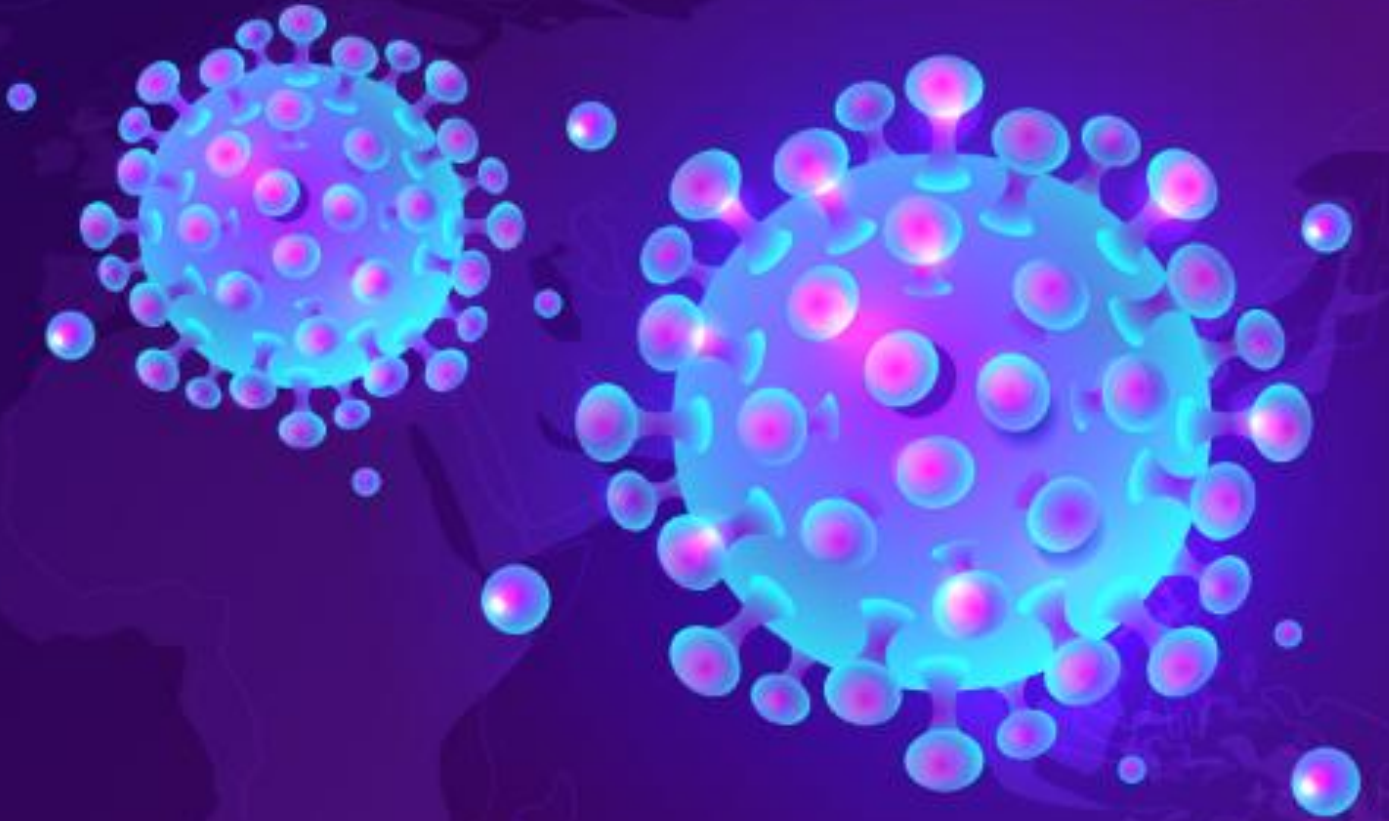


COVID-19

This project aims to analyze the global spread of COVID-19 using Python, focusing on tracking the number of confirmed cases, deaths, and recoveries. Data analysis libraries such as Pandas and Plotly were used to generate statistics and interactive visualizations that help in understanding the progression of the pandemic across different countries over time.



DATA

Index	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region
0	2020-01-22	Afghanistan	0	0	0	0	0	0	0	Eastern Mediterranean
1	2020-01-22	Albania	0	0	0	0	0	0	0	Europe
2	2020-01-22	Algeria	0	0	0	0	0	0	0	Africa
3	2020-01-22	Andorra	0	0	0	0	0	0	0	Europe
4	2020-01-22	Angola	0	0	0	0	0	0	0	Africa
5	2020-01-22	Antigua and Barbuda	0	0	0	0	0	0	0	Americas
6	2020-01-22	Argentina	0	0	0	0	0	0	0	Americas
7	2020-01-22	Armenia	0	0	0	0	0	0	0	Europe
8	2020-01-22	Australia	0	0	0	0	0	0	0	Western Pacific
9	2020-01-22	Austria	0	0	0	0	0	0	0	Europe
10	2020-01-22	Azerbaijan	0	0	0	0	0	0	0	Europe
11	2020-01-22	Bahamas	0	0	0	0	0	0	0	Americas
12	2020-01-22	Bahrain	0	0	0	0	0	0	0	Eastern Mediterranean
13	2020-01-22	Bangladesh	0	0	0	0	0	0	0	South-East Asia
14	2020-01-22	Barbados	0	0	0	0	0	0	0	Americas
15	2020-01-22	Belarus	0	0	0	0	0	0	0	Europe
16	2020-01-22	Belgium	0	0	0	0	0	0	0	Europe
17	2020-01-22	Belize	0	0	0	0	0	0	0	Americas
18	2020-01-22	Benin	0	0	0	0	0	0	0	Africa
19	2020-01-22	Bhutan	0	0	0	0	0	0	0	South-East Asia
20	2020-01-22	Bolivia	0	0	0	0	0	0	0	Americas
21	2020-01-22	Bosnia and Herzegovina	0	0	0	0	0	0	0	Europe
22	2020-01-22	Botswana	0	0	0	0	0	0	0	Africa
23	2020-01-22	Brazil	0	0	0	0	0	0	0	Americas
24	2020-01-22	Brunei	0	0	0	0	0	0	0	Western Pacific
25	2020-01-22	Bulgaria	0	0	0	0	0	0	0	Europe
26	2020-01-22	Burkina Faso	0	0	0	0	0	0	0	Africa
27	2020-01-22	Burma	0	0	0	0	0	0	0	South-East Asia
28	2020-01-22	Burundi	0	0	0	0	0	0	0	Africa
29	2020-01-22	Cabo Verde	0	0	0	0	0	0	0	Africa
30	2020-01-22	Cambodia	0	0	0	0	0	0	0	Western Pacific
31	2020-01-22	Cameroon	0	0	0	0	0	0	0	Africa
32	2020-01-22	Canada	0	0	0	0	0	0	0	Americas
33	2020-01-22	Central African Republic	0	0	0	0	0	0	0	Africa
34	2020-01-22	Chad	0	0	0	0	0	0	0	Africa
35	2020-01-22	Chile	0	0	0	0	0	0	0	Americas
36	2020-01-22	China	548	17	28	503	0	0	0	Western Pacific
37	2020-01-22	Colombia	0	0	0	0	0	0	0	Americas
38	2020-01-22	Comoros	0	0	0	0	0	0	0	Africa
39	2020-01-22	Congo (Brazzaville)	0	0	0	0	0	0	0	Africa

DESCRIBE



```
1 covid_19.describe()
```

	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered
count	3.515600e+04	35156.000000	3.515600e+04	3.515600e+04	35156.000000	35156.000000	35156.000000
mean	2.356663e+04	1234.068239	1.104813e+04	1.128443e+04	469.36375	18.603339	269.315593
std	1.499818e+05	7437.238354	6.454640e+04	8.997149e+04	3005.86754	115.706351	2068.063852
min	0.000000e+00	0.000000	0.000000e+00	-2.000000e+00	0.000000	-1918.000000	-16298.000000
25%	1.000000e+00	0.000000	0.000000e+00	0.000000e+00	0.000000	0.000000	0.000000
50%	2.500000e+02	4.000000	3.300000e+01	8.500000e+01	2.000000	0.000000	0.000000
75%	3.640250e+03	78.250000	1.286250e+03	1.454000e+03	75.000000	1.000000	20.000000
max	4.290259e+06	148011.000000	1.846641e+06	2.816444e+06	77255.000000	3887.000000	140050.000000

MISSING VALUUS



```
1 covid_19.isnull().sum()
```

Date	0
Country/Region	0
Confirmed	0
Deaths	0
Recovered	0
Active	0
New cases	0
New deaths	0
New recovered	0
WHO Region	0

CORE ANALYSIS

AESTHETICS BY COUNTRY



```
1 top_confirmed = covid_19.groupby('Country/Region')['Confirmed'].max().sort_values(ascending=False).head(10)
2
```

Confirmed	
Country/Region	
US	4290259
Brazil	2442375
India	1480073
Russia	816680
South Africa	452529
Mexico	395489
Peru	389717
Chile	347923
United Kingdom	301708
Iran	293606



```
1 top_deaths = covid_19.groupby('Country/Region')['Deaths'].max().sort_values(ascending=False).head(10)
2
```

Deaths	
Country/Region	
US	148011
Brazil	87618
United Kingdom	45844
Mexico	44022
Italy	35112
India	33408
France	30212
Spain	28752
Peru	18418
Iran	15912



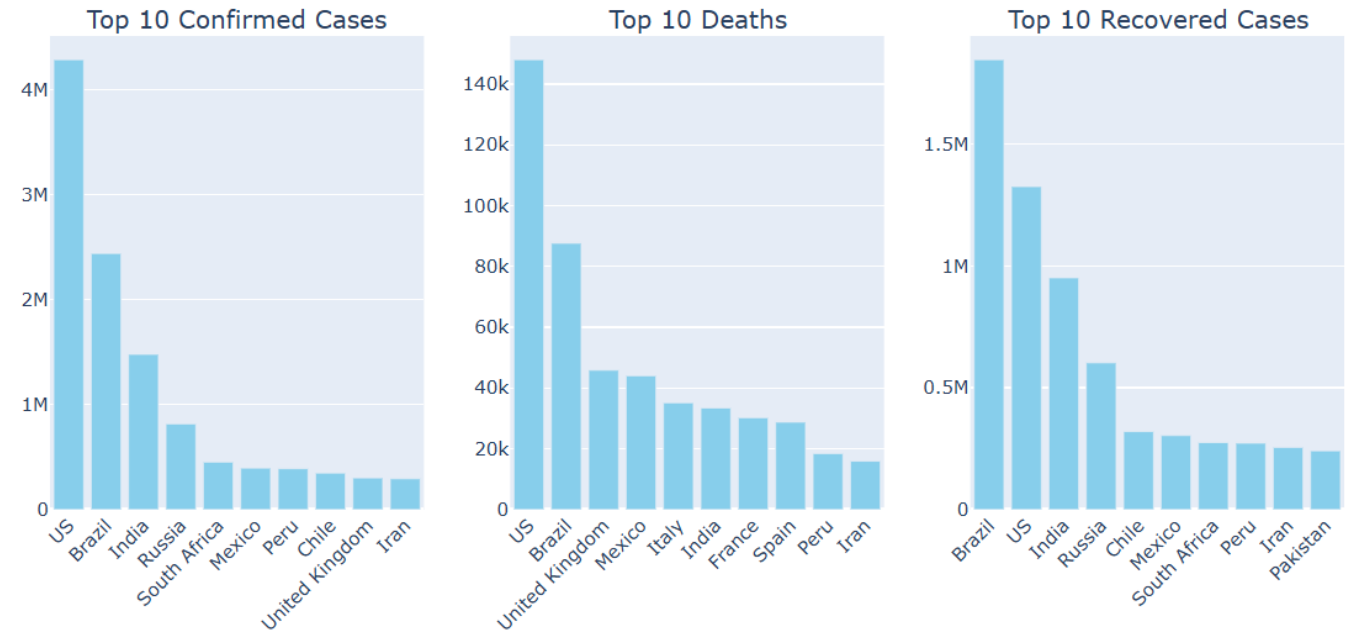
```
1 top_recovered = covid_19.groupby('Country/Region')['Recovered'].max().sort_values(ascending=False).head(10)
2
```

Recovered	
Country/Region	
Brazil	1846641
US	1325804
India	951166
Russia	602249
Chile	319954
Mexico	303810
South Africa	274925
Peru	272547
Iran	255144
Pakistan	241026

AESTHETICS BY COUNTRY

```
1
2 # Convert Series to DataFrames
3 top_confirmed_df = top_confirmed.reset_index()
4 top_deaths_df = top_deaths.reset_index()
5 top_recovered_df = top_recovered.reset_index()
6
7 # Create 3 columns (subplots) in one row
8 fig = make_subplots(
9     rows=1, cols=3,
10     subplot_titles=[
11         'Top 10 Confirmed Cases',
12         'Top 10 Deaths',
13         'Top 10 Recovered Cases'
14     ]
15 )
16
17 ## -----Top 10 Countries with the Highest Number of Confirmed Cases-----
18
19 fig.add_trace(
20     go.Bar(
21         x=top_confirmed_df['Country/Region'],
22         y=top_confirmed_df['Confirmed'],
23         name='Confirmed',
24         marker_color='skyblue'
25     ),
26     row=1, col=1
27 )
28
29 ## -----Top 10 Countries with the Highest Number of Deaths-----
30 fig.add_trace(
31     go.Bar(
32         x=top_deaths_df['Country/Region'],
33         y=top_deaths_df['Deaths'],
34         name='Deaths',
35         marker_color='skyblue'
36     ),
37     row=1, col=2
38 )
39
40 ## -----Top 10 Countries with the Highest Number of Recovered Cases-----
41 fig.add_trace(
42     go.Bar(
43         x=top_recovered_df['Country/Region'],
44         y=top_recovered_df['Recovered'],
45         name='Recovered',
46         marker_color='skyblue'
47     ),
48     row=1, col=3
49 )
50
51 # Update format
52 fig.update_layout(
53     title_text='COVID-19 Top 10 Countries: Confirmed, Deaths, and Recovered',
54     showlegend=False,
55     height=500, width=1000,
56     xaxis_tickangle=-45,
57     xaxis2_tickangle=-45,
58     xaxis3_tickangle=-45
59 )
60
61 )
62
63 fig.show()
64
```

COVID-19 Top 10 Countries: Confirmed, Deaths, and Recovered



FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)



```
1 latest = covid_19[covid_19['Date'] == covid_19['Date'].max()].copy()
2 latest = latest[latest['Confirmed'] > 0] # Avoid division by zero
3
4 latest['Death_Rate'] = latest['Deaths'] / latest['Confirmed']
5 latest['Recovery_Rate'] = latest['Recovered'] / latest['Confirmed']
6
7 latest.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region	Death_Rate	Recovery_Rate
34969	2020-07-27	Afghanistan	36263	1269	25198	9796	106	10	18	Eastern Mediterranean	0.034994	0.694868
34970	2020-07-27	Albania	4880	144	2745	1991	117	6	63	Europe	0.029508	0.562500
34971	2020-07-27	Algeria	27973	1163	18837	7973	616	8	749	Africa	0.041576	0.673399
34972	2020-07-27	Andorra	907	52	803	52	10	0	0	Europe	0.057332	0.885336
34973	2020-07-27	Angola	950	41	242	667	18	1	0	Africa	0.043158	0.254737



```
1 top_death_rate = latest.sort_values(by='Death_Rate', ascending=False).head(10)
2 top_death_rate.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region	Death_Rate	Recovery_Rate
35153	2020-07-27	Yemen	1691	483	833	375	10	4	36	Eastern Mediterranean	0.285630	0.492608
35146	2020-07-27	United Kingdom	301708	45844	1437	254427	688	7	3	Europe	0.151948	0.004763
34985	2020-07-27	Belgium	66428	9822	17452	39154	402	1	14	Europe	0.147859	0.262721
35054	2020-07-27	Italy	246286	35112	198593	12581	168	5	147	Europe	0.142566	0.806351
35030	2020-07-27	France	220352	30212	81212	108928	2551	17	267	Europe	0.137108	0.368556

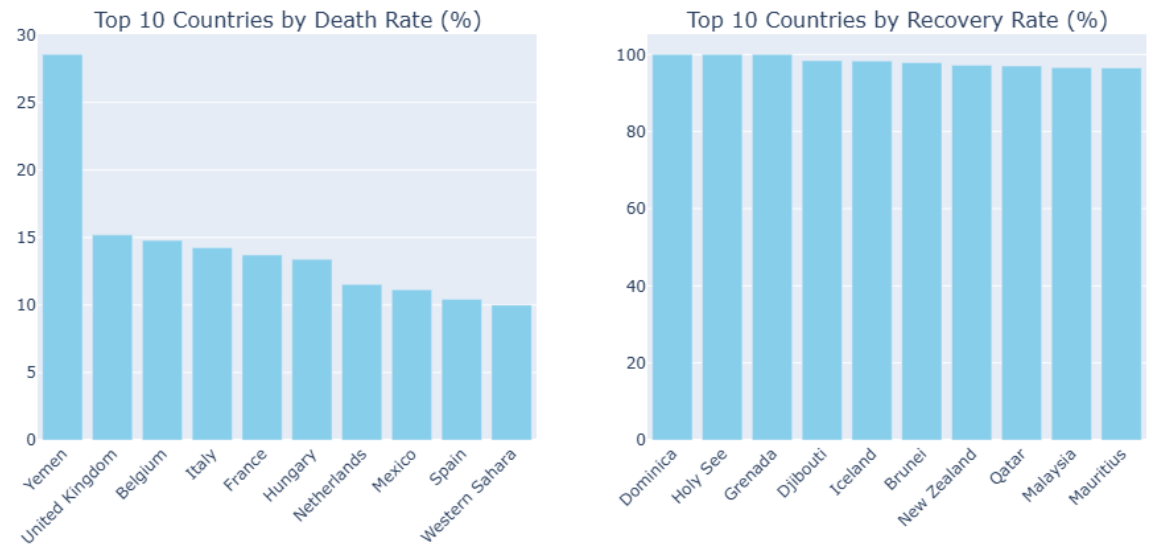


```
1 top_recovery_rate = latest.sort_values(by='Recovery_Rate', ascending=False).head(10)
2 top_recovery_rate.head()
```


FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)

```
1
2
3 # Convert Series to DataFrames
4 top_death_rate_df = top_death_rate[['Country/Region', 'Death_Rate']].reset_index(drop=True)
5 top_recovery_rate_df = top_recovery_rate[['Country/Region', 'Recovery_Rate']].reset_index(drop=True)
6
7
8 fig = make_subplots(
9     rows=1, cols=2,
10     subplot_titles=[
11         'Top 10 Countries by Death Rate (%)',
12         'Top 10 Countries by Recovery Rate (%)'
13     ]
14 )
15
16 # -----Drawing the highest mortality rates-----
17 fig.add_trace(
18     go.Bar(
19         x=top_death_rate_df['Country/Region'],
20         y=top_death_rate_df['Death_Rate'] * 100,
21         name='Death Rate (%)',
22         marker_color='skyblue'
23     ),
24     row=1, col=1
25 )
26
27 # -----Drawing the highest recovery rates-----
28 fig.add_trace(
29     go.Bar(
30         x=top_recovery_rate_df['Country/Region'],
31         y=top_recovery_rate_df['Recovery_Rate'] * 100,
32         name='Recovery Rate (%)',
33         marker_color='skyblue'
34     ),
35     row=1, col=2
36 )
37
38 # Update format
39 fig.update_layout(
40     title_text='COVID-19: Death vs Recovery Rates by Country (Top 10)',
41     showlegend=False,
42     height=500, width=1000,
43     xaxis_tickangle=-45,
44     xaxis2_tickangle=-45
45 )
46
47 fig.show()
48
```

COVID-19: Death vs Recovery Rates by Country (Top 10)



FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)



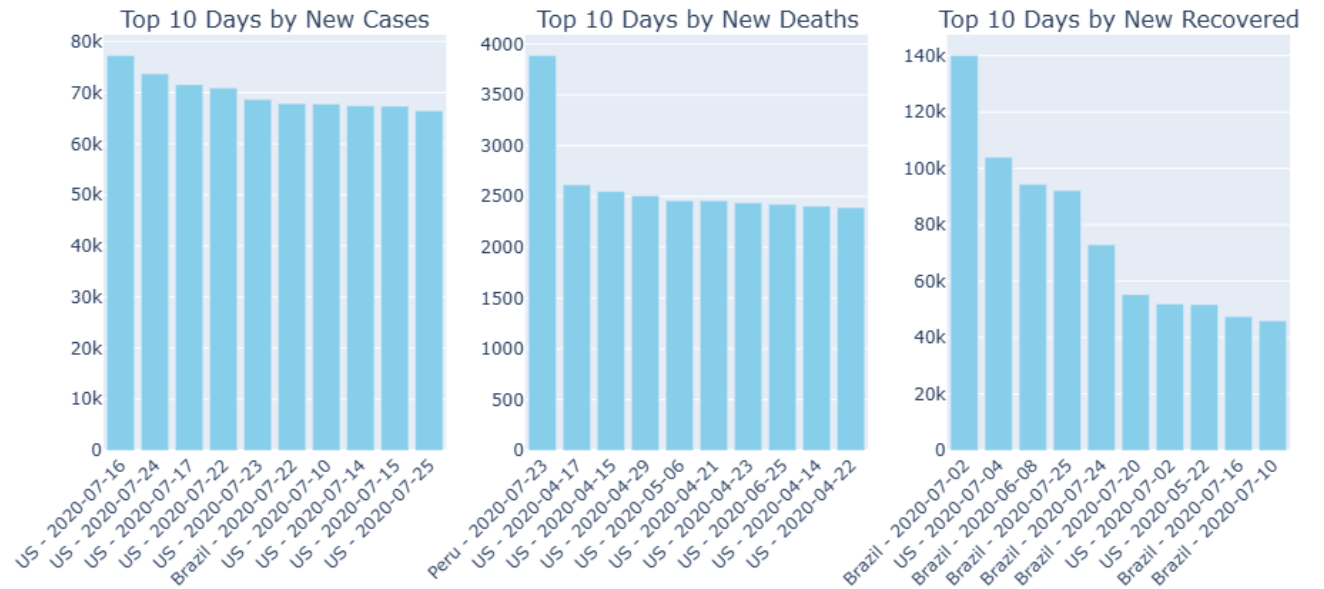
```
1 top_days_cases = covid_19.sort_values(by='New cases', ascending=False).head(10)
2 top_days_deaths = covid_19.sort_values(by='New deaths', ascending=False).head(10)
3 top_days_recovered = covid_19.sort_values(by='New recovered', ascending=False).head(10)
4
5 top_days_cases.head()
6 top_days_deaths.head()
7 top_days_recovered.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region
30317	2020-07-02	Brazil	1496858	61884	957692	477282	48105	1252	140050	Americas
30841	2020-07-04	US	2841241	129689	894325	1817227	45880	247	103921	Americas
25829	2020-06-08	Brazil	707412	37134	378257	292021	15654	679	94305	Americas
34618	2020-07-25	Brazil	2394513	86449	1785359	522705	51147	1211	92145	Americas
34431	2020-07-24	Brazil	2343366	85238	1693214	564914	55891	1156	72901	Americas

FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)

```
1 # Add a title containing the country and date
2 top_days_cases['Label'] = top_days_cases['Country/Region'] + ' - ' + top_days_cases['Date'].astype(str)
3 top_days_deaths['Label'] = top_days_deaths['Country/Region'] + ' - ' + top_days_deaths['Date'].astype(str)
4 top_days_recovered['Label'] = top_days_recovered['Country/Region'] + ' - ' + top_days_recovered['Date'].astype(str)
5
6
7 # Create 3 columns (subplots) in one row
8 fig = make_subplots(
9     rows=1, cols=3,
10     subplot_titles=[
11         'Top 10 Days by New Cases',
12         'Top 10 Days by New Deaths',
13         'Top 10 Days by New Recovered'
14     ]
15 )
16
17 # -----New injuries-----
18 fig.add_trace(
19     go.Bar(
20         x=top_days_cases['Label'],
21         y=top_days_cases['New cases'],
22         name='New Cases',
23         marker_color='skyblue'
24     ),
25     row=1, col=1
26 )
27
28 # -----New deaths-----
29 fig.add_trace(
30     go.Bar(
31         x=top_days_deaths['Label'],
32         y=top_days_deaths['New deaths'],
33         name='New Deaths',
34         marker_color='skyblue'
35     ),
36     row=1, col=2
37 )
38
39 # -----New recovery cases-----
40 fig.add_trace(
41     go.Bar(
42         x=top_days_recovered['Label'],
43         y=top_days_recovered['New recovered'],
44         name='New Recovered',
45         marker_color='skyblue'
46     ),
47     row=1, col=3
48 )
49
50
51 fig.update_layout(
52     title_text='Top 10 Days: New Cases, Deaths, and Recoveries',
53     showlegend=False,
54     height=500, width=1000,
55     xaxis_tickangle=-45,
56     xaxis2_tickangle=-45,
57     xaxis3_tickangle=-45
58 )
59
60 fig.show()
```

Top 10 Days: New Cases, Deaths, and Recoveries



FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)



```
1 df_nonzero = covid_19[covid_19['New cases'] > 0].copy()
2 df_nonzero['Death_Rate'] = df_nonzero['New deaths'] / df_nonzero['New cases']
3 df_nonzero['Recovery_Rate'] = df_nonzero['New recovered'] / df_nonzero['New cases']
4
5 df_nonzero.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region	Death_Rate	Recovery_Rate
223	2020-01-23	China	643	18	30	595	95	1	2	Western Pacific	0.010526	0.021053
337	2020-01-23	Singapore	1	0	0	1	1	0	0	Western Pacific	0.000000	0.000000
354	2020-01-23	Thailand	3	0	0	3	1	0	0	South-East Asia	0.000000	0.000000
368	2020-01-23	Vietnam	2	0	0	2	2	0	0	Western Pacific	0.000000	0.000000
410	2020-01-24	China	920	26	36	858	277	8	6	Western Pacific	0.028881	0.021661



```
1 top_death_rate_days = df_nonzero.sort_values(by='Death_Rate', ascending=False).head(10)
2 top_death_rate_days.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region	Death_Rate	Recovery_Rate
16330	2020-04-18	France	148086	19325	35983	92778	2	642	1563	Europe	321.000000	781.500000
18761	2020-05-01	France	167305	24597	50212	92496	6	218	736	Europe	36.333333	122.666667
30471	2020-07-02	United Kingdom	285285	44080	1373	239832	6	89	1	Europe	14.833333	0.166667
21753	2020-05-17	France	179693	28111	61327	90255	63	579	765	Europe	9.190476	12.142857
24564	2020-06-01	Greece	2918	179	1374	1365	1	4	0	Europe	4.000000	0.000000



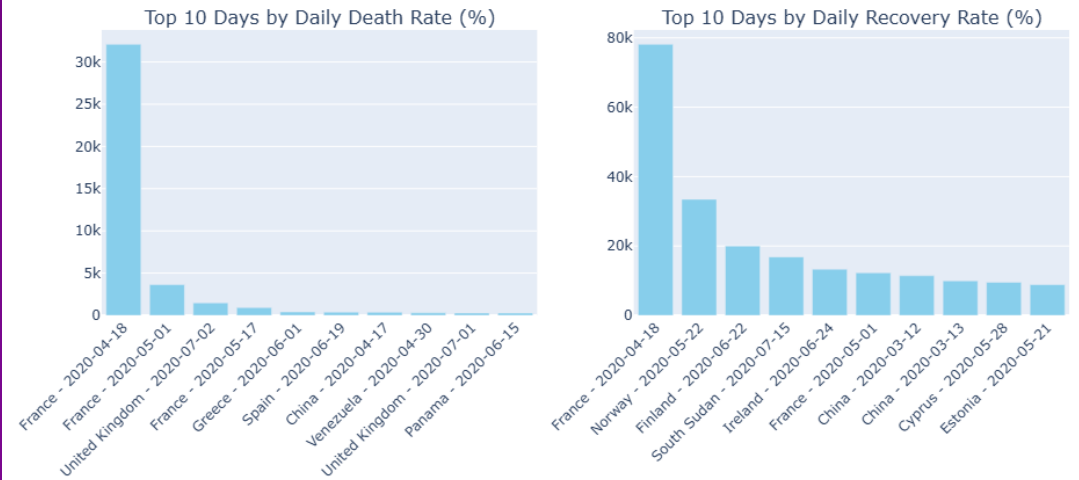
```
1 top_recovery_rate_days = df_nonzero.sort_values(by='Recovery_Rate', ascending=False).head(10)
2 top_recovery_rate_days.head()
```

	Date	Country/Region	Confirmed	Deaths	Recovered	Active	New cases	New deaths	New recovered	WHO Region	Death_Rate	Recovery_Rate
16330	2020-04-18	France	148086	19325	35983	92778	2	642	1563	Europe	321.0	781.500000
22753	2020-05-22	Norway	8332	235	7727	370	23	0	7695	Europe	0.0	334.565217
28484	2020-06-22	Finland	7144	327	6400	417	1	1	200	Europe	1.0	200.000000
32881	2020-07-15	South Sudan	2153	41	1175	937	5	0	842	Africa	0.0	168.400000
28881	2020-06-24	Ireland	25396	1726	23364	306	5	6	666	Europe	1.2	133.200000

FOR RATIOS (DEATHS/INFECTIONS, RECOVERIES/INFECTIONS)

```
1
2 top_death_rate_days['Label'] = top_death_rate_days['Country/Region'] + ' - ' + top_death_rate_days['Date'].astype(str)
3 top_recovery_rate_days['Label'] = top_recovery_rate_days['Country/Region'] + ' - ' + top_recovery_rate_days['Date'].astype(str)
4
5
6 fig = make_subplots(
7     rows=1, cols=2,
8     subplot_titles=[
9         'Top 10 Days by Daily Death Rate (%)',
10        'Top 10 Days by Daily Recovery Rate (%)'
11    ]
12 )
13
14 # -----Daily death rates-----
15 fig.add_trace(
16     go.Bar(
17         x=top_death_rate_days['Label'],
18         y=top_death_rate_days['Death_Rate'] * 100,
19         name='Death Rate',
20         marker_color='skyblue'
21     ),
22     row=1, col=1
23 )
24
25 # -----Daily recovery rates-----
26 fig.add_trace(
27     go.Bar(
28         x=top_recovery_rate_days['Label'],
29         y=top_recovery_rate_days['Recovery_Rate'] * 100,
30         name='Recovery Rate',
31         marker_color='skyblue'
32     ),
33     row=1, col=2
34 )
35
36 fig.update_layout(
37     title_text='Top 10 Days by Daily Death and Recovery Rates (Relative to New Cases)',
38     showlegend=False,
39     height=500,width=1000,
40     xaxis_tickangle=-45,
41     xaxis2_tickangle=-45
42 )
43
44 fig.show()
```

Top 10 Days by Daily Death and Recovery Rates (Relative to New Cases)



THE DEVELOPMENT OF INFECTIONS IN EGYPT



```
1 country = 'Egypt'
2 df_country = covid_19[covid_19['Country/Region'] == country]
3
4 df_melted = df_country.melt(
5     id_vars=['Date'],
6     value_vars=['Confirmed', 'Deaths', 'Recovered'],
7     var_name='Status',
8     value_name='Count'
9 )
10
11
12 fig = px.line(
13     df_melted,
14     x='Date',
15     y='Count',
16     color='Status',
17     title=f'COVID-19 Trends Over Time in {country}'
18 )
19
20 fig.update_layout(
21     xaxis_title='Date',
22     yaxis_title='Number of Cases',
23     xaxis_tickangle=45
24 )
25
26 fig.show()
```

