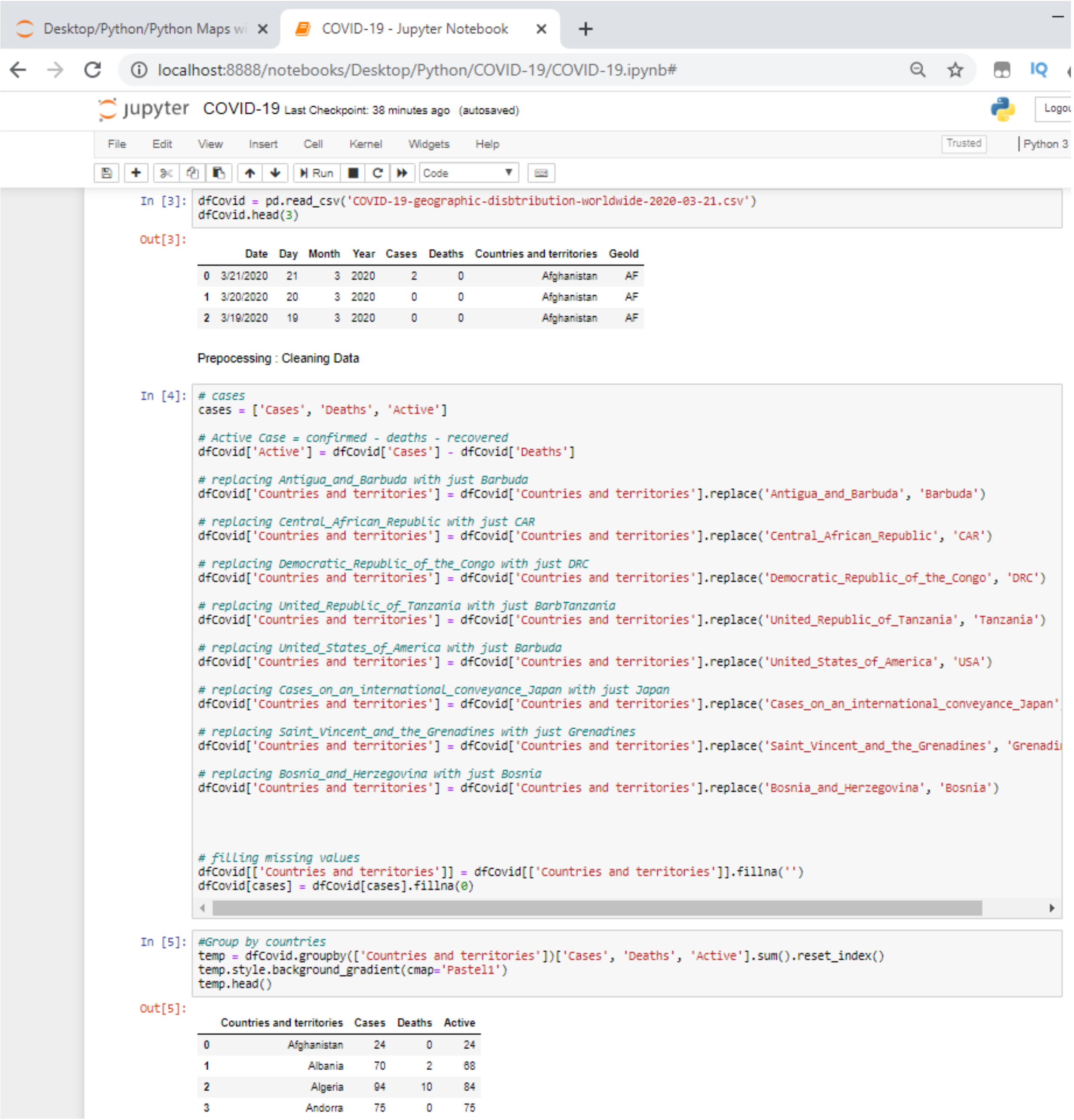
In [1]:

```
# install calmap
! pip install calmap
```

```
Requirement already satisfied: calmap in c:\users\admin\anaconda3\lib\site-packages (0.0.7)
Requirement already satisfied: pandas in c:\users\admin\anaconda3\lib\site-packages (from calmap) (0.24.2)
Requirement already satisfied: numpy in c:\users\admin\anaconda3\lib\site-packages (from calmap) (1.16.4)
Requirement already satisfied: matplotlib in c:\users\admin\anaconda3\lib\site-packages (from calmap) (3.1.0)
Requirement already satisfied: python-dateutil>=2.5.0 in c:\users\admin\anaconda3\lib\site-packages (from pandas->calmap) (2.8.
0)
Requirement already satisfied: pytz>=2011k in c:\users\admin\anaconda3\lib\site-packages (from pandas->calmap) (2019.1)
Requirement already satisfied: cycler>=0.10 in c:\users\admin\anaconda3\lib\site-packages (from matplotlib->calmap) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\admin\anaconda3\lib\site-packages (from matplotlib->calmap) (1.1.
0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users\admin\anaconda3\lib\site-packages (from mat
plotlib->calmap) (2.4.0)
Requirement already satisfied: six>=1.5 in c:\users\admin\anaconda3\lib\site-packages (from python-dateutil>=2.5.0->pandas->cal
map) (1.12.0)
Requirement already satisfied: setuptools in c:\users\admin\anaconda3\lib\site-packages (from kiwisolver>=1.0.1->matplotlib->ca
lmap) (41.0.1)
```

In [2]:

```
# essential libraries
import json
import random
from urllib.request import urlopen


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

# storing and anaysis
import numpy as np
import pandas as pd

# visualization
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import plotly.graph_objs as go
import plotly.figure_factory as ff
import calmap
#import folium

# color pallette
cnf = '#393e46' # confirmed - grey
dth = '#ff2e63' # death - red
rec = '#21bf73' # recovered - cyan
act = '#fe9801' # active case - yellow

# converter
from pandas.plotting import register_matplotlib_converters
register_matplotlib_converters()

# hide warnings
import warnings
warnings.filterwarnings('ignore')
```


Desktop/Python/Python Maps w
COVID-19 - Jupyter Notebook
localhost:8888/notebooks/Desktop/Python/COVID-19/COVID-19.ipynb#
IQ
jupyter
Logou
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

COVID-19 Last Checkpoint: 38 minutes ago (autosaved)

In [3]:

```
dfCovid = pd.read_csv('COVID-19-geographic-disbtribution-worldwide-2020-03-21.csv')
dfCovid.head(3)
```

Out[3]:

	Date	Day	Month	Year	Cases	Deaths	Countries and territories	GeoId
0	3/21/2020	21	3	2020	2	0	Afghanistan	AF
1	3/20/2020	20	3	2020	0	0	Afghanistan	AF
2	3/19/2020	19	3	2020	0	0	Afghanistan	AF

Prepocessing : Cleaning Data

In [4]:

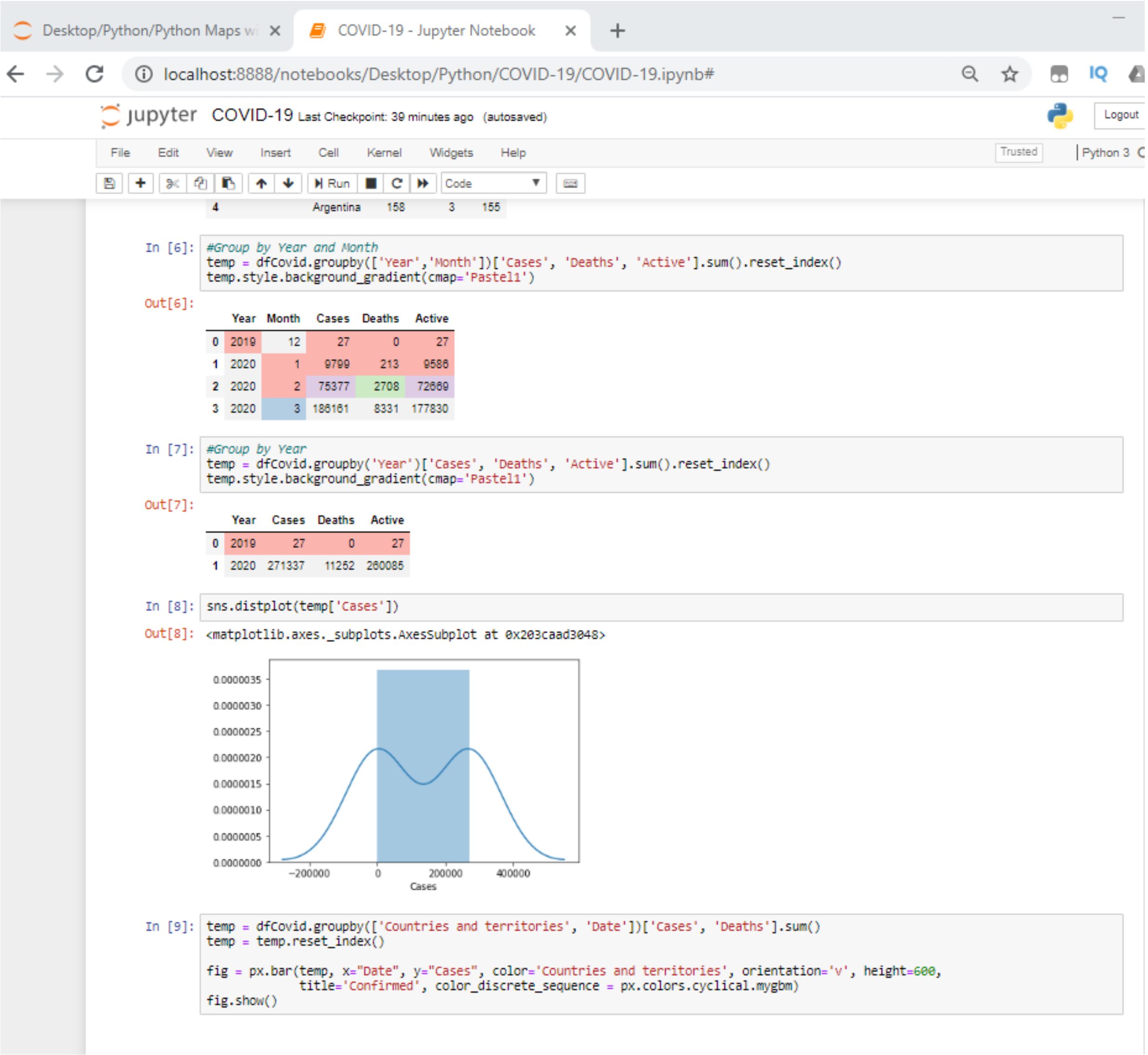
```
# cases
cases = ['Cases', 'Deaths', 'Active']

# Active Case = confirmed - deaths - recovered
dfCovid['Active'] = dfCovid['Cases'] - dfCovid['Deaths']

# replacing Antigua_and_Barbuda with just Barbuda
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Antigua_and_Barbuda', 'Barbuda')

# replacing Central_African_Republic with just CAR
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Central_African_Republic', 'CAR')

# replacing Democratic_Republic_of_the_Congo with just DRC
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Democratic_Republic_of_the_Congo', 'DRC')

# replacing United_Republic_of_Tanzania with just BarbTanzania
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('United_Republic_of_Tanzania', 'Tanzania')

# replacing United_States_of_America with just Barbuda
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('United_States_of_America', 'USA')

# replacing Cases_on_an_international_conveyance_Japan with just Japan
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Cases_on_an_international_conveyance_Japan'

# replacing Saint_Vincent_and_the_Grenadines with just Grenadines
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Saint_Vincent_and_the_Grenadines', 'Grenadi

# replacing Bosnia_and_Herzegovina with just Bosnia
dfCovid['Countries and territories'] = dfCovid['Countries and territories'].replace('Bosnia_and_Herzegovina', 'Bosnia')



# filling missing values
dfCovid[['Countries and territories']] = dfCovid[['Countries and territories']].fillna('')
dfCovid[cases] = dfCovid[cases].fillna(0)
```

In [5]:

```
#Group by countries
temp = dfCovid.groupby(['Countries and territories'])['Cases', 'Deaths', 'Active'].sum().reset_index()
temp.style.background_gradient(cmap='Pastel1')
temp.head()
```

Out[5]:

	Countries and territories	Cases	Deaths	Active
0	Afghanistan	24	0	24
1	Albania	70	2	68
2	Algeria	94	10	84
3	Andorra	75	0	75


Desktop/Python/Python Maps wi ×
COVID-19 - Jupyter Notebook ×
+
localhost:8888/notebooks/Desktop/Python/COVID-19/COVID-19.ipynb#
IQ
jupyter COVID-19 Last Checkpoint: 39 minutes ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

4	Argentina	158	3	155

In [6]:
```
#Group by Year and Month
temp = dfCovid.groupby(['Year','Month'])['Cases', 'Deaths', 'Active'].sum().reset_index()
temp.style.background_gradient(cmap='Pastel1')
```

Out[6]:

	Year	Month	Cases	Deaths	Active
0	2019	12	27	0	27
1	2020	1	9799	213	9586
2	2020	2	75377	2708	72669
3	2020	3	186161	8331	177830

In [7]:
```
#Group by Year
temp = dfCovid.groupby('Year')['Cases', 'Deaths', 'Active'].sum().reset_index()
temp.style.background_gradient(cmap='Pastel1')
```

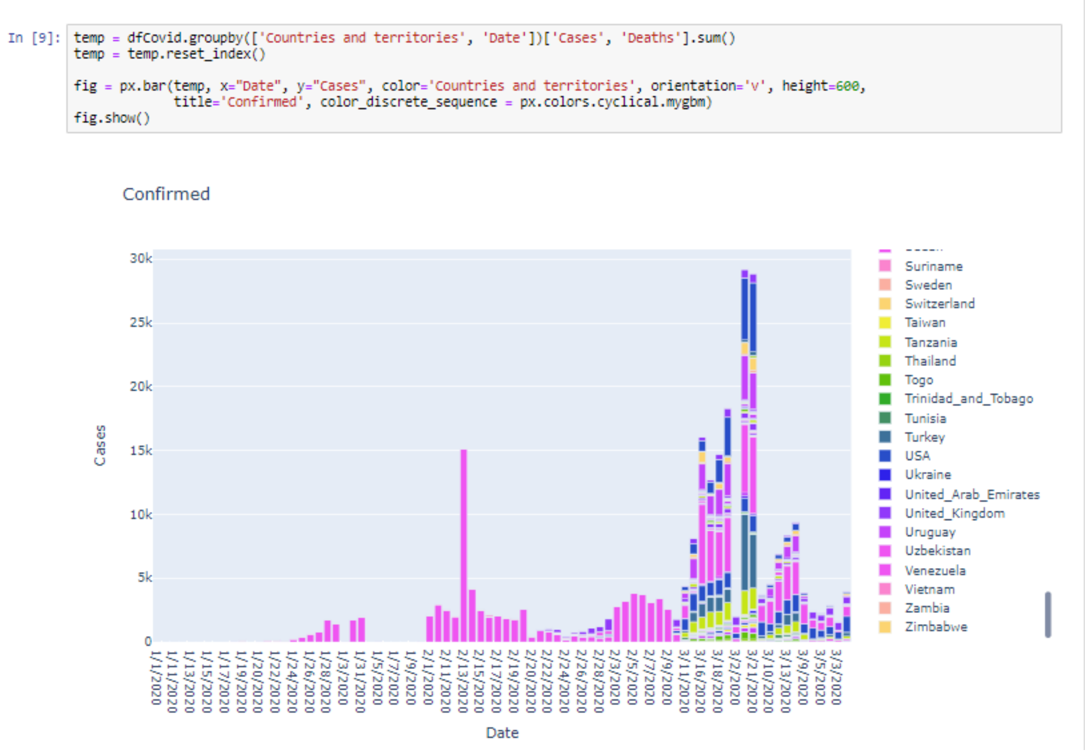
Out[7]:

	Year	Cases	Deaths	Active
0	2019	27	0	27
1	2020	271337	11252	260085

In [8]:
```
sns.distplot(temp['Cases'])
```

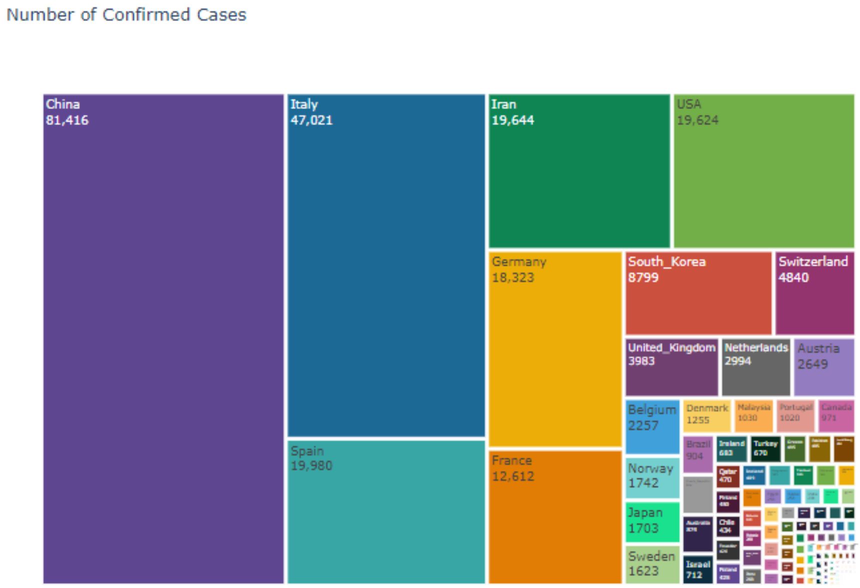
Out[8]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x203caad3048>
```

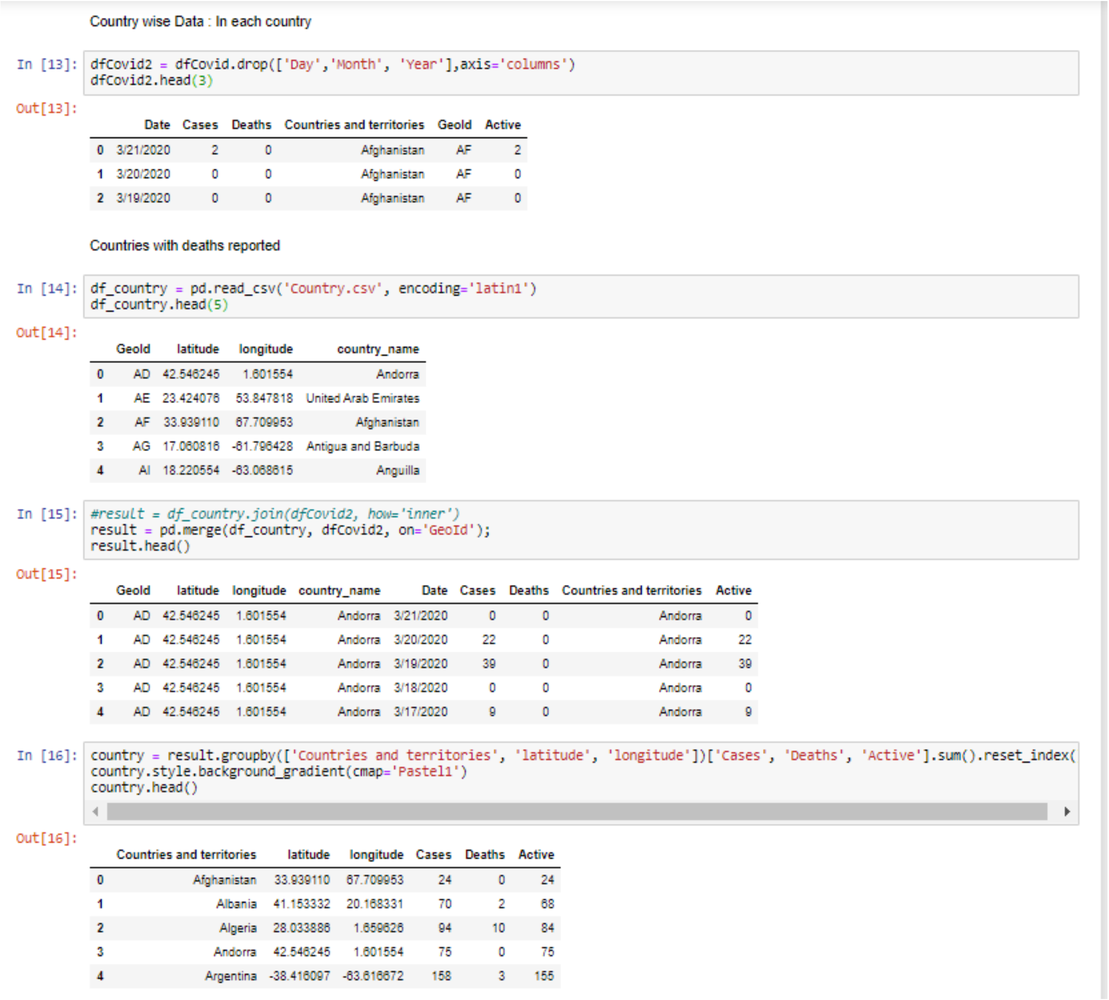
In [9]:
```
temp = dfCovid.groupby(['Countries and territories', 'Date'])['Cases', 'Deaths'].sum()
temp = temp.reset_index()

fig = px.bar(temp, x="Date", y="Cases", color='Countries and territories', orientation='v', height=600,
             title='Confirmed', color_discrete_sequence = px.colors.cyclical.mygbm)
fig.show()
```



In [9]:
```
temp = dfCovid.groupby(['Countries and territories', 'Date'])['Cases', 'Deaths'].sum()
temp = temp.reset_index()

fig = px.bar(temp, x="Date", y="Cases", color='Countries and territories', orientation='v', height=600,
             title='Confirmed', color_discrete_sequence = px.colors.cyclical.mygbm)
fig.show()
```

```
In [10]: fig = px.treemap(dfCovid.sort_values(by='Cases', ascending=False).reset_index(drop=True),
                          path=["Countries and territories"], values="Cases", height=700,
                          title='Number of Confirmed Cases',
                          color_discrete_sequence = px.colors.qualitative.Prism)
         fig.data[0].textinfo = 'label+text+value'
         fig.show()

         fig = px.treemap(dfCovid.sort_values(by='Deaths', ascending=False).reset_index(drop=True),
                          path=["Countries and territories"], values="Deaths", height=700,
                          title='Number of Deaths reported',
                          color_discrete_sequence = px.colors.qualitative.Prism)
         fig.data[0].textinfo = 'label+text+value'
         fig.show()
```





Country wise Data : In each country

```
In [13]: dfCovid2 = dfCovid.drop(['Day','Month', 'Year'],axis='columns')
         dfCovid2.head(3)
```

Out[13]:

	Date	Cases	Deaths	Countries and territories	GeoId	Active
0	3/21/2020	2	0	Afghanistan	AF	2
1	3/20/2020	0	0	Afghanistan	AF	0
2	3/19/2020	0	0	Afghanistan	AF	0

Countries with deaths reported

```
In [14]: df_country = pd.read_csv('Country.csv', encoding='latin1')
         df_country.head(5)
```

Out[14]:

	GeoId	latitude	longitude	country_name
0	AD	42.546245	1.601554	Andorra
1	AE	23.424076	53.847818	United Arab Emirates
2	AF	33.939110	67.709953	Afghanistan
3	AG	17.060816	-61.796428	Antigua and Barbuda
4	AI	18.220554	-63.068615	Anguilla

```
In [15]: #result = df_country.join(dfCovid2, how='inner')
         result = pd.merge(df_country, dfCovid2, on='GeoId');
         result.head()
```

Out[15]:

	GeoId	latitude	longitude	country_name	Date	Cases	Deaths	Countries and territories	Active
0	AD	42.546245	1.601554	Andorra	3/21/2020	0	0	Andorra	0
1	AD	42.546245	1.601554	Andorra	3/20/2020	22	0	Andorra	22
2	AD	42.546245	1.601554	Andorra	3/19/2020	39	0	Andorra	39
3	AD	42.546245	1.601554	Andorra	3/18/2020	0	0	Andorra	0
4	AD	42.546245	1.601554	Andorra	3/17/2020	9	0	Andorra	9

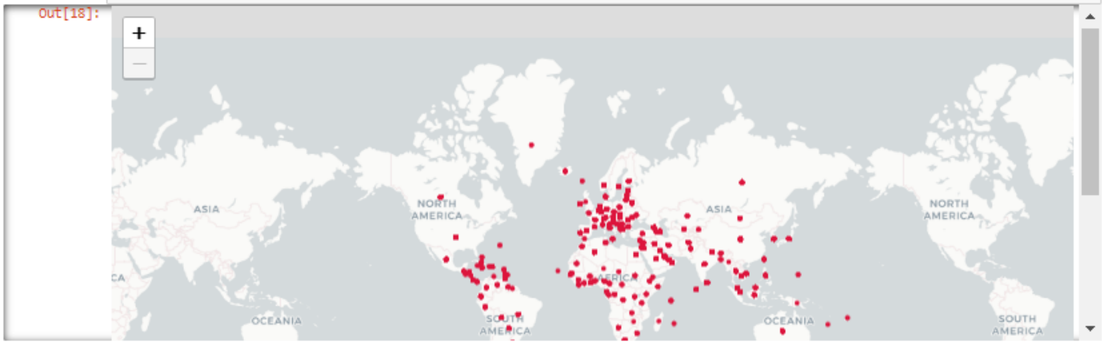
```
In [16]: country = result.groupby(['Countries and territories', 'latitude', 'longitude'])['Cases', 'Deaths', 'Active'].sum().reset_index(
         country.style.background_gradient(cmap='Pastel1')
         country.head()
```

Out[16]:

	Countries and territories	latitude	longitude	Cases	Deaths	Active
0	Afghanistan	33.939110	67.709953	24	0	24
1	Albania	41.153332	20.168331	70	2	68
2	Algeria	28.033886	1.659626	94	10	84
3	Andorra	42.546245	1.601554	75	0	75
4	Argentina	-38.416097	-63.616672	158	3	155

```
m = folium.Map(location=[0, 0], tiles='cartodbpositron',
               min_zoom=1, max_zoom=4, zoom_start=1)

for i in range(0, len(country)):
    folium.Circle(
        location=[country.iloc[i]['latitude'], country.iloc[i]['longitude']],
        color='crimson',
        tooltip =   '<li><bold>Countries and territories : '+str(country.iloc[i]['Countries and territories'])+
                    '<li><bold>Cases : '+str(country.iloc[i]['Cases'])+
                    '<li><bold>Deaths : '+str(country.iloc[i]['Deaths']),
            radius=int(country.iloc[i]['Cases'])**1.1).add_to(m)
m
```

Out[18]:



Countries with Confirmed Cases

In [19]:

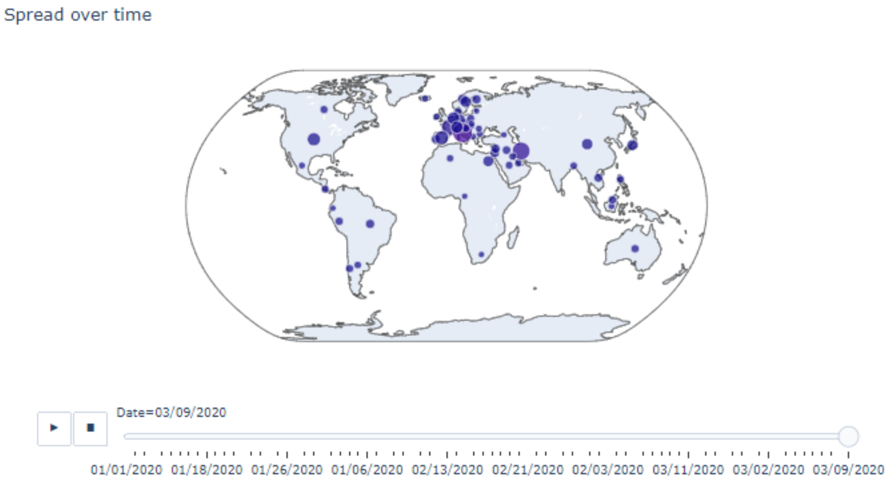
```
# Confirmed

fig = px.choropleth(country, locations="Countries and territories",
                    locationmode='country names', color="Cases",
                    hover_name="Countries and territories", range_color=[1,7000],
                    color_continuous_scale="aggrnyl",
                    title='Countries with Confirmed Cases')
fig.update(layout_coloraxis_showscale=False)
fig.show()
```

Countries with Confirmed Cases

Spread over time

In [21]:

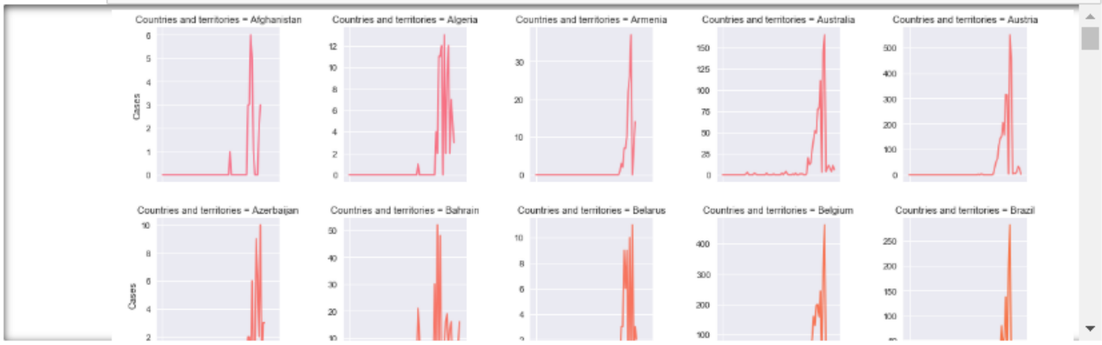
```
formated_gdf = country.groupby(['Date', 'Countries and territories'])['Cases', 'Deaths'].max()
formated_gdf = formated_gdf.reset_index()
formated_gdf['Date'] = pd.to_datetime(formated_gdf['Date'])
formated_gdf['Date'] = formated_gdf['Date'].dt.strftime('%m/%d/%Y')
formated_gdf['size'] = formated_gdf['Cases'].pow(0.3)

fig = px.scatter_geo(formated_gdf, locations="Countries and territories", locationmode='country names',
                     color="Cases", size='size', hover_name="Countries and territories",
                     range_color= [0, max(formated_gdf['Cases'])+2],
                     projection="natural earth", animation_frame="Date",
                     title='Spread over time')
fig.update(layout_coloraxis_showscale=False)
fig.show()
```



Country Wise

In [22]:

```
temp = country.groupby(['Date', 'Countries and territories'])['Cases'].sum()
temp = temp.reset_index().sort_values(by=['Date', 'Countries and territories'])

plt.style.use('seaborn')
g = sns.FacetGrid(temp, col="Countries and territories", hue="Countries and territories",
                  sharey=False, col_wrap=5)
g = g.map(plt.plot, "Date", "Cases")
g.set_xticklabels(rotation=90)
plt.show()
```


Countries and territories = Afghanistan
Countries and territories = Algeria
Countries and territories = Armenia
Countries and territories = Australia
Countries and territories = Austria
Cases
Countries and territories = Azerbaijan
Countries and territories = Bahrain
Countries and territories = Belarus
Countries and territories = Belgium
Countries and territories = Brazil
Cases