COVID-19 Data Exploration

This code provides an in-depth exploration of COVID-19 data, covering various aspects of the pandemic across different countries and continents.

It utilizes SQL queries to extract and analyze data from two tables: CovidDeaths and CovidVaccinations.

In [1]:

```
import pandas as pd
import numpy as np
import sqlite3 as sql
```

In [2]:

```
#create connection to database file
database= 'Covid.db'
conn = sql.connect(database)
```

In [3]:

```
df1 = pd.read_csv(r"C:\Users\mani ganesh\Desktop\CovidDeaths.csv")
df1.to_sql('coviddeaths', conn, if_exists = 'replace', index= False)
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 85171 entries, 0 to 85170
Data columns (total 59 columns):
 #   Column                                 Non-Null Count  Dtype  
---  ------                                 --------------  -----  
 0   iso_code                               85171 non-null  object 
 1   continent                              81060 non-null  object 
 2   location                               85171 non-null  object 
 3   date                                   85171 non-null  object 
 4   total_cases                            83072 non-null  float64
 5   new_cases                              83070 non-null  float64
 6   new_cases_smoothed                     82069 non-null  float64
 7   total_deaths                           73408 non-null  float64
 8   new_deaths                             73566 non-null  float64
 9   new_deaths_smoothed                    82069 non-null  float64
 10  total_cases_per_million                82623 non-null  float64
 11  new_cases_per_million                  82621 non-null  float64
 12  new_cases_smoothed_per_million         81625 non-null  float64
 13  total_deaths_per_million               72972 non-null  float64
 14  new_deaths_per_million                 73130 non-null  float64
 15  new_deaths_smoothed_per_million        81625 non-null  float64
 16  reproduction_rate                      68942 non-null  float64
 17  icu_patients                           8684 non-null   float64
 18  icu_patients_per_million               8684 non-null   float64
 19  hosp_patients                          10814 non-null  float64
 20  hosp_patients_per_million              10814 non-null  float64
 21  weekly_icu_admissions                  789 non-null    float64
 22  weekly_icu_admissions_per_million      789 non-null    float64
 23  weekly_hosp_admissions                 1295 non-null   float64
 24  weekly_hosp_admissions_per_million     1295 non-null   float64
 25  new_tests                              38945 non-null  float64
 26  total_tests                            38652 non-null  float64
 27  total_tests_per_thousand               38652 non-null  float64
 28  new_tests_per_thousand                 38945 non-null  float64
 29  new_tests_smoothed                     44625 non-null  float64
 30  new_tests_smoothed_per_thousand        44625 non-null  float64
 31  positive_rate                          42904 non-null  float64
 32  tests_per_case                         42311 non-null  float64
 33  tests_units                            46079 non-null  object 
 34  total_vaccinations                     9374 non-null   float64
 35  people_vaccinated                      8744 non-null   float64
 36  people_fully_vaccinated                6431 non-null   float64
 37  new_vaccinations                       7954 non-null   float64
 38  new_vaccinations_smoothed              15092 non-null  float64
 39  total_vaccinations_per_hundred         9374 non-null   float64
 40  people_vaccinated_per_hundred          8744 non-null   float64
 41  people_fully_vaccinated_per_hundred    6431 non-null   float64
 42  new_vaccinations_smoothed_per_million  15092 non-null  float64
 43  stringency_index                       72207 non-null  float64
 44  population                             84622 non-null  float64
 45  population_density                     79274 non-null  float64
 46  median_age                             76706 non-null  float64
 47  aged_65_older                          75830 non-null  float64
 48  aged_70_older                          76276 non-null  float64
 49  gdp_per_capita                         77046 non-null  float64
 50  extreme_poverty                        52449 non-null  float64
 51  cardiovasc_death_rate                  77634 non-null  float64
 52  diabetes_prevalence                    78779 non-null  float64
 53  female_smokers                         60828 non-null  float64
 54  male_smokers                           59931 non-null  float64
 55  handwashing_facilities                 39007 non-null  float64
 56  hospital_beds_per_thousand             70847 non-null  float64
 57  life_expectancy                        80833 non-null  float64
 58  human_development_index                77517 non-null  float64
dtypes: float64(54), object(5)
memory usage: 38.3+ MB
```

In [4]:
```
df1['date'] = pd.to_datetime(df1['date'])
print(df1['date'].dtype)
```

```
datetime64[ns]
```

In [5]:
```
df2 = pd.read_csv(r"C:\Users\mani ganesh\Desktop\CovidVaccinations.csv")
df2.to_sql('covidvaccination', conn, if_exists = 'replace', index= False)
df2.info()
```

```
 19  people_vaccinated_per_hundred          8744 non-null   float64
 20  people_fully_vaccinated_per_hundred    6431 non-null   float64
 21  new_vaccinations_smoothed_per_million  15092 non-null  float64
 22  stringency_index                       72207 non-null  float64
 23  population_density                     79274 non-null  float64
 24  median_age                             76706 non-null  float64
 25  aged_65_older                          75830 non-null  float64
 26  aged_70_older                          76276 non-null  float64
 27  gdp_per_capita                         77046 non-null  float64
 28  extreme_poverty                        52449 non-null  float64
 29  cardiovasc_death_rate                  77634 non-null  float64
 30  diabetes_prevalence                    78779 non-null  float64
 31  female_smokers                         60828 non-null  float64
 32  male_smokers                           59931 non-null  float64
 33  handwashing_facilities                 39007 non-null  float64
 34  hospital_beds_per_thousand             70847 non-null  float64
 35  life_expectancy                        80833 non-null  float64
 36  human_development_index                77517 non-null  float64
dtypes: float64(32), object(5)
memory usage: 24.0+ MB
```

In [6]:
```
df2['date'] = pd.to_datetime(df2['date'])
print(df2['date'].dtype)
```

```
datetime64[ns]
```

In [7]:
```
query = 'SELECT * FROM coviddeaths'
df = pd.read_sql_query(query,conn)
df.head()
```

Out[7]:

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed
0	AFG	Asia	Afghanistan	2/24/2020	1.0	1.0	NaN
1	AFG	Asia	Afghanistan	2/25/2020	1.0	0.0	NaN
2	AFG	Asia	Afghanistan	2/26/2020	1.0	0.0	NaN
3	AFG	Asia	Afghanistan	2/27/2020	1.0	0.0	NaN
4	AFG	Asia	Afghanistan	2/28/2020	1.0	0.0	NaN

5 rows × 59 columns

In [8]:
```
query = 'SELECT * FROM covidvaccination'
df = pd.read_sql_query(query,conn)
df.head()
```

Out[8]:

	iso_code	continent	location	date	new_tests	total_tests	total_tests_per_thousan
0	AFG	Asia	Afghanistan	2/24/2020	NaN	NaN	Na
1	AFG	Asia	Afghanistan	2/25/2020	NaN	NaN	Na
2	AFG	Asia	Afghanistan	2/26/2020	NaN	NaN	Na
3	AFG	Asia	Afghanistan	2/27/2020	NaN	NaN	Na
4	AFG	Asia	Afghanistan	2/28/2020	NaN	NaN	Na

5 rows × 37 columns

In [9]:
```
# Create a function query for reading the SQL query
def query(query):
    df = pd.read_sql_query(query, conn)
    display(df)
```

In [10]:
```
query( '''
SELECT *
FROM coviddeaths
order by 3,4
''')
```

0	AFG	Asia	Afghanistan	1/1/2021	51526.0	0.0
1	AFG	Asia	Afghanistan	1/10/2021	53489.0	89.0
2	AFG	Asia	Afghanistan	1/11/2021	53538.0	49.0
3	AFG	Asia	Afghanistan	1/12/2021	53584.0	46.0
4	AFG	Asia	Afghanistan	1/13/2021	53584.0	0.0
...	...	...	...	...	...	...
85166	ZWE	Africa	Zimbabwe	9/5/2020	6837.0	0.0
85167	ZWE	Africa	Zimbabwe	9/6/2020	6837.0	0.0
85168	ZWE	Africa	Zimbabwe	9/7/2020	7298.0	461.0
85169	ZWE	Africa	Zimbabwe	9/8/2020	7388.0	90.0
85170	ZWE	Africa	Zimbabwe	9/9/2020	7429.0	41.0

85171 rows × 59 columns

In [11]:
```
df1.columns
```

Out[11]:
```
Index(['iso_code', 'continent', 'location', 'date', 'total_cases', 'new_ca
ses',
       'new_cases_smoothed', 'total_deaths', 'new_deaths',
       'new_deaths_smoothed', 'total_cases_per_million',
       'new_cases_per_million', 'new_cases_smoothed_per_million',
       'total_deaths_per_million', 'new_deaths_per_million',
       'new_deaths_smoothed_per_million', 'reproduction_rate', 'icu_patien
ts',
       'icu_patients_per_million', 'hosp_patients',
       'hosp_patients_per_million', 'weekly_icu_admissions',
       'weekly_icu_admissions_per_million', 'weekly_hosp_admissions',
       'weekly_hosp_admissions_per_million', 'new_tests', 'total_tests',
       'total_tests_per_thousand', 'new_tests_per_thousand',
       'new_tests_smoothed', 'new_tests_smoothed_per_thousand',
       'positive_rate', 'tests_per_case', 'tests_units', 'total_vaccinatio
ns',
       'people_vaccinated', 'people_fully_vaccinated', 'new_vaccinations',
       'new_vaccinations_smoothed', 'total_vaccinations_per_hundred',
       'people_vaccinated_per_hundred', 'people_fully_vaccinated_per_hundr
ed',
       'new_vaccinations_smoothed_per_million', 'stringency_index',
       'population', 'population_density', 'median_age', 'aged_65_older',
       'aged_70_older', 'gdp_per_capita', 'extreme_poverty',
       'cardiovasc_death_rate', 'diabetes_prevalence', 'female_smokers',
       'male_smokers', 'handwashing_facilities', 'hospital_beds_per_thousa
nd',
       'life_expectancy', 'human_development_index'],
      dtype='object')
```

In [12]:
```
query('''
SELECT location, date, total_cases, new_cases, total_deaths,population
from coviddeaths
''')
```

	location	date	total_cases	new_cases	total_deaths	population
0	Afghanistan	2/24/2020	1.0	1.0	NaN	38928341.0
1	Afghanistan	2/25/2020	1.0	0.0	NaN	38928341.0
2	Afghanistan	2/26/2020	1.0	0.0	NaN	38928341.0
3	Afghanistan	2/27/2020	1.0	0.0	NaN	38928341.0
4	Afghanistan	2/28/2020	1.0	0.0	NaN	38928341.0
...	...	...	...	...	...	...
85166	Zimbabwe	4/26/2021	38102.0	16.0	1560.0	14862927.0
85167	Zimbabwe	4/27/2021	38164.0	62.0	1565.0	14862927.0
85168	Zimbabwe	4/28/2021	38191.0	27.0	1565.0	14862927.0
85169	Zimbabwe	4/29/2021	38235.0	44.0	1567.0	14862927.0
85170	Zimbabwe	4/30/2021	38257.0	22.0	1567.0	14862927.0

In [13]:

```
query('''
SELECT location, date, total_cases, total_deaths, (total_deaths/total_cases
from coviddeaths
order by 1,2
''')
```

	location	date	total_cases	total_deaths	DeathPercentage
0	Afghanistan	1/1/2021	51526.0	2191.0	4.252222
1	Afghanistan	1/10/2021	53489.0	2277.0	4.256950
2	Afghanistan	1/11/2021	53538.0	2288.0	4.273600
3	Afghanistan	1/12/2021	53584.0	2301.0	4.294192
4	Afghanistan	1/13/2021	53584.0	2301.0	4.294192
...	...	...	...	...	...
85166	Zimbabwe	9/5/2020	6837.0	206.0	3.013017
85167	Zimbabwe	9/6/2020	6837.0	206.0	3.013017
85168	Zimbabwe	9/7/2020	7298.0	210.0	2.877501
85169	Zimbabwe	9/8/2020	7388.0	218.0	2.950731
85170	Zimbabwe	9/9/2020	7429.0	222.0	2.988289

In [14]:

```
#create an death percentage for the location India
query('''
SELECT location, date, total_cases, total_deaths, (total_deaths/total_cases
from coviddeaths
WHERE location = 'India'
order by 1,2
''')
#WHERE location like %'india'%
```

	location	date	total_cases	total_deaths	DeathPercentage
0	India	1/1/2021	10286709.0	148994.0	1.448413
1	India	1/10/2021	10466595.0	151160.0	1.444214
2	India	1/11/2021	10479179.0	151327.0	1.444073
3	India	1/12/2021	10495147.0	151529.0	1.443801
4	India	1/13/2021	10512093.0	151727.0	1.443357
...	...	...	...	...	...
452	India	9/5/2020	4113811.0	70626.0	1.716802
453	India	9/6/2020	4204613.0	71642.0	1.703890
454	India	9/7/2020	4280422.0	72775.0	1.700183
455	India	9/8/2020	4370128.0	73890.0	1.690797
456	India	9/9/2020	4465863.0	75062.0	1.680795

457 rows × 5 columns

In [15]:

```
# Create an highest death percentage for every location
query('''
SELECT location, total_cases,total_deaths, population, MAX((total_deaths /
FROM coviddeaths
GROUP BY location;
''')
```

	location	total_cases	total_deaths	population	highestDeathPercentage
0	Afghanistan	56779.0	2512.0	3.892834e+07	4.424171
1	Africa	15265.0	834.0	1.340598e+09	5.463479
2	Albania	12.0	1.0	2.877800e+06	8.333333
3	Algeria	1983.0	313.0	4.385104e+07	15.784165
4	Andorra	761.0	51.0	7.726500e+04	6.701708
...	...	...	...	...	...
214	Vietnam	1046.0	35.0	9.733858e+07	3.346080
215	World	3196355.0	231576.0	7.794799e+09	7.245003
216	Yemen	6.0	2.0	2.982597e+07	33.333333
217	Zambia	40.0	2.0	1.838396e+07	5.000000
218	Zimbabwe	3.0	1.0	1.486293e+07	33.333333

219 rows × 5 columns

In [16]:

```
query('''
SELECT *, (total_deaths/total_cases)*100 as DeathPercentage
FROM coviddeaths
WHERE location = 'India'
AND total_cases = (SELECT MAX(total_cases)
                    FROM coviddeaths
                    WHERE location = 'India');
''')
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	tc
0	IND	Asia	India	4/30/2021	19164969.0	401993.0	364926.857	

1 rows × 60 columns

In [17]:

```
query('''
SELECT location, date, total_cases, population, (total_cases/population)*10
from coviddeaths
WHERE location = 'India'
order by 1,2
''')
```

```
     location        date   total_cases    population  totalcasePercentage
  0     India    1/1/2021    10286709.0  1.380004e+09             0.745411
  1     India   1/10/2021    10466595.0  1.380004e+09             0.758447
  2     India   1/11/2021    10479179.0  1.380004e+09             0.759358
  3     India   1/12/2021    10495147.0  1.380004e+09             0.760515
  4     India   1/13/2021    10512093.0  1.380004e+09             0.761743
...       ...         ...           ...           ...                  ...
452     India    9/5/2020     4113811.0  1.380004e+09             0.298101
453     India    9/6/2020     4204613.0  1.380004e+09             0.304681
454     India    9/7/2020     4280422.0  1.380004e+09             0.310175
455     India    9/8/2020     4370128.0  1.380004e+09             0.316675
456     India    9/9/2020     4465863.0  1.380004e+09             0.323612
```

In [18]:

```
#select the max cases in a day for India
query('''
SELECT location, date, total_cases, population, (total_cases / population)
FROM coviddeaths
WHERE location = 'India'
AND total_cases = (SELECT MAX(total_cases)
                    FROM coviddeaths
                    WHERE location = 'India');
''')
```

```
   location        date   total_cases    population  totalCasePercentage
0     India   4/30/2021    19164969.0  1.380004e+09             1.388761
```

In [19]:

```
#Select the highestTotalCasePercentage for every location
query('''
SELECT location, total_cases, population, MAX((total_cases / population) *
FROM coviddeaths
GROUP BY location
ORDER BY 1
''')
```

```
        location  total_cases    population  highestTotalCasePercentage
  0  Afghanistan      59745.0  3.892834e+07                    0.153474
  1       Africa    4557699.0  1.340598e+09                    0.339975
  2      Albania     131085.0  2.877800e+06                    4.555042
  3      Algeria     122108.0  4.385104e+07                    0.278461
  4      Andorra      13232.0  7.726500e+04                   17.125477
...          ...          ...           ...                         ...
214      Vietnam       2928.0  9.733858e+07                    0.003008
215        World  151399480.0  7.794799e+09                    1.942314
216        Yemen       6317.0  2.982597e+07                    0.021180
217       Zambia      91586.0  1.838396e+07                    0.498184
218     Zimbabwe      38257.0  1.486293e+07                    0.257399

219 rows × 4 columns
```

In [20]:

```
# Countries with Highest Death Count per Population
query('''
SELECT location, MAX(Total_deaths) as TotalDeathCount
FROM coviddeaths
Where continent is not null
GROUP BY location
ORDER BY 2 desc
''')
```

```
           location  TotalDeathCount
  0   United States         576232.0
  1          Brazil         403781.0
  2          Mexico         216907.0
  3           India         211853.0
  4  United Kingdom         127775.0
...             ...              ...
205         Curacao              NaN
206  Cayman Islands              NaN
207         Bermuda              NaN
208           Aruba              NaN
209        Anguilla              NaN

210 rows × 2 columns
```

In [21]:

```
# contintents with the highest death count per population
query('''
SELECT location, MAX(Total_deaths) as TotalDeathCount,population
FROM coviddeaths
Where continent is null
GROUP BY location
ORDER BY 2 desc
''')
```

	location	TotalDeathCount	population
0	World	3180238.0	7.794799e+09
1	Europe	1016750.0	7.486801e+08
2	North America	847942.0	5.920722e+08
3	European Union	688896.0	4.449191e+08
4	South America	672415.0	4.307598e+08
5	Asia	520286.0	4.639847e+09
6	Africa	121784.0	1.340598e+09
7	Oceania	1046.0	4.267781e+07
8	International	15.0	NaN

In [22]:

```
# continent with the highest death count per population
query('''
SELECT location, total_deaths, total_cases, population, (max(total_deaths)/
FROM coviddeaths
Where continent is null
GROUP BY location
ORDER BY 2 desc
''')
```

	location	total_deaths	total_cases	population	TotalDeathCount
0	World	3180238.0	151399480.0	7.794799e+09	2.100561
1	Europe	1016750.0	44863478.0	7.486801e+08	2.266320
2	North America	847942.0	37529488.0	5.920722e+08	2.259402
3	European Union	688896.0	30771214.0	4.449191e+08	2.238768
4	South America	672415.0	24878216.0	4.307598e+08	2.702826
5	Asia	520286.0	39526308.0	4.639847e+09	1.316303
6	Africa	121784.0	4557699.0	1.340598e+09	2.672050
7	Oceania	1046.0	43444.0	4.267781e+07	2.407697
8	International	15.0	721.0	NaN	2.080444

In [23]:
```
# death per day globaly
query('''
SELECT date, sum(new_cases) as total_cases, sum(new_deaths) as total_deaths
((sum(new_deaths)/sum(new_cases))*100) as DeathPercentage
FROM coviddeaths
Where continent is not null
GROUP BY date
ORDER BY 1
''')
```

```
           date  total_cases  total_deaths  DeathPercentage
  0    1/1/2020          NaN           NaN              NaN
  1    1/1/2021     533018.0        9662.0         1.812697
  2   1/10/2020          NaN           NaN              NaN
  3   1/10/2021     589134.0        8347.0         1.416825
  4   1/11/2020          NaN           NaN              NaN
...         ...          ...           ...              ...
481    9/5/2020     269991.0        4940.0         1.829691
482    9/6/2020     230465.0        3845.0         1.668366
483    9/7/2020     217882.0        9373.0         4.301870
484    9/8/2020     242199.0        4914.0         2.028910
485    9/9/2020     284822.0        6105.0         2.143444
```

In [24]:
```
# total number of deaths per cases in global
query('''
SELECT  sum(new_cases) as total_cases, sum(new_deaths) as total_deaths,
((sum(new_deaths)/sum(new_cases))*100) as DeathPercentage
FROM coviddeaths
ORDER BY 1,2
''')
```

```
   total_cases  total_deaths  DeathPercentage
0  482497587.0    10229544.0         2.120123
```

In [25]:

```
# join the table
query("""
SELECT *
FROM coviddeaths cd
JOIN covidvaccination cv
on cd.location = cv.location
and cd.date = cv.date
""")
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoo
0	AFG	Asia	Afghanistan	2/24/2020	1.0	1.0	
1	AFG	Asia	Afghanistan	2/25/2020	1.0	0.0	
2	AFG	Asia	Afghanistan	2/26/2020	1.0	0.0	
3	AFG	Asia	Afghanistan	2/27/2020	1.0	0.0	
4	AFG	Asia	Afghanistan	2/28/2020	1.0	0.0	
...	...	...	...	...	...	...	
85166	ZWE	Africa	Zimbabwe	4/26/2021	38102.0	16.0	34
85167	ZWE	Africa	Zimbabwe	4/27/2021	38164.0	62.0	41
85168	ZWE	Africa	Zimbabwe	4/28/2021	38191.0	27.0	30
85169	ZWE	Africa	Zimbabwe	4/29/2021	38235.0	44.0	31
85170	ZWE	Africa	Zimbabwe	4/30/2021	38257.0	22.0	30

85171 rows × 96 columns

```
In [26]: # total population vs vaccination
         query("""
         SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinatio
         FROM coviddeaths cd
         JOIN covidvaccination cv
         ON cd.location = cv.location
         AND cd.date = cv.date
         WHERE cd.continent is not null
         ORDER BY 1,2,3
         """)
```

```
             continent   location       date   population  new_vaccinations
    0           Africa    Algeria   1/1/2021   43851043.0               NaN
    1           Africa    Algeria  1/10/2021   43851043.0               NaN
    2           Africa    Algeria  1/11/2021   43851043.0               NaN
    3           Africa    Algeria  1/12/2021   43851043.0               NaN
    4           Africa    Algeria  1/13/2021   43851043.0               NaN
  ...              ...        ...        ...          ...               ...
81055  South America  Venezuela   9/5/2020   28435943.0               NaN
81056  South America  Venezuela   9/6/2020   28435943.0               NaN
81057  South America  Venezuela   9/7/2020   28435943.0               NaN
81058  South America  Venezuela   9/8/2020   28435943.0               NaN
81059  South America  Venezuela   9/9/2020   28435943.0               NaN

81060 rows × 5 columns
```

In [27]:

```
# total population vs vaccination
query("""
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinatio
,sum(cv.new_vaccinations) OVER (PARTITION BY cd.location ORDER BY cd.locati
cd.date) as RollingPeopleVaccinated
FROM coviddeaths cd
JOIN covidvaccination cv
ON cd.location = cv.location
AND cd.date = cv.date
WHERE cd.continent is not null
ORDER BY 1,2,3
""")
```

```
        continent    location       date   population  new_vaccinations  RollingPeopleVaccin
    0      Africa     Algeria   1/1/2021   43851043.0               NaN
    1      Africa     Algeria  1/10/2021   43851043.0               NaN
    2      Africa     Algeria  1/11/2021   43851043.0               NaN
    3      Africa     Algeria  1/12/2021   43851043.0               NaN
    4      Africa     Algeria  1/13/2021   43851043.0               NaN
  ...         ...         ...        ...          ...               ...
81055       South   Venezuela   9/5/2020   28435943.0               NaN
          America
81056       South   Venezuela   9/6/2020   28435943.0               NaN
          America
81057       South   Venezuela   9/7/2020   28435943.0               NaN
          America
81058       South   Venezuela   9/8/2020   28435943.0               NaN
```

In [28]:
```
#Using CTE to perform Calculation on Partition By in previous query
query('''
With PopvsVac (Continent, Location, Date, Population, New_Vaccinations, Rol
(
Select cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinatio
, SUM(cv.new_vaccinations) OVER (Partition by cd.Location Order by cd.locat
From CovidDeaths cd
Join covidvaccination cv
On cd.location = cv.location
and cd.date = cv.date
where cd.continent is not null
)
Select *, (RollingPeopleVaccinated/Population)*100
From PopvsVac
''')
```

	Continent	Location	Date	Population	New_Vaccinations	RollingPeopleVacc
0	Asia	Afghanistan	1/1/2021	38928341.0	NaN	
1	Asia	Afghanistan	1/10/2021	38928341.0	NaN	
2	Asia	Afghanistan	1/11/2021	38928341.0	NaN	
3	Asia	Afghanistan	1/12/2021	38928341.0	NaN	
4	Asia	Afghanistan	1/13/2021	38928341.0	NaN	
...	...	...	...	...	...	
81055	Africa	Zimbabwe	9/5/2020	14862927.0	NaN	49
81056	Africa	Zimbabwe	9/6/2020	14862927.0	NaN	49
81057	Africa	Zimbabwe	9/7/2020	14862927.0	NaN	49
81058	Africa	Zimbabwe	9/8/2020	14862927.0	NaN	49

Key Insights

Countries with Highest Infection Rate: The code identifies countries with the highest infection rates compared to their population. This analysis provides insights into the relative vulnerability of different regions to the virus.

Countries with Highest Death Count per Population: The code identifies countries with the highest death counts per population. This analysis highlights the disparities in the severity of the pandemic's impact across different regions.

Total Cases vs Total Deaths: The code examines the relationship between total cases and total deaths, providing an indication of the likelihood of succumbing to COVID-19 in different countries.

Total Cases vs Population: The code analyzes total cases relative to population size, showing the percentage of the population infected with COVID-19 in different regions.

Global Numbers: The code aggregates global COVID-19 data, providing an overview of the pandemic's overall impact.

Total Population vs Vaccinations: The code explores the relationship between total