

```
import pandas as pd
df = pd.read_csv('owid-covid-data.csv')
df.head()
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
0	AFG	Asia	Afghanistan	2020-01-05	0	0	NaN	0	0	NaN
1	AFG	Asia	Afghanistan	2020-01-06	0	0	NaN	0	0	NaN
2	AFG	Asia	Afghanistan	2020-01-07	0	0	NaN	0	0	NaN
3	AFG	Asia	Afghanistan	2020-01-08	0	0	NaN	0	0	NaN
4	AFG	Asia	Afghanistan	2020-01-09	0	0	NaN	0	0	NaN

5 rows × 67 columns

```
# Keep only useful columns
columns_needed = ['location', 'date', 'total_cases', 'new_cases', 'total_deaths', 'new_deaths', 'population']
df = df[columns_needed]

# Remove rows with nulls in key columns
df.dropna(subset=['location', 'date', 'total_cases'], inplace=True)

# Remove aggregate rows (e.g., World, Asia)
regions = ['World', 'Asia', 'Africa', 'Europe', 'North America', 'South America',
           'European Union', 'International', 'High income', 'Low income', 'Upper middle income', 'Lower middle income']
df = df[~df['location'].isin(regions)]

# Convert 'date' to datetime format
df['date'] = pd.to_datetime(df['date'])

# Check the cleaned data
pd.DataFrame(df)
```

	location	date	total_cases	new_cases	total_deaths	new_deaths	population
0	Afghanistan	2020-01-05	0	0	0	0	41128772.0
1	Afghanistan	2020-01-06	0	0	0	0	41128772.0
2	Afghanistan	2020-01-07	0	0	0	0	41128772.0
3	Afghanistan	2020-01-08	0	0	0	0	41128772.0
4	Afghanistan	2020-01-09	0	0	0	0	41128772.0
...
14513	Antigua and Barbuda	2023-01-30	9106	0	146	0	93772.0
14514	Antigua and Barbuda	2023-01-31	9106	0	146	0	93772.0
14515	Antigua and Barbuda	2023-02-01	9106	0	146	0	93772.0
14516	Antigua and Barbuda	2023-02-02	9106	0	146	0	93772.0
14517	Antigua and Barbuda	2023-02-03	9106	0	146	0	NaN

12844 rows × 7 columns

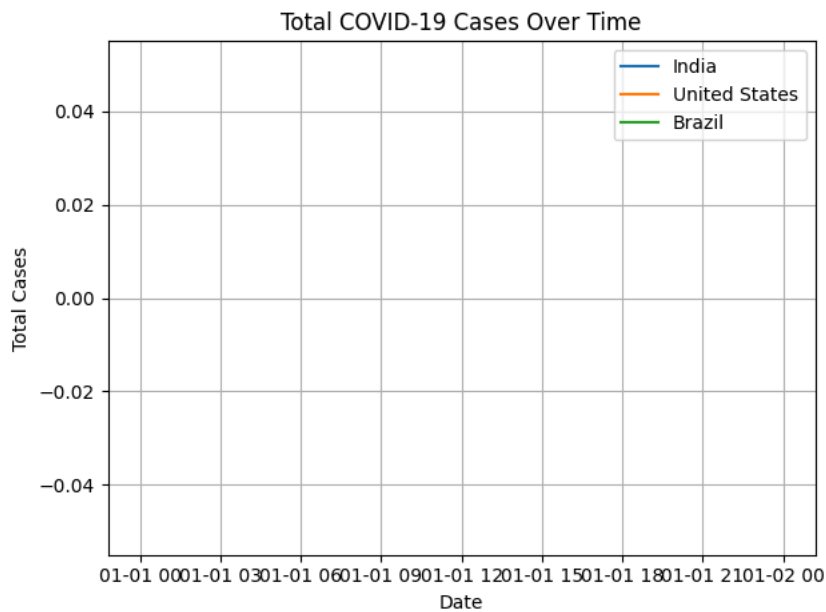
```
import matplotlib.pyplot as plt

# Filter a few countries for comparison
countries = ['India', 'United States', 'Brazil']
df_countries = df[df['location'].isin(countries)]

# Plot
plt.figure(figsize=(12,6))
for country in countries:
    country_data = df_countries[df_countries['location'] == country]
    plt.plot(country_data['date'], country_data['total_cases'], label=country)

plt.title('Total COVID-19 Cases Over Time')
plt.xlabel('Date')
plt.ylabel('Total Cases')
```

```
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

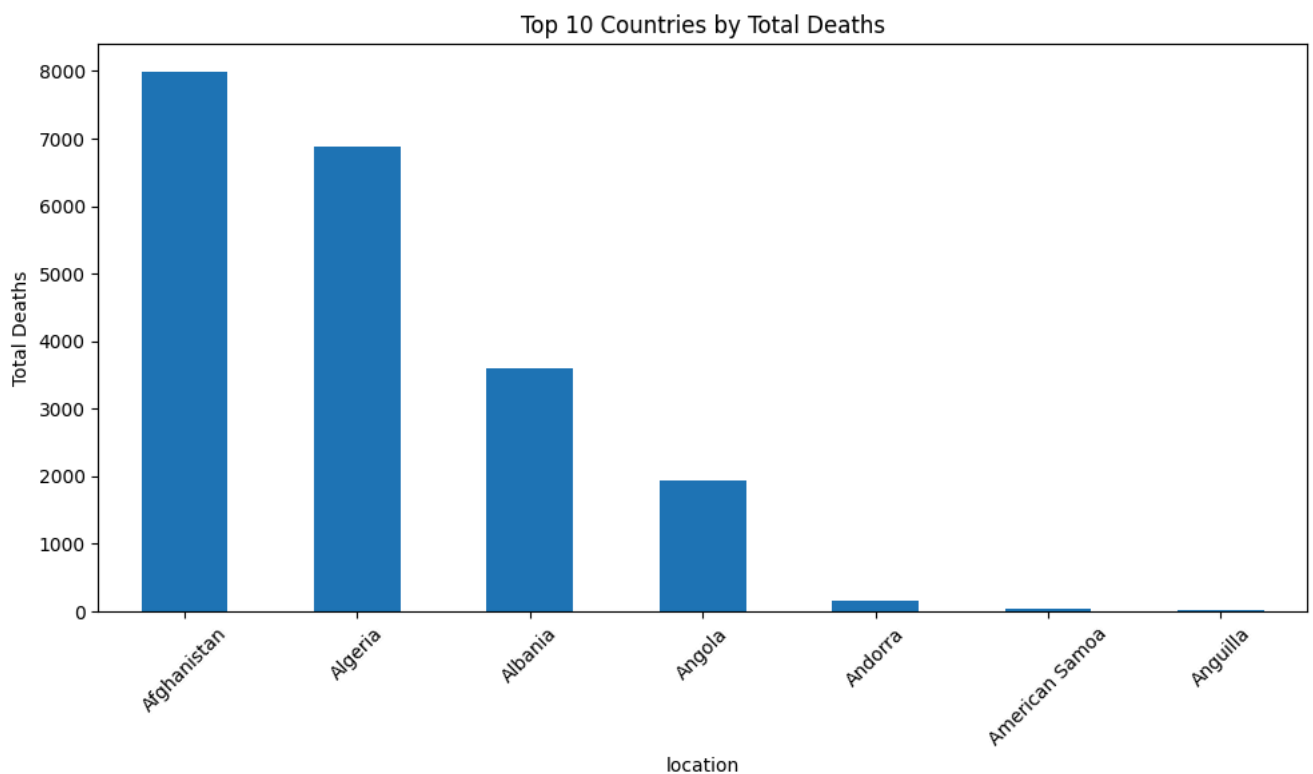


```
# Get the latest date from the dataset
latest_date = df['date'].max()

# Filter data for the latest date
latest_data = df[df['date'] == latest_date]

# Top 10 countries by total deaths
top_deaths = latest_data.groupby('location')['total_deaths'].sum().sort_values(ascending=False).head(10)

# Plot
top_deaths.plot(kind='bar', figsize=(10,6), title='Top 10 Countries by Total Deaths')
plt.ylabel('Total Deaths')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



```
import matplotlib.pyplot as plt
```

```
import matplotlib.pyplot as plt
import pandas as pd
```

```
# Step 1: Remove null or 0 total_cases
df_valid = df[(df['total_cases'].notna()) & (df['total_cases'] > 0)]

# Step 2: Pick top 3 countries with highest total cases (latest date)
latest_date = df_valid['date'].max()
top3 = df_valid[df_valid['date'] == latest_date].sort_values(by='total_cases', ascending=False).head(3)['location'].tolist()

print("Top 3 countries by total cases:", top3) # Check selected countries

# Step 3: Filter only top 3 countries
df_plot = df_valid[df_valid['location'].isin(top3)]

# Step 4: Plot
plt.figure(figsize=(12,6))
for country in top3:
    data = df_plot[df_plot['location'] == country]
    plt.plot(data['date'], data['total_cases'], label=country)

plt.title('Top 3 Countries - Total COVID-19 Cases Over Time')
plt.xlabel('Date')
plt.ylabel('Total Cases')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Top 3 countries by total cases: ['Albania', 'Algeria', 'Afghanistan']

