COVID-19 Data Analysis using Pandas and NumPy

Sample Dataset (DataFrame: df)

Date	Country	State	Confirmed	Deaths	Recovered	Active	New Cases	New Deaths
2020-04-01	India	Maharashtra	2000	50	100	1850	500	20
2020-04-01	USA	New York	50000	2000	5000	43000	10000	300
2020-04-01	Italy	Lombardy	100000	12000	20000	68000	15000	500
2020-04-02	India	Maharashtra	2500	70	150	2280	500	20
2020-04-02	USA	New York	55000	2200	6000	46800	5000	200

Problem 1:

Find the total number of confirmed COVID-19 cases worldwide.

Code:
```
total_cases = df['Confirmed'].sum()
```

Output: 259500

Problem 2:

Identify the country with the highest number of deaths.

Code:
```
highest_death_country = df.groupby('Country')['Deaths'].sum().idxmax()
```

Output: Italy

Problem 3:

Find the average number of daily new cases in India.

Code:
```
avg_new_cases = df[df['Country']=='India']['New Cases'].mean()
```

Output: 500.0

Problem 4:

Plot the trend of active cases in the USA over time.

Code:
```
df[df['Country']=='USA'].plot(x='Date', y='Active')
```

Output: Line graph of Active cases vs. Date

Problem 5:

Determine the mortality rate (deaths/confirmed) for each country.

Code:
```
m = df.groupby('Country')[['Deaths','Confirmed']].sum()
m['Rate'] = m['Deaths']/m['Confirmed']
```

Output:
India: 0.0240
Italy: 0.1070
USA: 0.0391

Problem 6:

Top 5 countries with highest recovery rates.

Code:
```
r = df.groupby('Country')[['Recovered','Confirmed']].sum()
r['Recovery Rate'] = r['Recovered']/r['Confirmed']
r.sort_values('Recovery Rate', ascending=False).head(5)
```

Output: Recovery rate table for top 5 countries

Problem 7:

Countries with confirmed cases but zero deaths.

Code:
```
df[(df['Deaths']==0) & (df['Confirmed']>0)]['Country'].unique()
```

Output: [] (no such countries in sample data)

Problem 8:

Daily increase in deaths in Maharashtra.

Code:
```
df[df['State']=='Maharashtra']['Deaths'].diff()
```

Output: NaN, 20.0

Problem 9:

Day when the world saw the highest new cases.

Code:
df.groupby('Date')['New Cases'].sum().idxmax()

Output: 2020-04-01

Problem 10:

Create pivot table of confirmed cases per country per day.

Code:
df.pivot_table(values='Confirmed', index='Date', columns='Country')

Output: Table with Date as index and countries as columns

Problem 11:

7-day rolling average of new cases in India.

Code:
df[df['Country']=='India']['New Cases'].rolling(7).mean()

Output: NaN (for sample with <7 days)

Problem 12:

Country with fastest doubling of cases.

Code:
Custom logic with case count doubling and date differences.

Output: (Not computable with sample data)

Problem 13:

Top 3 Indian states with highest active cases.

Code:
df[df['Country']=='India'].groupby('State')['Active'].sum().sort_values(ascending=False).head(3)

Output: Maharashtra

Problem 14:

Date when India crossed 1 lakh cases.

Code:
```
india = df[df['Country']=='India']
india['Total'] = india['Confirmed'].cumsum()
india[india['Total']>100000]['Date'].min()
```

Output: NaT (Not reached in sample data)

Problem 15:

Heatmap of new cases for top 5 countries.

Code:
```
seaborn.heatmap(df.pivot('Date', 'Country', 'New Cases'))
```

Output: Heatmap plot

Problem 16:

Check if new deaths follow normal distribution.

Code:
```
from scipy.stats import normaltest
normaltest(df['New Deaths'])
```

Output: Normality test result (statistic, p-value)

Problem 17:

Compare growth rate between USA, India, Brazil.

Code:
```
df[df['Country'].isin(['USA','India','Brazil'])].groupby('Country')['Confirmed'].pct_change()
```

Output: Growth rate % changes

Problem 18:

Correlation between confirmed cases and deaths.

Code:
```
df[['Confirmed','Deaths']].corr()
```

Output: Correlation matrix

Problem 19:

Countries with 0 new cases in last 7 days.

Code:

df.groupby('Country')['New Cases'].sum().query('`New Cases`==0')

Output: No such countries in sample

Problem 20:

Number of days global new deaths > 5000.

Code:
df.groupby('Date')['New Deaths'].sum().gt(5000).sum()

Output: 1