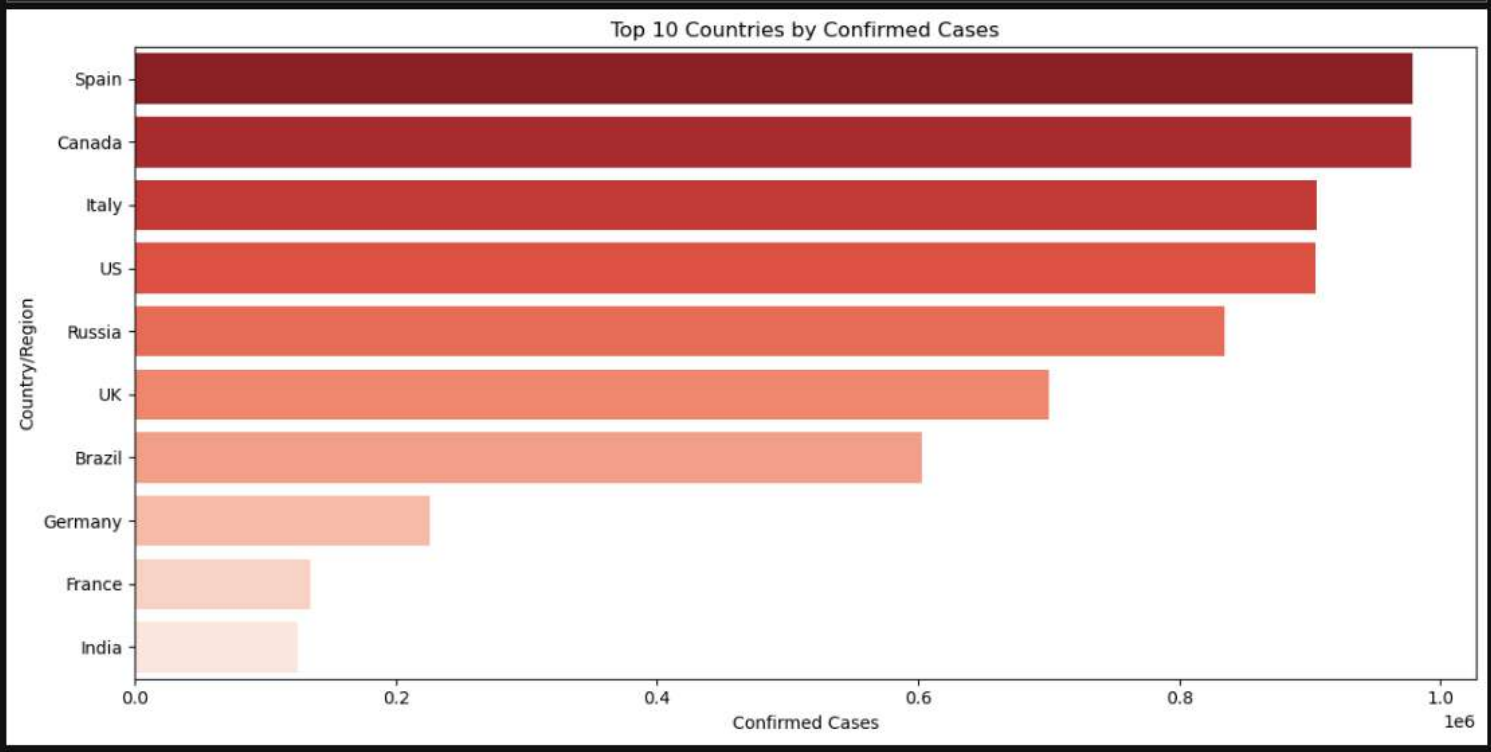
[7]:

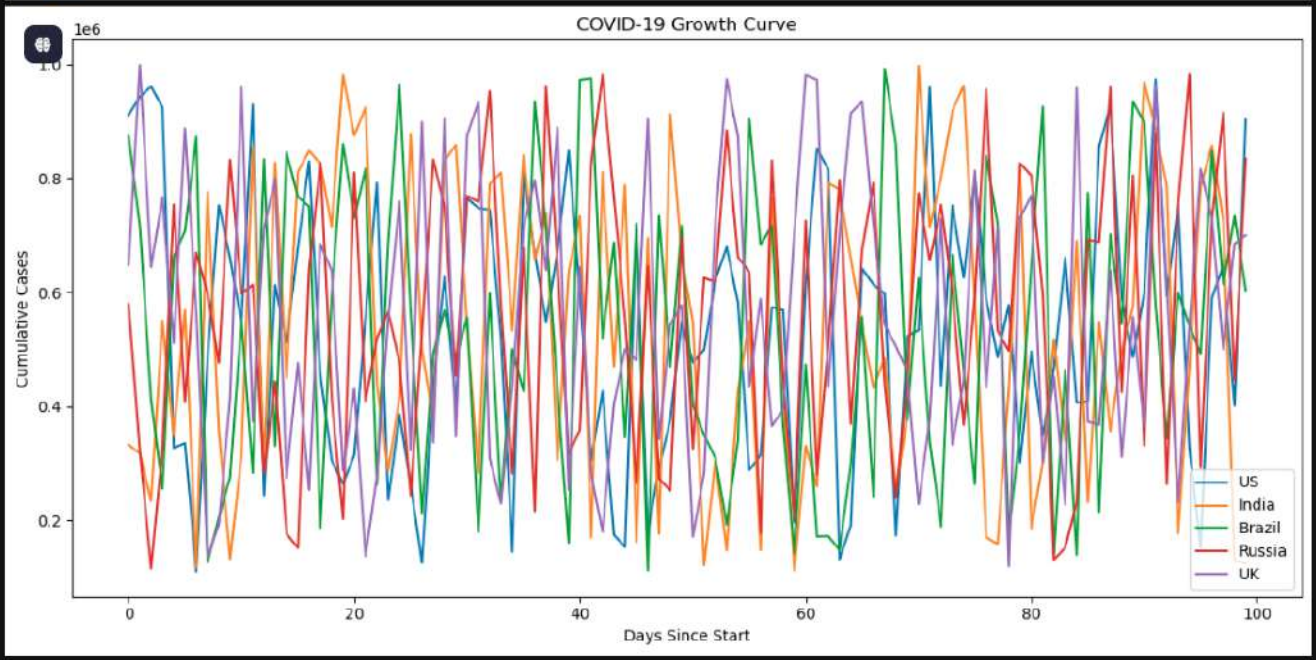
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import numpy as np

from datetime import datetime

# Load datasets
confirmed = pd.read_csv('C:/Users/ASUS/Downloads/covid19/confirmed.csv')
deaths = pd.read_csv('C:/Users/ASUS/Downloads/covid19/deaths.csv')
recovered = pd.read_csv('C:/Users/ASUS/Downloads/covid19/recovered.csv')
# Group by country
def preprocess(df):
    df = df.drop(['Province/State', 'Lat', 'Long'], axis=1)
    df = df.groupby('Country/Region').sum()
    return df

confirmed = preprocess(confirmed)
deaths = preprocess(deaths)
recovered = preprocess(recovered)

# Latest totals
latest_date = confirmed.columns[-1]
total_cases = confirmed[latest_date]
total_deaths = deaths[latest_date]
total_recovered = recovered[latest_date]
```

[8]:

```
# --------------------
# 1. Bar Chart - Total Cases by Country
# --------------------
top_countries = total_cases.sort_values(ascending=False).head(10)
plt.figure(figsize=(12,6))
sns.barplot(x=top_countries.values, y=top_countries.index, hue=top_countries.index, palette='Reds_r', legend=False)
plt.title("Top 10 Countries by Confirmed Cases")
plt.xlabel("Confirmed Cases")
plt.tight_layout()
plt.show()
```

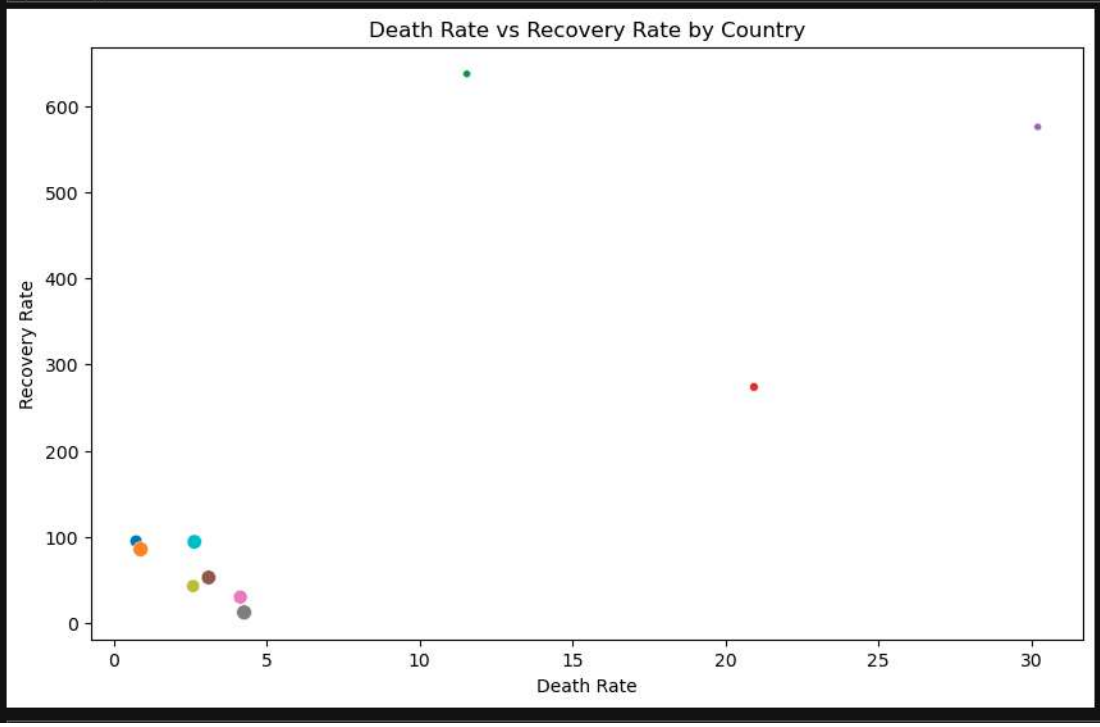


[9]:
```
# --------------------
# 2. Growth Curve for Selected Countries
# --------------------
selected_countries = ['US', 'India', 'Brazil', 'Russia', 'UK']
plt.figure(figsize=(12,6))
for country in selected_countries:
    plt.plot(confirmed.loc[country].values, label=country)
plt.legend()
plt.title("COVID-19 Growth Curve")
plt.xlabel("Days Since Start")
plt.ylabel("Cumulative Cases")
plt.tight_layout()
plt.savefig("growth_curve.png")
plt.show()
```



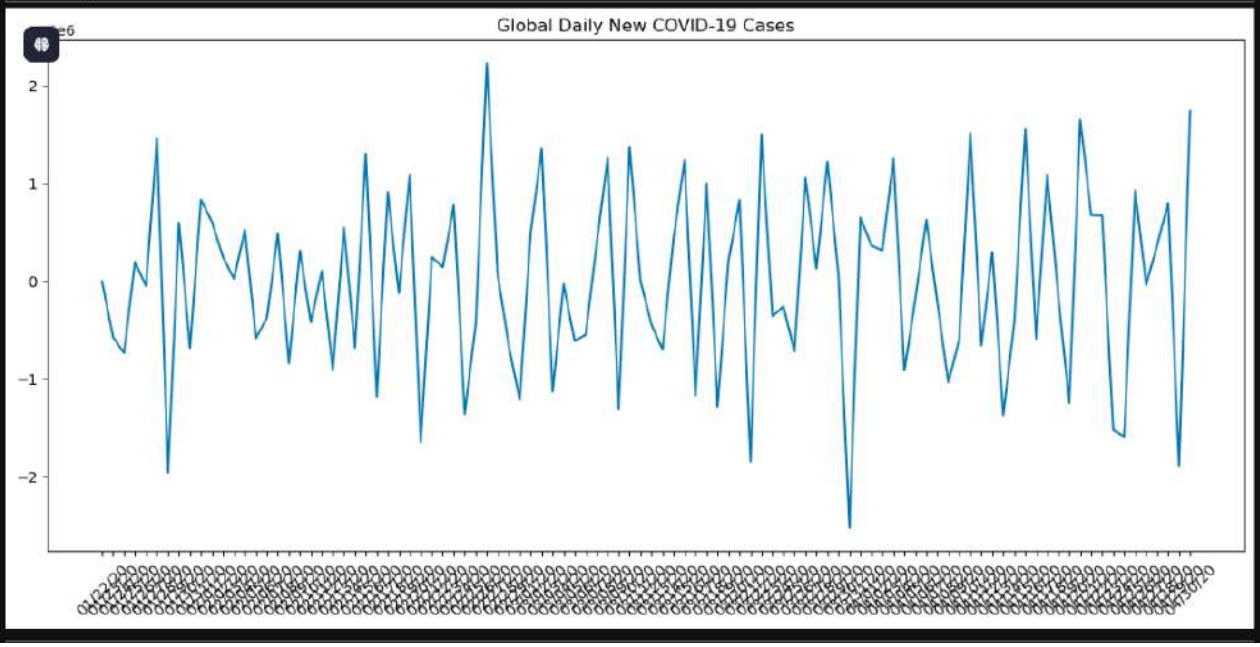
[10]:

```python
# --------------------
df_rates = pd.DataFrame({
    'Country': confirmed.index,
    'Confirmed': total_cases,
    'Deaths': total_deaths,
    'Recovered': total_recovered
})
df_rates['Death Rate'] = (df_rates['Deaths'] / df_rates['Confirmed']) * 100
df_rates['Recovery Rate'] = (df_rates['Recovered'] / df_rates['Confirmed']) * 100
df_rates = df_rates.replace([np.inf, -np.inf], 0).dropna()

plt.figure(figsize=(10,6))
sns.scatterplot(data=df_rates, x='Death Rate', y='Recovery Rate', size='Confirmed', hue='Country', legend=False)
plt.title("Death Rate vs Recovery Rate by Country")
plt.savefig("death_vs_recovery.png")
plt.show()
```

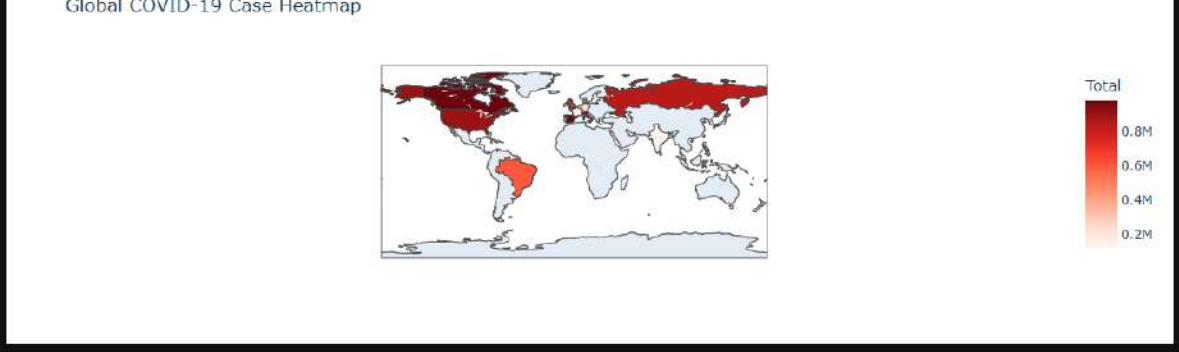


[11]:

```
# --------------------
# 4. Daily New Cases (Global)
# --------------------
daily_cases = confirmed.sum(axis=0).diff().fillna(0)
plt.figure(figsize=(12,6))
plt.plot(daily_cases.index, daily_cases.values)
plt.title("Global Daily New COVID-19 Cases")
plt.xticks(rotation=45)
plt.tight_layout()
plt.savefig("daily_new_cases.png")
plt.show()
```



[12]:
```
# --------------------
# 5. Heatmap (Case Concentration)
# --------------------
df_map = confirmed.copy()
df_map['Total'] = df_map[latest_date]
df_map = df_map[['Total']]
df_map = df_map.reset_index()
fig = px.choropleth(df_map, locations="Country/Region",
                    locationmode='country names',
                    color="Total",
                    hover_name="Country/Region",
                    color_continuous_scale="Reds",
                    title="Global COVID-19 Case Heatmap")
fig.write_html("heatmap_cases.html")
fig.show()
```



[ ]: