COVID-19 Global Data Tracker Project Implementation *Project Overview* This Jupyter Notebook implements a comprehensive COVID-19 data analysis pipeline following the project guidelines. We will track global COVID-19 trends, analyze cases, deaths, recoveries, and vaccinations across countries and time.

In [ ]:
```
# Importing necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from datetime import datetime
```

In [4]:
```
df=pd.read_csv('owid-covid-data.csv')
print("Dataset loaded successfully")
```

```
Dataset loaded successfully
```

In [34]:
```
# Display the first few rows of dataset
print(f"Dataset shape: {df.shape}")
print("\nFirst 5 rows:")
display(df.head())
```

```
Dataset shape: (350085, 67)

First 5 rows:
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	tota
0	AFG	Asia	Afghanistan	2020-01-03	NaN	0.0	NaN	
1	AFG	Asia	Afghanistan	2020-01-04	NaN	0.0	NaN	
2	AFG	Asia	Afghanistan	2020-01-05	NaN	0.0	NaN	
3	AFG	Asia	Afghanistan	2020-01-06	NaN	0.0	NaN	
4	AFG	Asia	Afghanistan	2020-01-07	NaN	0.0	NaN	

5 rows × 67 columns

In [11]:
```
# Check the number of feature columns
print("\nColumns in the dataset:")
print(len(df.columns))
```

```
Columns in the dataset:
67
```

In [13]:
```
print("\nMissing values summary:")
print(df.isnull().sum())
```

```
Missing values summary:
iso_code                                      0
continent                                 16665
location                                      0
date                                          0
total_cases                               37997
                                            ...
population                                    0
excess_mortality_cumulative_absolute     337901
excess_mortality_cumulative              337901
excess_mortality                         337901
excess_mortality_cumulative_per_million  337901
Length: 67, dtype: int64
```

In [9]:
```
# Convert date column to datetime
df['date'] = pd.to_datetime(df['date'])
```

In [14]:
```
df.describe()
```

Out[14]:

	date	total_cases	new_cases	new_cases_smoothed	total_deaths
count	350085	3.120880e+05	3.404570e+05	3.391980e+05	2.905010e+05
mean	2021-11-25 18:35:40.040275712	6.683354e+06	9.601634e+03	9.637066e+03	8.602180e+04
min	2020-01-01 00:00:00	1.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00
25%	2020-12-16 00:00:00	8.090000e+03	0.000000e+00	2.860000e-01	1.270000e+02
50%	2021-11-26 00:00:00	7.020500e+04	2.000000e+00	2.485700e+01	1.328000e+03
75%	2022-11-06 00:00:00	7.409558e+05	2.640000e+02	4.978570e+02	1.192200e+04
max	2023-10-24 00:00:00	7.714071e+08	8.401961e+06	6.402036e+06	6.972139e+06
std	NaN	4.068903e+07	1.102769e+05	9.447784e+04	4.398873e+05

8 rows × 63 columns

In [17]:
```
# Select countries of interest
countries_of_interest = ['Kenya', 'United States', 'India', 'Brazil', 'United Kingd
                         'South Africa', 'Germany', 'China', 'Japan', 'Australia']

# Filter data for selected countries
```

```
df_filtered = df[df['location'].isin(countries_of_interest)].copy()
print(df_filtered)
```

```
       iso_code      continent       location       date  total_cases  \
18020       AUS        Oceania      Australia 2020-01-03          NaN
18021       AUS        Oceania      Australia 2020-01-04          NaN
18022       AUS        Oceania      Australia 2020-01-05          NaN
18023       AUS        Oceania      Australia 2020-01-06          NaN
18024       AUS        Oceania      Australia 2020-01-07          NaN
...         ...            ...            ...        ...          ...
330861      USA  North America  United States 2023-10-14  103436829.0
330862      USA  North America  United States 2023-10-15  103436829.0
330863      USA  North America  United States 2023-10-16  103436829.0
330864      USA  North America  United States 2023-10-17  103436829.0
330865      USA  North America  United States 2023-10-18  103436829.0

        new_cases  new_cases_smoothed  total_deaths  new_deaths  \
18020         0.0                 NaN           NaN         0.0
18021         0.0                 NaN           NaN         0.0
18022         0.0                 NaN           NaN         0.0
18023         0.0                 NaN           NaN         0.0
18024         0.0                 NaN           NaN         0.0
...           ...                 ...           ...         ...
330861        NaN                 NaN     1136920.0         NaN
330862        NaN                 NaN     1136920.0         NaN
330863        NaN                 NaN     1136920.0         NaN
330864        NaN                 NaN     1136920.0         NaN
330865        NaN                 NaN     1136920.0         NaN

        new_deaths_smoothed  ...  male_smokers  handwashing_facilities  \
18020                   NaN  ...          16.5                     NaN
18021                   NaN  ...          16.5                     NaN
18022                   NaN  ...          16.5                     NaN
18023                   NaN  ...          16.5                     NaN
18024                   NaN  ...          16.5                     NaN
...                     ...  ...           ...                     ...
330861                  NaN  ...          24.6                     NaN
330862                  NaN  ...          24.6                     NaN
330863                  NaN  ...          24.6                     NaN
330864                  NaN  ...          24.6                     NaN
330865                  NaN  ...          24.6                     NaN

        hospital_beds_per_thousand  life_expectancy  human_development_index  \
18020                         3.84            83.44                    0.944
18021                         3.84            83.44                    0.944
18022                         3.84            83.44                    0.944
18023                         3.84            83.44                    0.944
18024                         3.84            83.44                    0.944
...                            ...              ...                      ...
330861                        2.77            78.86                    0.926
330862                        2.77            78.86                    0.926
330863                        2.77            78.86                    0.926
330864                        2.77            78.86                    0.926
330865                        2.77            78.86                    0.926

         population  excess_mortality_cumulative_absolute  \
18020    26177410.0                                   NaN
18021    26177410.0                                   NaN
18022    26177410.0                                 -42.7
```

```
18023    26177410.0                                    NaN
18024    26177410.0                                    NaN
...             ...                                    ...
330861  338289856.0                                    NaN
330862  338289856.0                                    NaN
330863  338289856.0                                    NaN
330864  338289856.0                                    NaN
330865  338289856.0                                    NaN

        excess_mortality_cumulative  excess_mortality  \
18020                           NaN               NaN
18021                           NaN               NaN
18022                         -1.44             -1.44
18023                           NaN               NaN
18024                           NaN               NaN
...                             ...               ...
330861                          NaN               NaN
330862                          NaN               NaN
330863                          NaN               NaN
330864                          NaN               NaN
330865                          NaN               NaN

        excess_mortality_cumulative_per_million
18020                                       NaN
18021                                       NaN
18022                                 -1.663417
18023                                       NaN
18024                                       NaN
...                                         ...
330861                                      NaN
330862                                      NaN
330863                                      NaN
330864                                      NaN
330865                                      NaN

[13856 rows x 67 columns]
```

In [18]:
```
# Check for missing values in key columns
key_columns = ['total_cases', 'new_cases', 'total_deaths', 'new_deaths',
               'total_vaccinations', 'people_vaccinated', 'people_fully_vaccinated'
print("\nMissing values in key columns:")
print(df_filtered[key_columns].isnull().sum())
```

```
Missing values in key columns:
total_cases                303
new_cases                  171
total_deaths               518
new_deaths                 161
total_vaccinations        7010
people_vaccinated         7854
people_fully_vaccinated   7878
dtype: int64
```

In [20]:
```
# Fill missing values with 0 for numerical columns (assuming no data means no cases
for col in key_columns:
```

```
    df_filtered[col] = df_filtered[col].fillna(0)
print(df_filtered[col])
```

```
18020     0.0
18021     0.0
18022     0.0
18023     0.0
18024     0.0
         ...
330861    0.0
330862    0.0
330863    0.0
330864    0.0
330865    0.0
Name: people_fully_vaccinated, Length: 13856, dtype: float64
```

In [24]:
```
df_filtered['death_rate'] = df_filtered['total_deaths'] / df_filtered['total_cases'
df_filtered['death_rate'] = df_filtered['death_rate'].replace([np.inf, -np.inf], np

print("\n Death Rate:")
print(df_filtered['death_rate'])
```

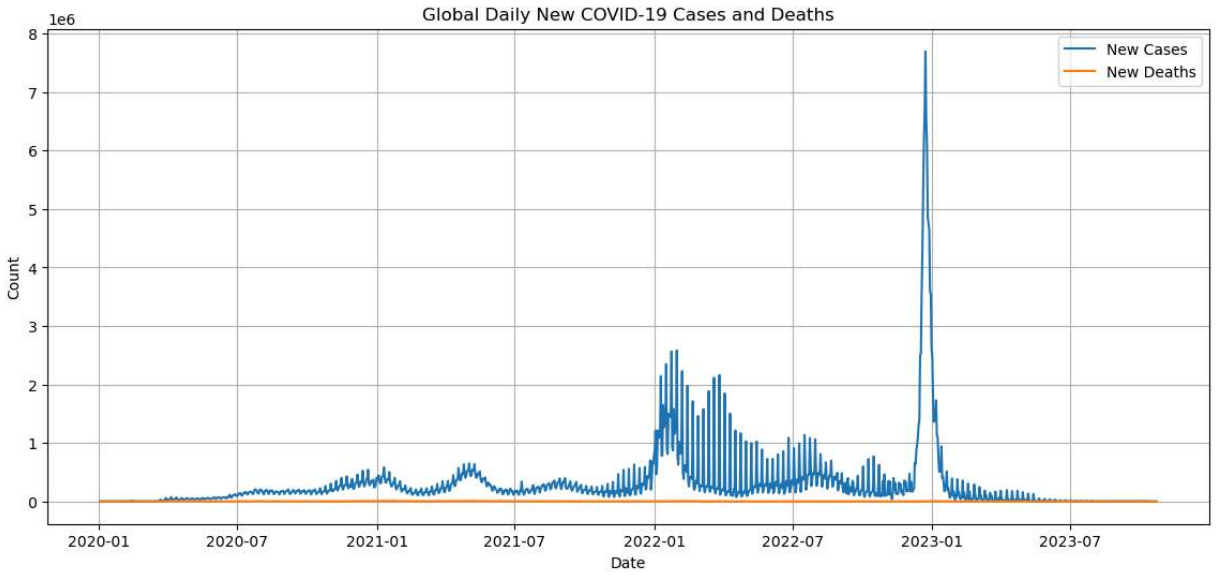
```
 Death Rate:
18020          NaN
18021          NaN
18022          NaN
18023          NaN
18024          NaN
            ...
330861    0.010991
330862    0.010991
330863    0.010991
330864    0.010991
330865    0.010991
Name: death_rate, Length: 13856, dtype: float64
```

In [26]:
```
# Filter out dates before 2020 (if any)
df_filtered = df_filtered[df_filtered['date'] >= '2020-01-01']
print(df_filtered)
```

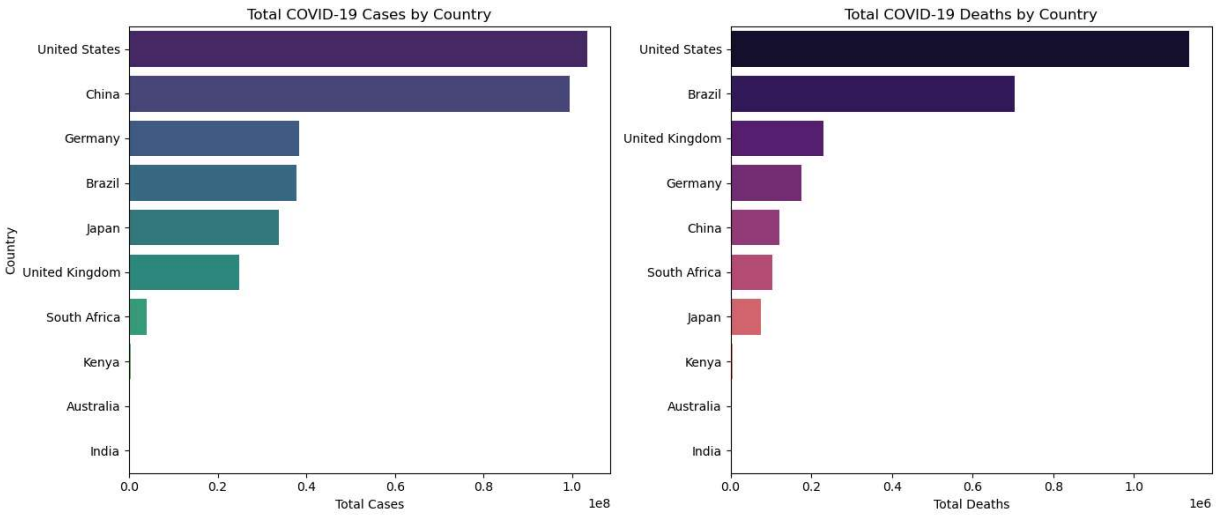
```
       iso_code      continent       location       date  total_cases  \
18020       AUS        Oceania      Australia 2020-01-03          0.0
18021       AUS        Oceania      Australia 2020-01-04          0.0
18022       AUS        Oceania      Australia 2020-01-05          0.0
18023       AUS        Oceania      Australia 2020-01-06          0.0
18024       AUS        Oceania      Australia 2020-01-07          0.0
...         ...            ...            ...        ...          ...
330861      USA  North America  United States 2023-10-14  103436829.0
330862      USA  North America  United States 2023-10-15  103436829.0
330863      USA  North America  United States 2023-10-16  103436829.0
330864      USA  North America  United States 2023-10-17  103436829.0
330865      USA  North America  United States 2023-10-18  103436829.0

        new_cases  new_cases_smoothed  total_deaths  new_deaths  \
18020         0.0                 NaN           0.0         0.0
18021         0.0                 NaN           0.0         0.0
18022         0.0                 NaN           0.0         0.0
18023         0.0                 NaN           0.0         0.0
18024         0.0                 NaN           0.0         0.0
...           ...                 ...           ...         ...
330861        0.0                 NaN     1136920.0         0.0
330862        0.0                 NaN     1136920.0         0.0
330863        0.0                 NaN     1136920.0         0.0
330864        0.0                 NaN     1136920.0         0.0
330865        0.0                 NaN     1136920.0         0.0

        new_deaths_smoothed  ...  handwashing_facilities  \
18020                   NaN  ...                     NaN
18021                   NaN  ...                     NaN
18022                   NaN  ...                     NaN
18023                   NaN  ...                     NaN
18024                   NaN  ...                     NaN
...                     ...  ...                     ...
330861                  NaN  ...                     NaN
330862                  NaN  ...                     NaN
330863                  NaN  ...                     NaN
330864                  NaN  ...                     NaN
330865                  NaN  ...                     NaN

        hospital_beds_per_thousand  life_expectancy  human_development_index  \
18020                         3.84            83.44                    0.944
18021                         3.84            83.44                    0.944
18022                         3.84            83.44                    0.944
18023                         3.84            83.44                    0.944
18024                         3.84            83.44                    0.944
...                            ...              ...                      ...
330861                        2.77            78.86                    0.926
330862                        2.77            78.86                    0.926
330863                        2.77            78.86                    0.926
330864                        2.77            78.86                    0.926
330865                        2.77            78.86                    0.926

         population  excess_mortality_cumulative_absolute  \
18020    26177410.0                                   NaN
18021    26177410.0                                   NaN
18022    26177410.0                                 -42.7
```

```
18023    26177410.0                                    NaN
18024    26177410.0                                    NaN
...             ...                                    ...
330861  338289856.0                                    NaN
330862  338289856.0                                    NaN
330863  338289856.0                                    NaN
330864  338289856.0                                    NaN
330865  338289856.0                                    NaN

        excess_mortality_cumulative  excess_mortality  \
18020                           NaN               NaN
18021                           NaN               NaN
18022                         -1.44             -1.44
18023                           NaN               NaN
18024                           NaN               NaN
...                             ...               ...
330861                          NaN               NaN
330862                          NaN               NaN
330863                          NaN               NaN
330864                          NaN               NaN
330865                          NaN               NaN

        excess_mortality_cumulative_per_million  death_rate
18020                                       NaN         NaN
18021                                       NaN         NaN
18022                                 -1.663417         NaN
18023                                       NaN         NaN
18024                                       NaN         NaN
...                                         ...         ...
330861                                      NaN    0.010991
330862                                      NaN    0.010991
330863                                      NaN    0.010991
330864                                      NaN    0.010991
330865                                      NaN    0.010991

[13856 rows x 68 columns]
```

In [27]:
```
# Aggregate global data (sum across all countries)
global_df = df_filtered.groupby('date')[['new_cases', 'new_deaths']].sum().reset_in

plt.figure(figsize=(14, 6))
plt.plot(global_df['date'], global_df['new_cases'], label='New Cases')
plt.plot(global_df['date'], global_df['new_deaths'], label='New Deaths')
plt.title('Global Daily New COVID-19 Cases and Deaths')
plt.xlabel('Date')
plt.ylabel('Count')
plt.legend()
plt.grid(True)
plt.show()
```

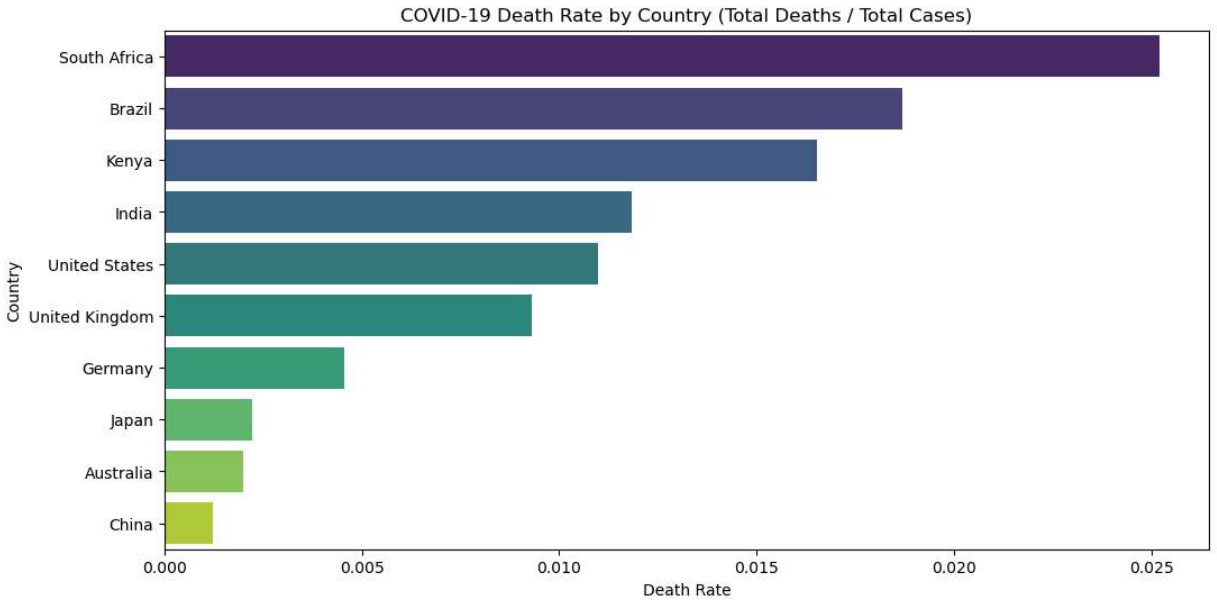


In [28]:
```
# Get latest data for each country
latest_data = df_filtered.sort_values('date').groupby('location').last().reset_inde

plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
sns.barplot(data=latest_data.sort_values('total_cases', ascending=False),
            x='total_cases', y='location', palette='viridis')
plt.title('Total COVID-19 Cases by Country')
plt.xlabel('Total Cases')
plt.ylabel('Country')

plt.subplot(1, 2, 2)
sns.barplot(data=latest_data.sort_values('total_deaths', ascending=False),
            x='total_deaths', y='location', palette='magma')
plt.title('Total COVID-19 Deaths by Country')
plt.xlabel('Total Deaths')
plt.ylabel('')
plt.tight_layout()
plt.show()
```

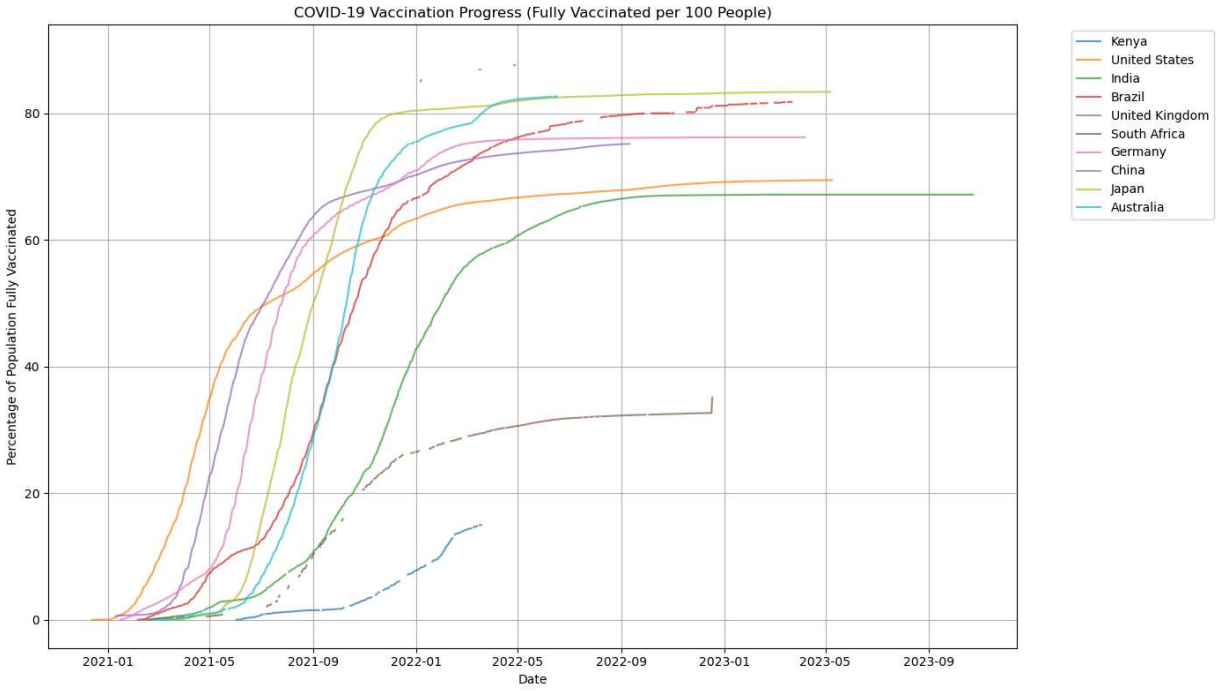


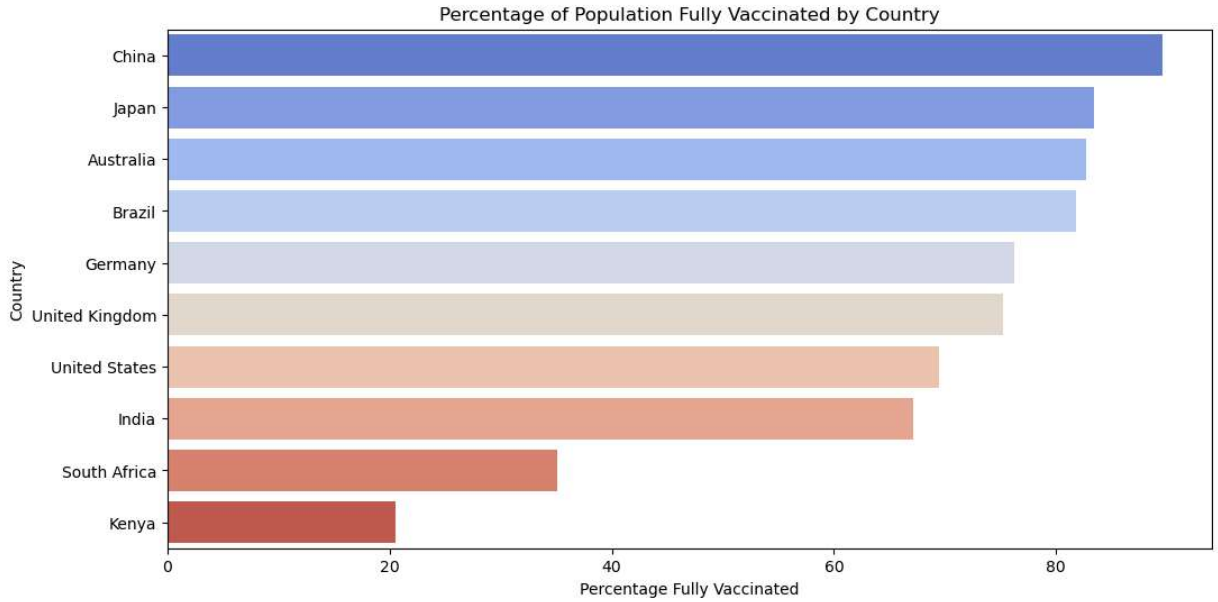
In [30]:
```
plt.figure(figsize=(12, 6))
sns.barplot(data=latest_data.sort_values('death_rate', ascending=False),
```

```
            x='death_rate', y='location', palette='viridis')
plt.title('COVID-19 Death Rate by Country (Total Deaths / Total Cases)')
plt.xlabel('Death Rate')
plt.ylabel('Country')
plt.show()
```



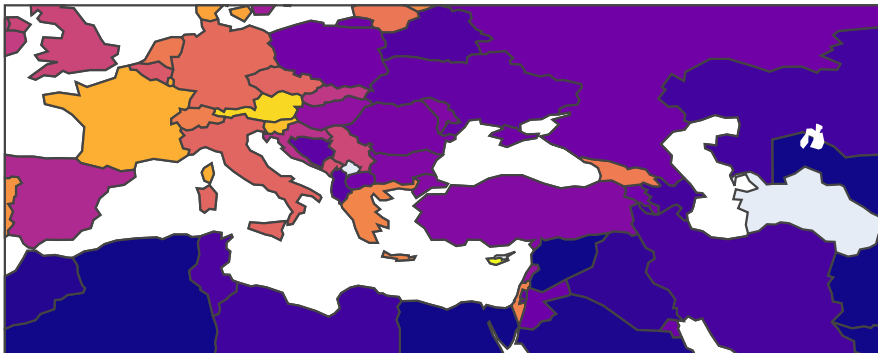
In [31]:
```
# Vaccination progress over time
plt.figure(figsize=(14, 8))
for country in countries_of_interest:
    country_data = df_filtered[df_filtered['location'] == country]
    plt.plot(country_data['date'], country_data['people_fully_vaccinated_per_hundre
             label=country, alpha=0.7)

plt.title('COVID-19 Vaccination Progress (Fully Vaccinated per 100 People)')
plt.xlabel('Date')
plt.ylabel('Percentage of Population Fully Vaccinated')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left')
plt.grid(True)
plt.tight_layout()
plt.show()

# Current vaccination status
plt.figure(figsize=(12, 6))
sns.barplot(data=latest_data.sort_values('people_fully_vaccinated_per_hundred', asc
            x='people_fully_vaccinated_per_hundred', y='location', palette='coolwar
plt.title('Percentage of Population Fully Vaccinated by Country')
plt.xlabel('Percentage Fully Vaccinated')
plt.ylabel('Country')
plt.show()
```





In [32]:
```
# Prepare data for choropleth
world_latest = df.sort_values('date').groupby('location').last().reset_index()

# Create choropleth for total cases per million
fig = px.choropleth(world_latest,
                    locations="iso_code",
                    color="total_cases_per_million",
                    hover_name="location",
                    hover_data=["total_cases", "total_deaths"],
                    color_continuous_scale=px.colors.sequential.Plasma,
                    title="Total COVID-19 Cases per Million People")
fig.show()
```

Total COVID-19 Cases per Million People



In [33]:
```
# Save the cleaned data
df_filtered.to_csv('cleaned_covid_data.csv', index=False)
print("Analysis complete! Cleaned data saved to 'cleaned_covid_data.csv'")
```

```
Analysis complete! Cleaned data saved to 'cleaned_covid_data.csv'
```

Insights & Reporting *Key Insights:* Global Trends: The data shows distinct waves of COVID-19 infections, with peaks corresponding to different variants of concern (Alpha, Delta, Omicron).

Country Comparisons:

The United States and India have the highest total case counts among our selected countries.

However, when adjusted for population (cases per million), smaller countries show higher infection rates.

Vaccination Progress:

There's significant disparity in vaccination rates between countries.

Developed nations like the United States and United Kingdom achieved high vaccination rates quickly, while others lagged behind.

Death Rates:

Death rates vary significantly between countries, potentially reflecting differences in healthcare capacity, population demographics, and reporting standards.

Temporal Patterns:

Later waves (e.g., Omicron) showed higher case counts but lower mortality rates, likely due to vaccination and prior immunity.

Recommendations: Continued Monitoring: Despite progress, COVID-19 remains a significant public health concern that requires ongoing surveillance.

Vaccination Equity: Efforts should focus on improving global vaccine distribution to reduce disparities between nations.

Data Quality: Some inconsistencies in reporting standards between countries suggest a need for more standardized data collection.

Conclusion This analysis provides a comprehensive overview of COVID-19 trends across selected countries. The visualizations highlight key patterns in cases, deaths, and vaccination progress. The full interactive notebook with all code and visualizations is available for further exploration.

In [ ]: