

▼ CSSE COVID-19 Dataset

เป็นข้อมูลสรุปรายงานการติดเชื้อ COVID-19 จากทั่วโลกตั้งแต่วันที่ 22 มกราคม 2020 ถึง 12 มีนาคม 2020

ประกอบด้วยไฟล์

- time_series_covid19_confirmed_global.csv
- time_series_covid19_deaths_global.csv
- time_series_covid19_recovered_global.csv

Field description

- **Province/State:** China - province name; US/Canada/Australia/ - city name, state/province name; Others - name of the event (e.g., "Diamond Princess" cruise ship); other countries - blank.
- **Country/Region:** country/region name conforming to WHO (will be updated).
- **Confirmed:** the number of confirmed cases(accumulated).
- **Deaths:** the number of deaths(accumulated).
- **Recovered:** the number of recovered cases(accumulated).

จากไฟล์จงทำการเขียนโปรแกรมเพื่อตอบคำถามต่อไปนี้

1. จงแสดงอัตราการรักษาหายของเฉพาะผู้ป่วยในสหภาพยุโรปเป็นเปอร์เซ็นต์
2. จงหา Country/Region ที่มียอดจำนวนผู้ติดเชื้อล่าสุดมากที่สุด 10 อันดับพร้อมพล็อตกราฟแสดงจำนวนผู้ติดเชื้อของแต่ละ Country/Region โดยเรียงลำดับจากมากไปน้อย (ไม่ต้องแยกเป็น Province/State)
3. จงพล็อตกราฟเส้นเพื่อแสดง trend ยอดผู้ติดเชื้อในประเทศไทยในแต่ละสัปดาห์(กำหนดให้แนบข้อมูลทุกวันอาทิตย์)
4. จงหา ยอดผู้ติดเชื้อของแต่ละวันบนเรือไดมอนด์พริ้นเซสพร้อมพล็อตกราฟ แล้วระบุวันที่มีการติดเชื้อมากที่สุด
5. จงหา Country/Region ที่ล่าสุดมีเปอร์เซ็นต์อัตราการเสียชีวิตมากที่สุด 20 อันดับ พร้อมระบุว่าจาก Country/Region ในกลุ่มดังกล่าว ส่วนใหญ่อยู่ในทวีปอะไร
6. จงหา ยอดผู้ป่วยที่กำลังรักษาตัวล่าสุด (สมมติว่ายังไม่เสียชีวิตหรือหายดี) ของแต่ละ Province/State ในประเทศจีน พร้อมพล็อตกราฟแสดงจำนวนในแต่ละ Province/State

```
1 !pip install country-converter --upgrade
```

☞

Collecting country-converter

Downloading <https://files.pythonhosted.org/packages/90/cd/0222b99afed7e4e55>

| 51kB 1.7MB/s

Requirement already satisfied, skipping upgrade: pandas>=0.17.0 in /usr/local

Requirement already satisfied, skipping upgrade: numpy>=1.13.3 in /usr/local/

Requirement already satisfied, skipping upgrade: python-dateutil>=2.6.1 in /u

Requirement already satisfied, skipping upgrade: pytz>=2017.2 in /usr/local/l

Requirement already satisfied, skipping upgrade: six>=1.5 in /usr/local/lib/p

Building wheels for collected packages: country-converter

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import country_converter as coco
5 cc = coco
```

1. จงแสดงอัตราการรักษาหายของเฉพาะผู้ป่วยในสหภาพยุโรปเป็นเปอร์เซ็นต์

```
1 #Solution
2 confirmed_case1 = pd.read_csv('time_series_covid19_confirmed_global.csv',delimi
3 recovered_case1 = pd.read_csv('time_series_covid19_recovered_global.csv',delimi
4
5 confirmed = list(confirmed_case1['Country/Region'])
6 confirmed = cc.convert(names = confirmed, to = 'continent')
7 confirmed_case1['Continent'] = confirmed
8 confirmed_Europe = confirmed_case1.loc[(confirmed_case1['Continent']=='Europe')
9
10 recovered = list(recovered_case1['Country/Region'])
11 recovered = cc.convert(names = recovered, to = 'continent')
12 recovered_case1['Continent'] = recovered
13 recovered_Europe = recovered_case1.loc[(recovered_case1['Continent']=='Europe')
14
15 print('% of Recovered ' , (recovered_Europe/confirmed_Europe)*100)
16
```

```
⌘ WARNING:root:Diamond Princess not found in regex
WARNING:root:MS Zaandam not found in regex
WARNING:root:Diamond Princess not found in regex
WARNING:root:MS Zaandam not found in regex
% of Recovered 25.100562880261357
```

2. จงหา Country/Region ที่มียอดจำนวนผู้ติดเชื้อล่าสุดมากที่สุด 10 อันดับพร้อมพล็อตกราฟแสดงจำนวนผู้ติดเชื้อของแต่ละ Country/Region โดยเรียงลำดับจากมากไปน้อย (ไม่ต้องแยกเป็น Province/State)

```
1 #Solution
```

```

2 confirmed_case2 = pd.read_csv('time_series_covid19_confirmed_global.csv',delimi
3 confirmed_case2 = confirmed_case2.groupby('Country/Region').agg('sum')
4 confirmed =list(confirmed_case2)
5 confirmed.remove('Lat')
6 confirmed.remove('Long')
7 confirmed_case2['Max'] = confirmed_case2[confirmed].max(axis=1)
8 confirmed_case2_sort_max = confirmed_case2.sort_values('Max', axis=0 , ascendin
9 confirmed_case2_sort_max.loc[:,['Max']].head(10)

```



	Max
Country/Region	
US	555313
Spain	166831
Italy	156363
France	133670
Germany	127854
United Kingdom	85206
China	83134
Iran	71686
Turkey	56956
Belgium	29647

3. จงพล็อตกราฟเส้นเพื่อแสดง trend ยอดผู้ติดเชื้อในประเทศไทยในแต่ละสัปดาห์(กำหนดให้นำข้อมูลทุกวันอาทิตย์)

```

1 #Solution
2 confirmed_case3 = pd.read_csv('time_series_covid19_confirmed_global.csv',delimi
3 d3 = confirmed_case3.loc[confirmed_case3['Country/Region']=='Thailand']
4 dat3 = pd.DataFrame(columns=['Weeks' , 'Value'])
5 dd = (d3.shape[1] - 9)//7
6 for i in range(dd):
7     dat3.at[i+1,'Weeks'] = i+1
8     dat3.at[i+1,'Value'] = d3.iloc[:,9+7*i][209]
9 dat3.plot(kind='scatter',x = 'Weeks',y = 'Value')

```



<matplotlib.axes._subplots.AxesSubplot at 0x7f046dfdbe48>

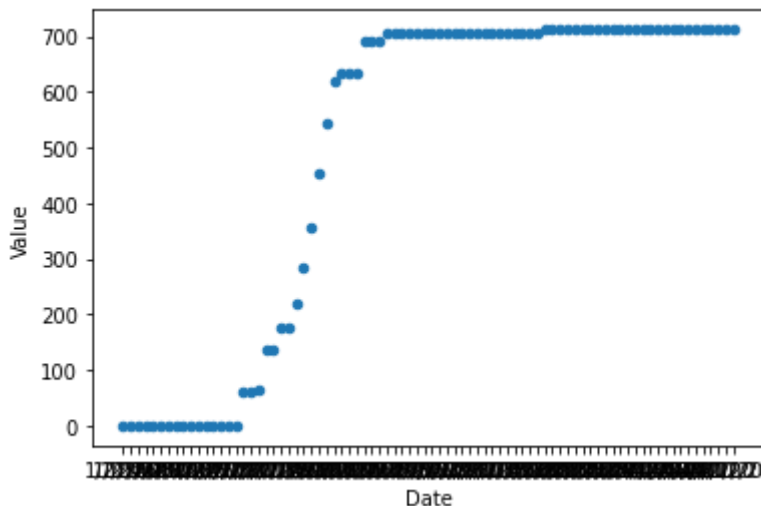


4. จงหา ยอดผู้ติดเชื้อของแต่ละวันบนเรือไดมอนด์พริ้นเซส พร้อมพล็อตกราฟ แล้วระบุวันที่มีการติดเชื้อมากที่สุด

0- | |

```
1 #Solution
2 confirmed_case4 = pd.read_csv('time_series_covid19_confirmed_global.csv',delimi
3 d4 = confirmed_case4.loc[confirmed_case4['Country/Region']=='Diamond Princess']
4 cols4 = [i for i in list(d4.columns) if i not in ['Province/State','Country/Reg
5 dat4 = d4.drop(columns=['Province/State','Country/Region','Lat','Long'])
6 data4 = pd.DataFrame(cols4, columns=['Date'])
7 l4=[]
8 for i in range(len(cols4)):
9     l4.append(d4.at[88 ,cols4[i]])
10 data4['Value'] = l4
11 data4.plot(kind='scatter' , x = 'Date' ,y = 'Value')
12 data4.max()
13
```

```
☞ Date      4/9/20
   Value      712
   dtype: object
```



5. จงหา Country/Region ที่ล่าสุดมีเปอร์เซ็นต์อัตราการเสียชีวิตมากที่สุด 20 อันดับ พร้อมระบุว่า จาก Country/Region ในกลุ่มดังกล่าว ส่วนใหญ่อยู่ในทวีปอะไร

```
1 #Solution
2 deaths case5 = pd.read_csv('time series covid19 deaths global.csv',delimiter=',',
https://colab.research.google.com/drive/1cRMc5hbzdgEyzFdgJJOG5ZMDp13YcZAD?hl=en#scrollTo=PHYgWMUn54oj&uniqifier=1 4/7
```

```
3 deaths = list(deaths_case5['Country/Region'])
4 deaths_continent = cc.convert(names = deaths, to = 'continent')
5 deaths_case5['Continent'] = deaths_continent
6 deaths_case5 = deaths_case5.groupby(['Country/Region', 'Continent']).agg('sum')
7 deaths_case5
8
9 confirmed_case5 = pd.read_csv('time_series_covid19_confirmed_global.csv', delimi
10 confirmed = list(confirmed_case5['Country/Region'])
11 confirmed_continent = cc.convert(names = confirmed, to = 'continent')
12 confirmed_case5['Continent'] = confirmed_continent
13 confirmed_case5 = confirmed_case5.groupby(['Country/Region', 'Continent']).agg('
14 confirmed_case5
15 resolve_case = (deaths_case5/confirmed_case5)*100
16 resolve_case
17 resolve_top = resolve_case.sort_values('4/12/20', axis=0, ascending=False)
18 resolve_top.loc[:, ['4/12/20']].head(20)
19
```



```
WARNING:root:Diamond Princess not found in regex
WARNING:root:MS Zaandam not found in regex
WARNING:root:Diamond Princess not found in regex
WARNING:root:MS Zaandam not found in regex
```

```
1 deaths_case5 = deaths_case5.groupby('Country/Region').agg('sum')
2 deaths.remove('Lat')
3 deaths.remove('Long')
4 deaths_case5['Max'] = deaths_case5[deaths].max(axis=1)
5 confirmed_case5 = confirmed_case2
6 deaths_case5['% Deaths/Confirmed'] = (deaths_case5['Max']/confirmed_case5["Max"]
7 deaths_case5_sort_max = deaths_case5.sort_values('% Deaths/Confirmed',axis=0,as
8 deaths_case5_sort_max.loc[:,['% Deaths/Confirmed']].head(20)
```

```
Malawi          Africa          15.384615
```

6. จงหายอดผู้ป่วยที่กำลังรักษาตัวล่าสุด (สมมติว่ายังไม่เสียชีวิตหรือ

- ▼ หายดี) ของแต่ละ Province/State ในประเทศจีน พร้อมพล็อตกราฟแสดงจำนวนในแต่ละ Province/State

```
Italy          Europe          12 726156
```

```
1 #Solution
2 confirmed_case6 = pd.read_csv('time_series_covid19_confirmed_global.csv',delimi
3 deaths_case6 = pd.read_csv('time_series_covid19_deaths_global.csv',delimiter=',
4 recovered_case6 = pd.read_csv('time_series_covid19_recovered_global.csv',delimi
5
6 confirmed_case6 = confirmed_case6.drop(columns='Lat')
7 confirmed_case6 = confirmed_case6.drop(columns='Long')
8 deaths_case6 = deaths_case6.drop(columns='Lat')
9 deaths_case6 = deaths_case6.drop(columns='Long')
10 recovered_case6 = recovered_case6.drop(columns='Lat')
11 recovered_case6 = recovered_case6.drop(columns='Long')
12
13 confirmed_case6 = confirmed_case6[confirmed_case6['Country/Region'] == 'China']
14 deaths_case6 = deaths_case6[deaths_case6['Country/Region'] == 'China']
15 recovered_case6 = recovered_case6[recovered_case6['Country/Region'] == 'China']
16
17 l6 = recovered_case6['Province/State'].values.tolist()
18 diff = pd.DataFrame(l6 ,columns=['Province/State'])
19
20 dl=[]
21 for i in range(len(l6)) :
22     c = int(confirmed_case6.loc[confirmed_case6['Province/State'] == l6[i]].max(
23     d = int(deaths_case6.loc[deaths_case6['Province/State'] == l6[i]].max(axis=1
24     r = int(recovered_case6.loc[recovered_case6['Province/State'] == l6[i]].max(
25     resolve = c-d-r
26     dl.append(resolve)
27
28 diff['Diff'] = dl
29 diff.plot(kind='bar',x = 'Province/State',y= 'Diff')
30
```



<matplotlib.axes._subplots.AxesSubplot at 0x7f046d905ef0>

