Name: Swaraj Londhe

Roll no: CS6-04

PRN: 202401100123

Topic: COVID-19 Data Analysis Using Numpy and Pandas

Dataset: https://www.kaggle.com/datasets/imdevskp/corona-virus-report

Problem Statements and Solutions:

1. Total Confirmed Cases Worldwide

```
[ ] total_confirmed = int(df['Confirmed'].sum())
    print(total_confirmed)
```

```
828508482
```

2. Total Deaths Worldwide

```
total_deaths = int(df['Deaths'].sum())
print(total_deaths)
```

```
43384903
```

3.Total Recovered Cases Worldwide

```
[ ] total_recovered = int(df['Recovered'].sum())
    print(total_recovered)
```

```
388408229
```

4. Top 10 Countries with Highest Confirmed Cases

```
[ ] top_countries_confirmed = (
        df.groupby('Country/Region')['Confirmed']
        .max()
        .sort_values(ascending=False)
        .head(10)
    )
    print(top_countries_confirmed)
```

```
Country/Region
US              4290259
Brazil          2442375
India           1480073
Russia           816680
South Africa     452529
Mexico           395489
Peru             389717
Chile            347923
United Kingdom   300111
Iran             293606
Name: Confirmed, dtype: int64
```

5.Top 10 Countries with Highest Death Counts

```
[ ] top_countries_deaths = (
        df.groupby('Country/Region')['Deaths']
        .max()
        .sort_values(ascending=False)
        .head(10)
    )
    print(top_countries_deaths)
```

```
Country/Region
US              148011
Brazil           87618
United Kingdom   45759
Mexico           44022
Italy            35112
India            33408
France           30096
Spain            28752
Peru             18418
Iran             15912
Name: Deaths, dtype: int64
```

6. Daily Global New Confirmed Cases

```
daily_confirmed = (
    df.groupby('Date')['Confirmed']
    .sum()
    .reset_index()
)
print(daily_confirmed)
```

```
          Date  Confirmed
0   2020-01-22        555
1   2020-01-23        654
2   2020-01-24        941
3   2020-01-25       1434
4   2020-01-26       2118
..         ...        ...
183 2020-07-23   15510481
184 2020-07-24   15791645
185 2020-07-25   16047190
186 2020-07-26   16251796
187 2020-07-27   16480485

[188 rows x 2 columns]
```

7. Daily Global Death Count Trend

```
daily_deaths = (
    df.groupby('Date')['Deaths']
    .sum()
    .reset_index()
)
print(daily_deaths)
```

```
          Date  Deaths
0   2020-01-22      17
1   2020-01-23      18
2   2020-01-24      26
3   2020-01-25      42
4   2020-01-26      56
..         ...     ...
183 2020-07-23  633506
184 2020-07-24  639650
185 2020-07-25  644517
186 2020-07-26  648621
187 2020-07-27  654036

[188 rows x 2 columns]
```

8. Total Active Cases per WHO Region

```
active_by_region = (
    df.groupby('WHO Region')['Active']
    .sum()
    .sort_values(ascending=False)
)
print(active_by_region)
```

```
WHO Region
Americas                225832458
Europe                  106406678
Eastern Mediterranean    24108160
South-East Asia          23629904
Africa                   10158119
Western Pacific           6580031
Name: Active, dtype: int64
```

9. Country with the First Confirmed Case

```
first_case = (
    df[df['Confirmed'] > 0]
    .sort_values('Date')
    .iloc[0][['Country/Region', 'Date']]
)
print(first_case)
```

```
Country/Region         China
Date                2020-01-22
Name: 48, dtype: object
```

10. Country with the Highest Recovery Rate

```
recovery_rate = (
    df.groupby('Country/Region')
    .agg({'Recovered': 'max', 'Confirmed': 'max'})
)
recovery_rate['Recovery Rate (%)'] = (recovery_rate['Recovered'] / recovery_rate['Confirmed']) * 100
highest_recovery_country = recovery_rate.sort_values('Recovery Rate (%)', ascending=False).head(1)
print(highest_recovery_country)
```

```
                Recovered  Confirmed  Recovery Rate (%)
Country/Region
Dominica               18         18              100.0
```

11. Country with the Highest Death Rate

```
death_rate = (
    df.groupby('Country/Region')
    .agg({'Deaths': 'max', 'Confirmed': 'max'})
)
death_rate['Death Rate (%)'] = (death_rate['Deaths'] / death_rate['Confirmed']) * 100
highest_death_country = death_rate.sort_values('Death Rate (%)', ascending=False).head(1)
print(highest_death_country)
```

```
                Deaths  Confirmed  Death Rate (%)
Country/Region
Yemen              483       1691        28.56298
```

12. Growth of Confirmed Cases in India Over Tim

```
india_growth = (
    df[df['Country/Region'] == 'India']
    .groupby('Date')['Confirmed']
    .sum()
    .reset_index()
)
print(india_growth)
```

```
           Date  Confirmed
0    2020-01-22          0
1    2020-01-23          0
2    2020-01-24          0
3    2020-01-25          0
4    2020-01-26          0
..          ...        ...
183  2020-07-23    1288108
184  2020-07-24    1337024
185  2020-07-25    1385635
186  2020-07-26    1435616
187  2020-07-27    1480073

[188 rows x 2 columns]
```

13. Growth of Deaths in USA Over Time

```
usa_deaths = (
    df[df['Country/Region'] == 'US']
    .groupby('Date')['Deaths']
    .sum()
    .reset_index()
)
print(usa_deaths)
```

```
           Date  Deaths
0    2020-01-22       0
1    2020-01-23       0
2    2020-01-24       0
3    2020-01-25       0
4    2020-01-26       0
..          ...     ...
183  2020-07-23  144430
184  2020-07-24  145560
185  2020-07-25  146465
186  2020-07-26  146935
187  2020-07-27  148011

[188 rows x 2 columns]
```

14. WHO Region with the Highest Total Confirmed Cases

```
region_confirmed = (
    df.groupby('WHO Region')['Confirmed']
    .sum()
    .sort_values(ascending=False)
)
top_region = region_confirmed.head(1)
print(top_region)
```

```
WHO Region
Americas    402261194
Name: Confirmed, dtype: int64
```

15. Average Number of New Cases Per Day Globally

+ Code + Text

```
avg_daily_confirmed = (
    df.groupby('Date')['Confirmed']
    .sum()
    .mean()
)
print(int(avg_daily_confirmed))
```

```
440960
```

16. Daily New Cases in a Specific Country (e.g., Italy)

```
italy_daily = (
    df[df['Country/Region'] == 'Italy']
    .groupby('Date')['Confirmed']
    .sum()
    .reset_index()
)
print(italy_daily)
```

```
          Date  Confirmed
0   2020-01-22          0
1   2020-01-23          0
2   2020-01-24          0
3   2020-01-25          0
4   2020-01-26          0
..         ...        ...
183 2020-07-23     245338
184 2020-07-24     245590
185 2020-07-25     245864
186 2020-07-26     246118
187 2020-07-27     246286

[188 rows x 2 columns]
```

17. Countries with Zero Deaths Despite Confirmed Cases

```
zero_death_countries = (
    df.groupby('Country/Region')
    .agg({'Confirmed': 'max', 'Deaths': 'max'})
)
zero_death_countries = zero_death_countries[(zero_death_countries['Confirmed'] > 0) & (zero_death_countries['Deaths'] == 0)]
print(zero_death_countries.index.tolist())
```

```
['Bhutan', 'Cambodia', 'Dominica', 'Eritrea', 'Fiji', 'Greenland', 'Grenada', 'Holy See', 'Laos', 'Mongolia', 'Saint Kitts and Nevis', 'Saint Lucia', 'Saint Vincent and the Grenadine
```

18. Comparison of Case Trends Between Two Countries (USA vs India)

```
comparison = (
    df[df['Country/Region'].isin(['US', 'India'])]
    .groupby(['Date', 'Country/Region'])['Confirmed']
    .sum()
    .unstack()
    .fillna(0)
    .reset_index()
)
print(comparison)
```

```
Country/Region        Date    India       US
0                 2020-01-22        0        1
1                 2020-01-23        0        1
2                 2020-01-24        0        2
3                 2020-01-25        0        2
4                 2020-01-26        0        5
..                       ...      ...      ...
183               2020-07-23  1288108  4038816
184               2020-07-24  1337024  4112531
185               2020-07-25  1385635  4178970
186               2020-07-26  1435616  4233923
187               2020-07-27  1480073  4290259

[188 rows x 3 columns]
```

19. Find the Date When Global Active Cases Were Highest

```
peak_active_day = (
    df.groupby('Date')['Active']
    .sum()
    .sort_values(ascending=False)
    .head(1)
)
print(peak_active_day)
```

```
Date
2020-07-27    6358362
Name: Active, dtype: int64
```

20. Top 5 Countries with the Most Active Cases at Their Peak

```
peak_active_countries = (
    df.groupby('Country/Region')['Active']
    .max()
    .sort_values(ascending=False)
    .head(5)
)
print(peak_active_countries)
```

```
Country/Region
US                2816444
Brazil             583000
India              495499
United Kingdom     254352
Russia             245382
Name: Active, dtype: int64
```