In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
data=pd.read_csv("country_wise_latest.csv")
```

In [3]:
```
data.head()
```

Out[3]:

	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases
0	Afghanistan	27532	546	7660	19326	658	42	1502	1.98
1	Albania	1788	39	1086	663	66	1	9	2.18
2	Algeria	11385	811	8078	2496	117	12	135	7.12
3	Andorra	855	52	792	11	1	0	1	6.08
4	Angola	166	8	64	94	11	1	0	4.82

In [4]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 187 entries, 0 to 186
Data columns (total 15 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   Country/Region          187 non-null    object 
 1   Confirmed               187 non-null    int64  
 2   Deaths                  187 non-null    int64  
 3   Recovered               187 non-null    int64  
 4   Active                  187 non-null    int64  
 5   New cases               187 non-null    int64  
 6   New deaths              187 non-null    int64  
 7   New recovered           187 non-null    int64  
 8   Deaths / 100 Cases      187 non-null    float64
 9   Recovered / 100 Cases   187 non-null    float64
 10  Deaths / 100 Recovered  187 non-null    float64
 11  Confirmed last week     187 non-null    int64  
 12  1 week change           187 non-null    int64  
 13  1 week % increase       187 non-null    float64
 14  WHO Region              187 non-null    object 
dtypes: float64(4), int64(9), object(2)
memory usage: 22.0+ KB
```

In [5]:
```
data.shape
```

Out[5]: (187, 15)

In [6]:
```
data = data.rename(columns={'Country/Region': 'Country'})
data.head()
```

Out[6]:

	Country	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	Deaths / 100 Cases	Re
0	Afghanistan	27532	546	7660	19326	658	42	1502	1.98	
1	Albania	1788	39	1086	663	66	1	9	2.18	
2	Algeria	11385	811	8078	2496	117	12	135	7.12	
3	Andorra	855	52	792	11	1	0	1	6.08	
4	Angola	166	8	64	94	11	1	0	4.82	

In [7]:
```
countries_by_deaths = data.sort_values(by='Deaths', ascending=True)
print(countries_by_deaths[['Country', 'Deaths','WHO Region']])
```

```
                              Country  Deaths       WHO Region
142  Saint Vincent and the Grenadines       0         Americas
75                           Holy See       0           Europe
130                Papua New Guinea       0  Western Pacific
118                         Namibia       0           Africa
30                         Cambodia       0  Western Pacific
..                              ...     ...              ...
61                           France   29606           Europe
85                            Italy   34514           Europe
177                  United Kingdom   42373           Europe
23                           Brazil   47748         Americas
173                              US  118434         Americas

[187 rows x 3 columns]
```

In [8]:
```
data.isnull().sum()
```

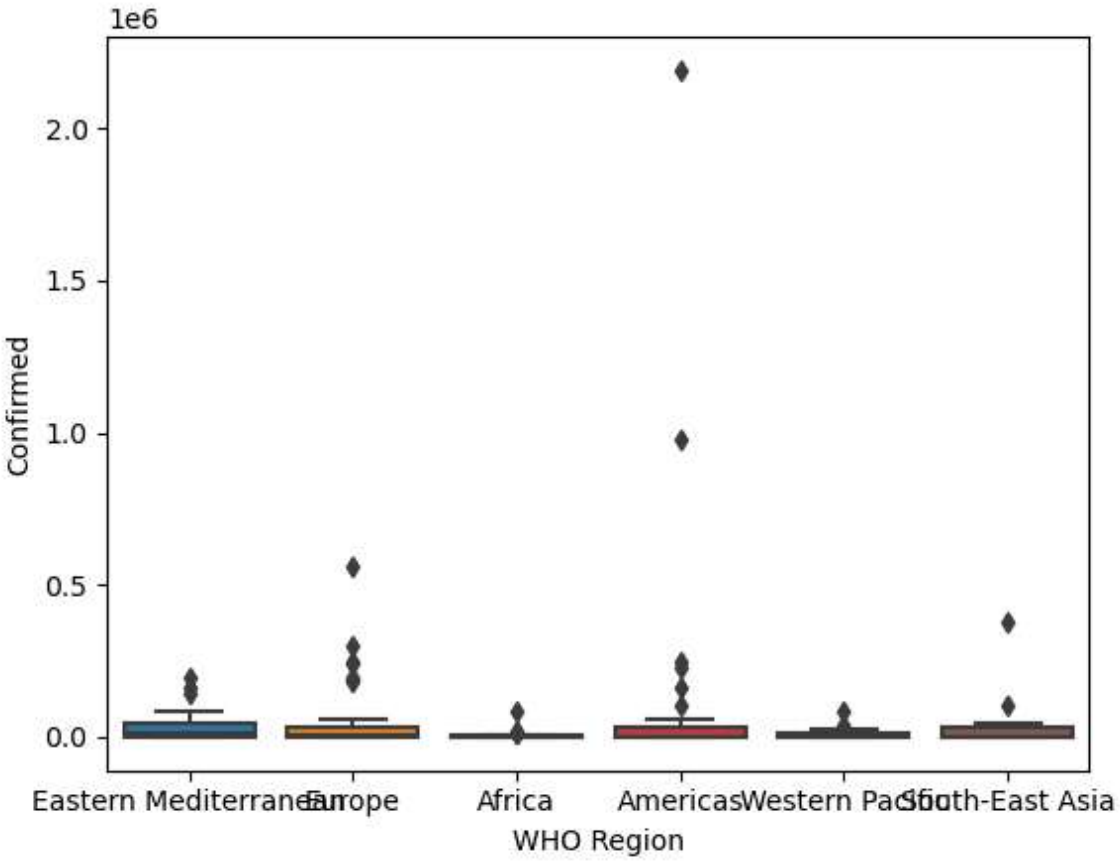
Out[8]:
```
Country                   0
Confirmed                 0
Deaths                    0
Recovered                 0
Active                    0
New cases                 0
New deaths                0
New recovered             0
Deaths / 100 Cases        0
Recovered / 100 Cases     0
Deaths / 100 Recovered    0
Confirmed last week       0
1 week change             0
1 week % increase         0
WHO Region                0
dtype: int64
```

In [9]:
```
data.columns
```

Out[9]:
```
Index(['Country', 'Confirmed', 'Deaths', 'Recovered', 'Active', 'New case
s',
       'New deaths', 'New recovered', 'Deaths / 100 Cases',
       'Recovered / 100 Cases', 'Deaths / 100 Recovered',
       'Confirmed last week', '1 week change', '1 week % increase',
       'WHO Region'],
      dtype='object')
```

In [10]:
```
sns.boxplot(x="WHO Region",y="Confirmed",data=data)
```

Out[10]:
```
<Axes: xlabel='WHO Region', ylabel='Confirmed'>
```

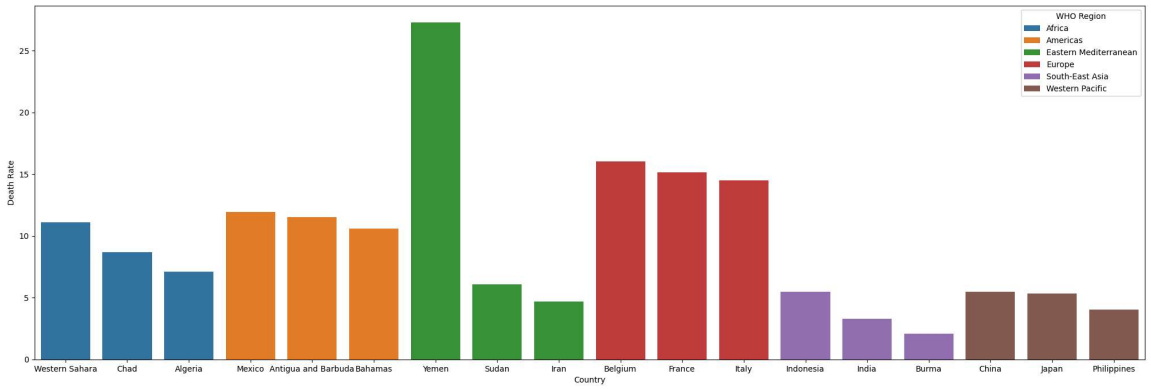


finding death rate of top 3 countries in each region

In [11]:
```
data['Death Rate'] = (data['Deaths'] / data['Confirmed']) * 100
top_countries_per_region = data.groupby('WHO Region').apply(
    lambda x: x.nlargest(3, 'Death Rate')
).reset_index(drop=True)
```

In [12]:
```
plt.figure(figsize=(25, 8))
sns.barplot(x='Country', y='Death Rate', hue='WHO Region', data=top_countri
```

Out[12]:
```
<Axes: xlabel='Country', ylabel='Death Rate'>
```

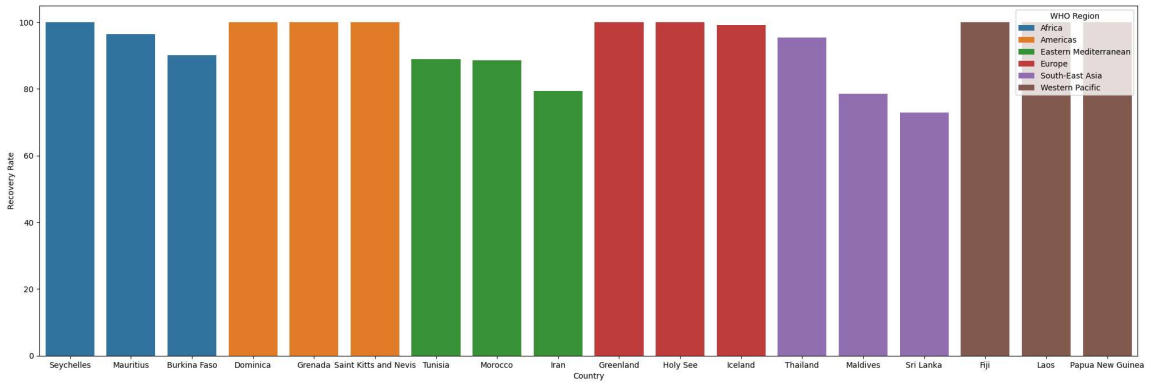


finding recovery rate of top 3 countries in each region¶

In [13]:
```
data['Recovery Rate'] = (data['Recovered'] / data['Confirmed']) * 100
top_countries_per_region = data.groupby('WHO Region').apply(
    lambda x: x.nlargest(3, 'Recovery Rate')
).reset_index(drop=True)
```

In [14]:
```
plt.figure(figsize=(25, 8))
sns.barplot(x='Country', y='Recovery Rate', hue='WHO Region', data=top_coun
```

Out[14]:
```
<Axes: xlabel='Country', ylabel='Recovery Rate'>
```



finding countries with most cases in each region top 3

In [15]:
```
top_countries_most_cases_per_region = data.groupby('WHO Region').apply(
    lambda x: x.nlargest(3, 'Confirmed')
).reset_index(drop=True)
print(top_countries_most_cases_per_region[['WHO Region', 'Country', 'Confir
```

```
                WHO Region         Country  Confirmed
0                   Africa    South Africa      83890
1                   Africa         Nigeria      18480
2                   Africa           Ghana      12929
3                 Americas              US    2191052
4                 Americas          Brazil     978142
5                 Americas            Peru     244388
6   Eastern Mediterranean            Iran     197647
7   Eastern Mediterranean        Pakistan     165062
8   Eastern Mediterranean    Saudi Arabia     145991
9                   Europe          Russia     560321
10                  Europe  United Kingdom     301935
11                  Europe           Spain     245268
12        South-East Asia           India     380532
13        South-East Asia      Bangladesh     102292
14        South-East Asia       Indonesia      42762
15        Western Pacific           China      84494
16        Western Pacific       Singapore      41473
17        Western Pacific     Philippines      27799
```

comparision by region

In [16]:
```
regional_data = data.groupby('WHO Region').agg({
    'Confirmed': 'sum',
    'Deaths': 'sum',
    'Recovered': 'sum'
}).reset_index()
```

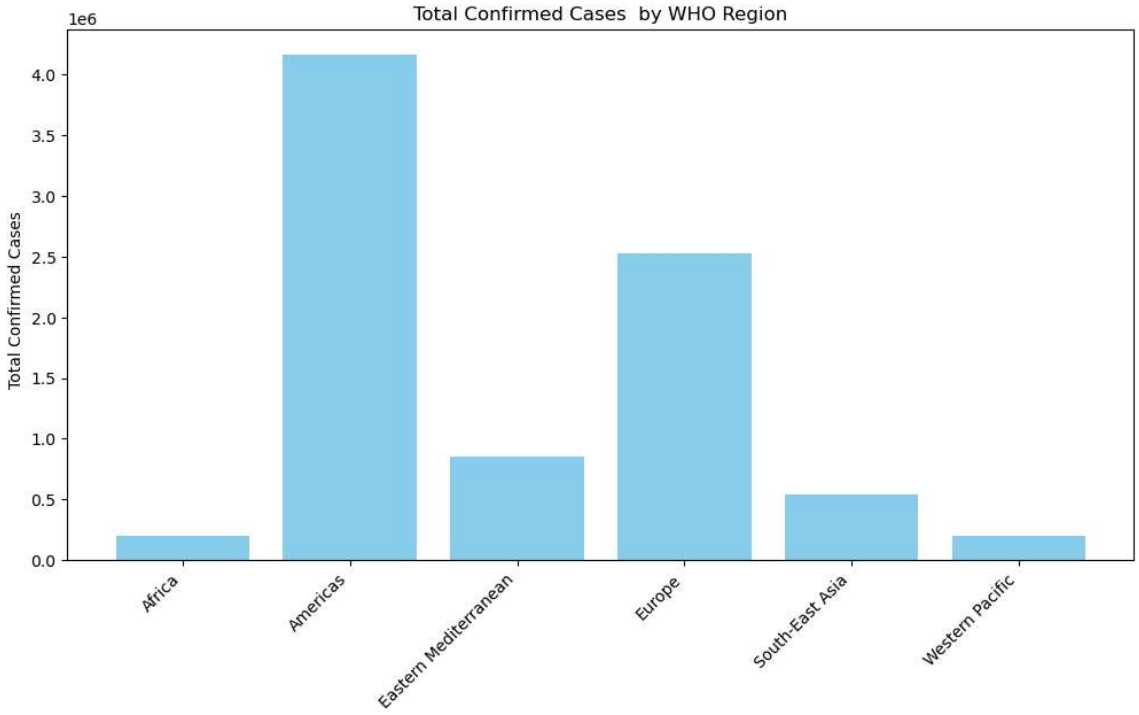
In [17]:
```
regional_data['Death Rate (%)'] = (regional_data['Deaths'] / regional_data[
regional_data['Recovery Rate (%)'] = (regional_data['Recovered'] / regional
```

In [18]:
```
print(regional_data)
```

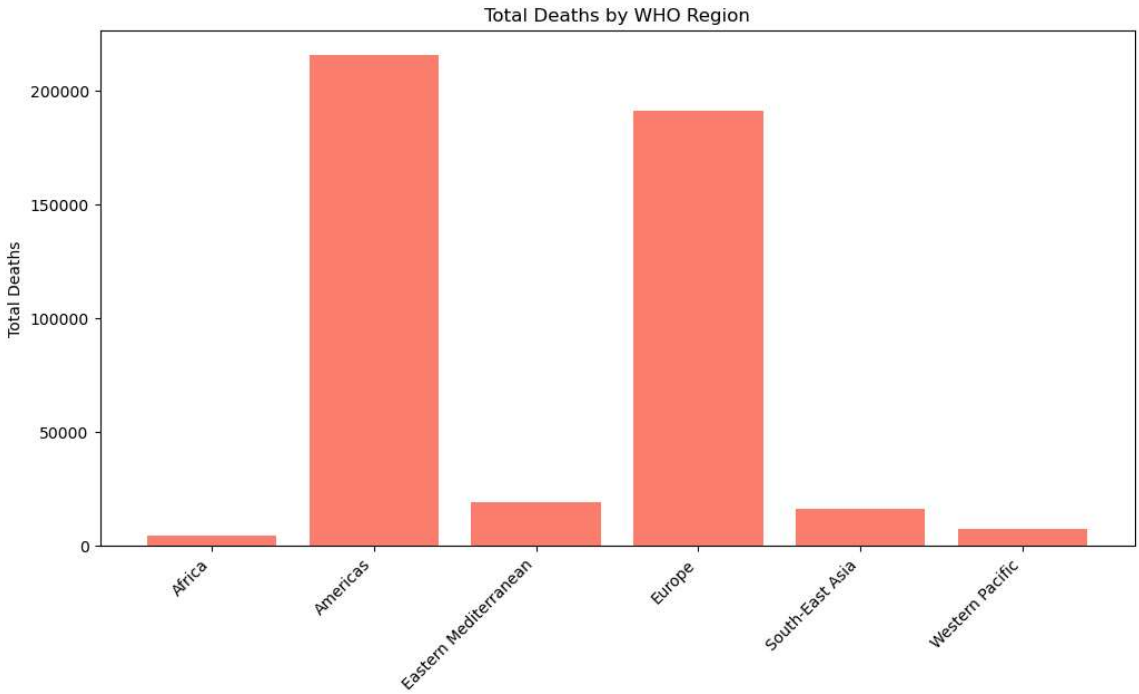
```
              WHO Region  Confirmed  Deaths  Recovered  Death Rate (%)  \
0                 Africa     199343    4591      95773        2.303066
1               Americas    4164325  215632    1675117        5.178078
2  Eastern Mediterranean     856105   19036     514839        2.223559
3                 Europe    2525014  191096    1372022        7.568116
4        South-East Asia     541036   16360     269166        3.023828
5        Western Pacific     202412    7250     163465        3.581803

   Recovery Rate (%)
0          48.044326
1          40.225415
2          60.137366
3          54.337204
4          49.750109
5          80.758552
```

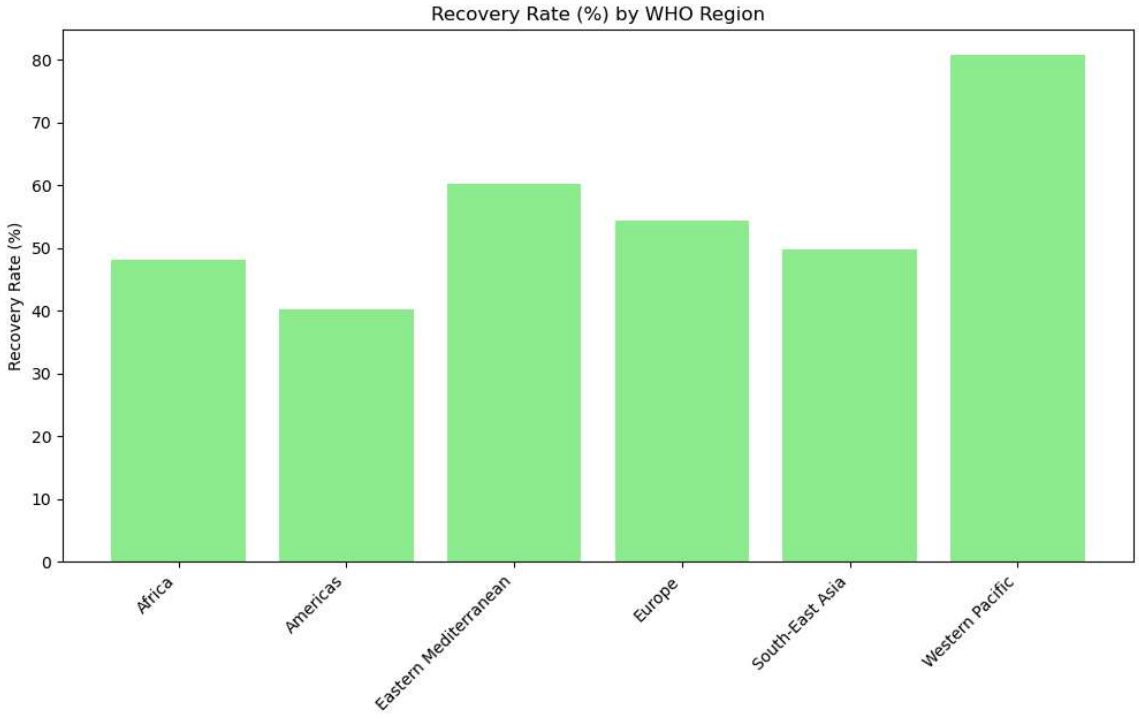
In [19]:
```
plt.figure(figsize=(12, 6))
plt.bar(regional_data['WHO Region'], regional_data['Confirmed'], color='sky
plt.title('Total Confirmed Cases  by WHO Region')
plt.ylabel('Total Confirmed Cases  ')
plt.xticks(rotation=45, ha='right')
plt.show()
```



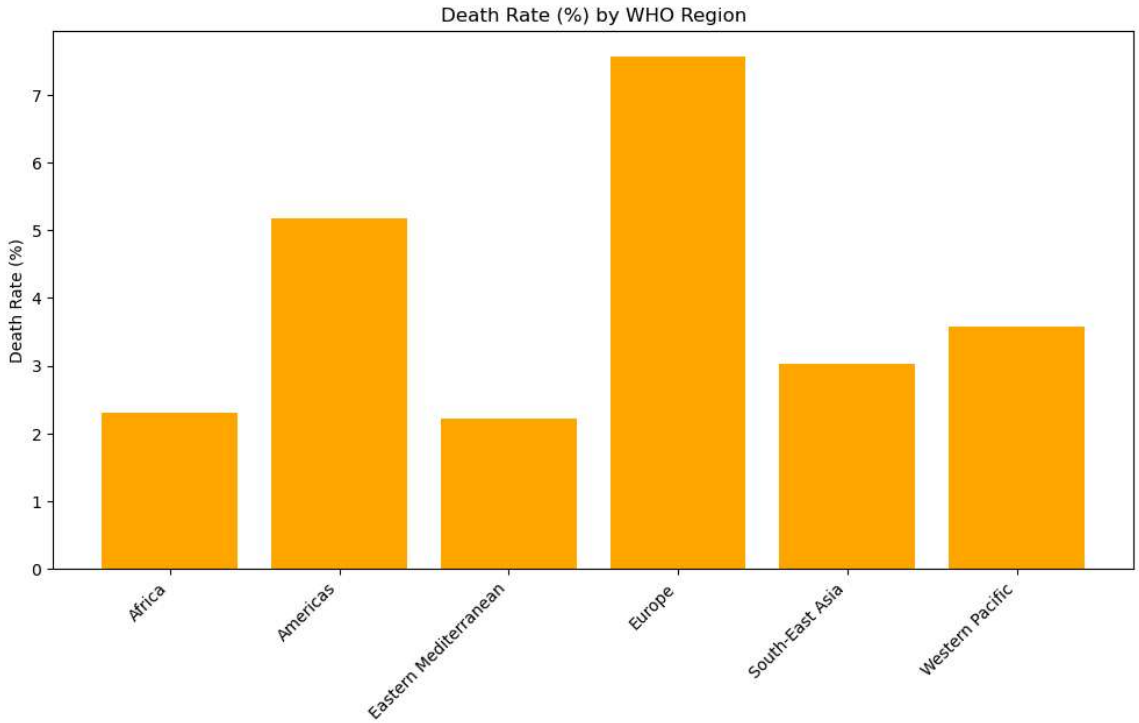
In [20]:
```
plt.figure(figsize=(12, 6))
plt.bar(regional_data['WHO Region'], regional_data['Deaths'], color='salmon
plt.title('Total Deaths by WHO Region')
plt.ylabel('Total Deaths')
plt.xticks(rotation=45, ha='right')
plt.show()
```



In [21]:
```
plt.figure(figsize=(12, 6))
plt.bar(regional_data['WHO Region'], regional_data['Recovery Rate (%)'], co
plt.title('Recovery Rate (%) by WHO Region')
plt.ylabel('Recovery Rate (%)')
plt.xticks(rotation=45, ha='right')
plt.show()
```



In [22]:
```
plt.figure(figsize=(12, 6))
plt.bar(regional_data['WHO Region'], regional_data['Death Rate (%)'], color
plt.title('Death Rate (%) by WHO Region')
plt.ylabel('Death Rate (%)')
plt.xticks(rotation=45, ha='right')
plt.show()
```



Situation According to Death & Recovery Rate

In [27]:
```
data['Recovery Rate (%)'] = (data['Recovered'] / data['Confirmed']) * 100
data['Death Rate (%)'] = (data['Deaths'] / data['Confirmed']) * 100

recovery_criteria = data['Recovery Rate (%)'].mean()
death_criteria = data['Death Rate (%)'].mean()

data['Situation'] = data.apply(
    lambda row: 'Critical' if (row['Recovery Rate (%)'] < recovery_criteria
                else 'Not Critical', axis=1
)

critical_countries = data[data['Critical'] == 'Critical']

print(critical_countries[['Country', 'Recovery Rate (%)', 'Death Rate (%)',
```

	Country	Recovery Rate (%)	Death Rate (%)	Situation
4	Angola	38.554217	4.819277	Critical
16	Belgium	27.712600	16.045271	Critical
23	Brazil	54.652596	4.881500	Critical
25	Bulgaria	52.830702	5.171475	Critical
32	Canada	0.000000	8.207021	Critical
37	Colombia	37.343100	3.417713	Critical
51	Ecuador	49.133348	8.324338	Critical
52	Egypt	26.821579	3.842417	Critical
61	France	37.901491	15.161416	Critical
67	Greece	42.578246	5.825844	Critical
70	Guatemala	19.295585	3.783283	Critical
73	Guyana	55.737705	6.557377	Critical
80	Indonesia	39.282541	5.469810	Critical
82	Iraq	44.068126	3.328538	Critical
98	Liberia	46.125461	6.088561	Critical
109	Mauritania	22.689769	4.001650	Critical
112	Moldova	55.333435	3.387761	Critical
120	Netherlands	0.361470	12.312197	Critical
125	North Macedonia	39.365352	4.631218	Critical
133	Philippines	25.504515	4.014533	Critical
134	Poland	49.385781	4.243108	Critical
149	Sierra Leone	55.817610	4.009434	Critical
159	Sudan	36.982544	6.072319	Critical
161	Sweden	0.000000	9.016291	Critical
163	Syria	0.000000	3.743316	Critical
166	Tanzania	35.952849	4.125737	Critical
173	US	27.343714	5.405349	Critical
177	United Kingdom	0.434862	14.033815	Critical
184	Yemen	30.033003	27.282728	Critical

In [ ]: