MS SQL Server – Data Exploration:

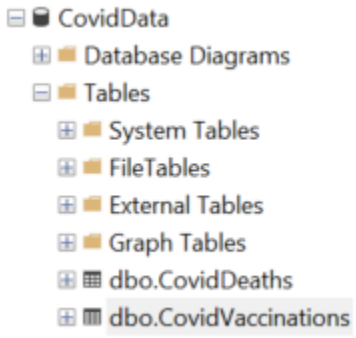
Background: I have downloaded a csv file from the '*Our World In Data'* website. This dataset contains information about COVID deaths and vaccinations from 2020 to 2024. This dataset consists of 365,565 rows and 26 columns for each Covid Deaths and Covid Vaccinations tables.

Objective: To perform data exploration of the two tables Covid Deaths and Covid Vaccinations and find out:

- The number of continents
- The possibility of dying if contracting COVID in USA
- The percentage of population infected with COVID in USA
- The countries with highest COVID infection per population
- The countries with highest death per population from COVID
- The continents with highest death per population from COVID
- The number of people in the world that has received COVID vaccines as the day goes by

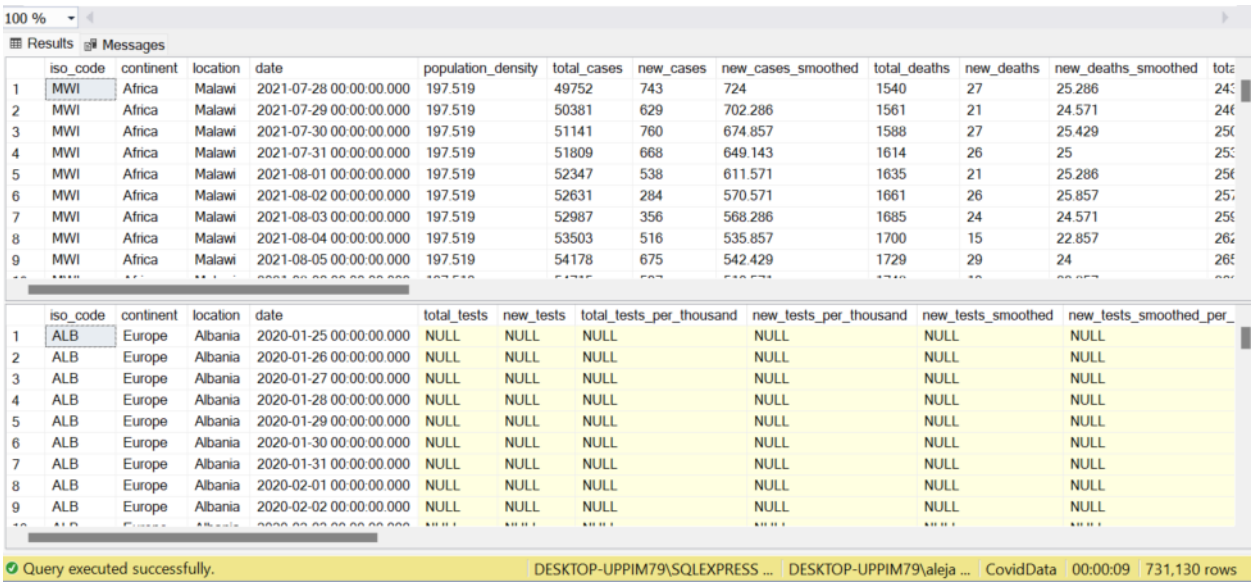
Steps Taken:

1. The first step in this project was to import each xlsx table for Covid Deaths and Covid Vaccinations in SQL Server and see if query was working:




```
SELECT *
FROM dbo.CovidDeaths


SELECT *
FROM dbo.CovidVaccinations
```



100 %

Results Messages

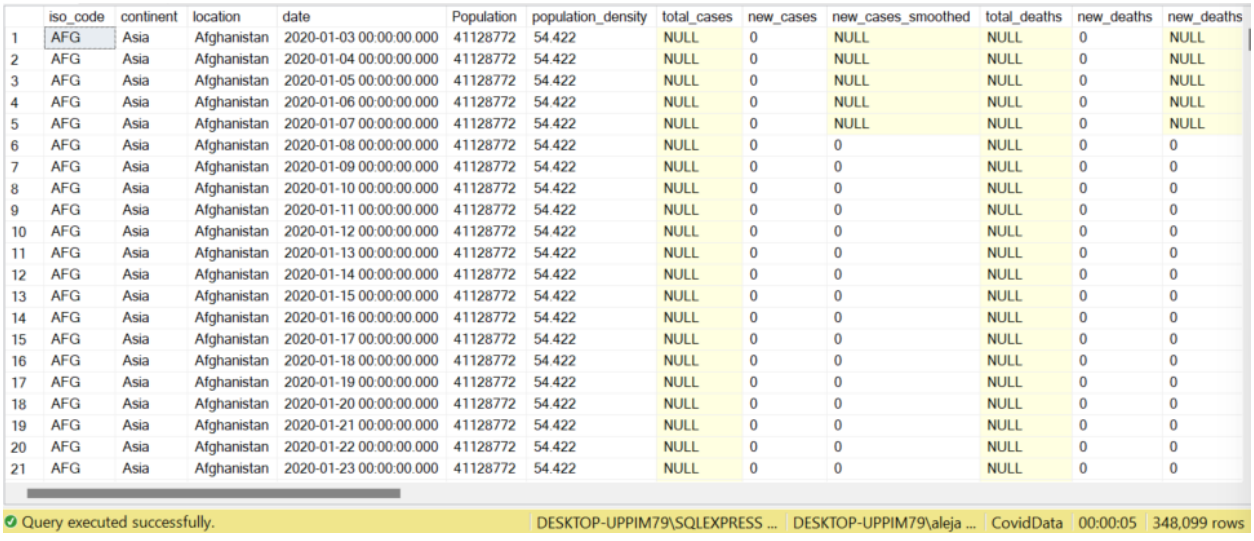
	iso_code	continent	location	date	population_density	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	tota
1	MWI	Africa	Malawi	2021-07-28 00:00:00.000	197.519	49752	743	724	1540	27	25.286	24
2	MWI	Africa	Malawi	2021-07-29 00:00:00.000	197.519	50381	629	702.286	1561	21	24.571	24
3	MWI	Africa	Malawi	2021-07-30 00:00:00.000	197.519	51141	760	674.857	1588	27	25.429	25
4	MWI	Africa	Malawi	2021-07-31 00:00:00.000	197.519	51809	668	649.143	1614	26	25	25
5	MWI	Africa	Malawi	2021-08-01 00:00:00.000	197.519	52347	538	611.571	1635	21	25.286	25
6	MWI	Africa	Malawi	2021-08-02 00:00:00.000	197.519	52631	284	570.571	1661	26	25.857	25
7	MWI	Africa	Malawi	2021-08-03 00:00:00.000	197.519	52987	356	568.286	1685	24	24.571	25
8	MWI	Africa	Malawi	2021-08-04 00:00:00.000	197.519	53503	516	535.857	1700	15	22.857	26
9	MWI	Africa	Malawi	2021-08-05 00:00:00.000	197.519	54178	675	542.429	1729	29	24	26

	iso_code	continent	location	date	total_tests	new_tests	total_tests_per_thousand	new_tests_per_thousand	new_tests_smoothed	new_tests_smoothed_per_
1	ALB	Europe	Albania	2020-01-25 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
2	ALB	Europe	Albania	2020-01-26 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
3	ALB	Europe	Albania	2020-01-27 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
4	ALB	Europe	Albania	2020-01-28 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
5	ALB	Europe	Albania	2020-01-29 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
6	ALB	Europe	Albania	2020-01-30 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
7	ALB	Europe	Albania	2020-01-31 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
8	ALB	Europe	Albania	2020-02-01 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL
9	ALB	Europe	Albania	2020-02-02 00:00:00.000	NULL	NULL	NULL	NULL	NULL	NULL

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:09 731,130 rows

2. To obtain the dataset from the COVID Deaths table where there are no null values in the continent field:

```
/*Number of continents*/
/*Use the IS NOT NULL constraint to not select null values; return only values where continent has values*/
SELECT *
FROM CovidDeaths
WHERE continent IS NOT NULL
ORDER BY location, date;
```



	iso_code	continent	location	date	Population	population_density	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths
1	AFG	Asia	Afghanistan	2020-01-03 00:00:00.000	41128772	54.422	NULL	0	NULL	NULL	0	NULL
2	AFG	Asia	Afghanistan	2020-01-04 00:00:00.000	41128772	54.422	NULL	0	NULL	NULL	0	NULL
3	AFG	Asia	Afghanistan	2020-01-05 00:00:00.000	41128772	54.422	NULL	0	NULL	NULL	0	NULL
4	AFG	Asia	Afghanistan	2020-01-06 00:00:00.000	41128772	54.422	NULL	0	NULL	NULL	0	NULL
5	AFG	Asia	Afghanistan	2020-01-07 00:00:00.000	41128772	54.422	NULL	0	NULL	NULL	0	NULL
6	AFG	Asia	Afghanistan	2020-01-08 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
7	AFG	Asia	Afghanistan	2020-01-09 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
8	AFG	Asia	Afghanistan	2020-01-10 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
9	AFG	Asia	Afghanistan	2020-01-11 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
10	AFG	Asia	Afghanistan	2020-01-12 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
11	AFG	Asia	Afghanistan	2020-01-13 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
12	AFG	Asia	Afghanistan	2020-01-14 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
13	AFG	Asia	Afghanistan	2020-01-15 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
14	AFG	Asia	Afghanistan	2020-01-16 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
15	AFG	Asia	Afghanistan	2020-01-17 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
16	AFG	Asia	Afghanistan	2020-01-18 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
17	AFG	Asia	Afghanistan	2020-01-19 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
18	AFG	Asia	Afghanistan	2020-01-20 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
19	AFG	Asia	Afghanistan	2020-01-21 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
20	AFG	Asia	Afghanistan	2020-01-22 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0
21	AFG	Asia	Afghanistan	2020-01-23 00:00:00.000	41128772	54.422	NULL	0	0	NULL	0	0

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:05 348,099 rows

3. To obtain a list of unique continents in the dataset:

```
/*Number of continents*/
/*To remove duplicates from query results, use the DISTINCT constraint:*/
SELECT DISTINCT continent
FROM CovidDeaths
WHERE continent IS NOT NULL;
```

	continent
1	North America
2	Asia
3	Africa
4	Oceania
5	South America
6	Europe

4. Next, to find the probability of dying if contracting COVID in the USA I used the CONVERT and NULLIF functions to convert values into float as the data type, and to return two values if there were either null values or any float numbers:

```
SELECT location,
       date,
       total_cases,
       total_deaths,
       (CONVERT(float, total_deaths) / NULLIF(CONVERT(float, total_cases), 0))*100 AS Death_Percentage
FROM CovidDeaths
WHERE location LIKE '%United States%' AND continent IS NOT NULL
ORDER BY 1, 2
```

	location	date	total_cases	total_deaths	Death_Percentage
1	United States	2020-01-03 00:00:00.000	NULL	NULL	NULL
2	United States	2020-01-04 00:00:00.000	NULL	NULL	NULL
3	United States	2020-01-05 00:00:00.000	NULL	NULL	NULL
4	United States	2020-01-06 00:00:00.000	NULL	NULL	NULL
5	United States	2020-01-07 00:00:00.000	NULL	NULL	NULL
6	United States	2020-01-08 00:00:00.000	NULL	NULL	NULL
7	United States	2020-01-09 00:00:00.000	NULL	NULL	NULL
8	United States	2020-01-10 00:00:00.000	NULL	NULL	NULL
9	United States	2020-01-11 00:00:00.000	NULL	NULL	NULL
10	United States	2020-01-12 00:00:00.000	NULL	NULL	NULL
11	United States	2020-01-13 00:00:00.000	NULL	NULL	NULL
12	United States	2020-01-14 00:00:00.000	NULL	NULL	NULL
13	United States	2020-01-15 00:00:00.000	NULL	NULL	NULL
14	United States	2020-01-16 00:00:00.000	NULL	NULL	NULL
15	United States	2020-01-17 00:00:00.000	NULL	NULL	NULL
16	United States	2020-01-18 00:00:00.000	NULL	NULL	NULL
17	United States	2020-01-19 00:00:00.000	NULL	NULL	NULL
18	United States	2020-01-20 00:00:00.000	1	NULL	NULL
19	United States	2020-01-21 00:00:00.000	1	NULL	NULL
20	United States	2020-01-22 00:00:00.000	1	NULL	NULL
21	United States	2020-01-23 00:00:00.000	1	NULL	NULL
22	United States	2020-01-24 00:00:00.000	1	NULL	NULL

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:00 2,896 rows

5. Next to find the percentage of population infected with COVID in the United States I divided the total cases by the population, and then multiplying that by 100. I also created an alias for the new column using the AS keyword:

```
SELECT location, date, population, total_cases, (total_cases/population)*100 AS Percentage_of_Population_Infected
FROM CovidDeaths
WHERE location LIKE '%UNITED STATES%'
ORDER BY 1, 2
```

	location	date	population	total_cases	Percentage_of_Population_Infected
1	United States	2020-01-03 00:00:00.000	1893	NULL	NULL
2	United States	2020-01-04 00:00:00.000	1893	NULL	NULL
3	United States	2020-01-05 00:00:00.000	1893	NULL	NULL
4	United States	2020-01-06 00:00:00.000	1893	NULL	NULL
5	United States	2020-01-07 00:00:00.000	1893	NULL	NULL
6	United States	2020-01-08 00:00:00.000	1893	NULL	NULL
7	United States	2020-01-09 00:00:00.000	1893	NULL	NULL
8	United States	2020-01-10 00:00:00.000	1893	NULL	NULL
9	United States	2020-01-11 00:00:00.000	1893	NULL	NULL
10	United States	2020-01-12 00:00:00.000	1893	NULL	NULL
11	United States	2020-01-13 00:00:00.000	1893	NULL	NULL
12	United States	2020-01-14 00:00:00.000	1893	NULL	NULL
13	United States	2020-01-15 00:00:00.000	1893	NULL	NULL
14	United States	2020-01-16 00:00:00.000	1893	NULL	NULL
15	United States	2020-01-17 00:00:00.000	1893	NULL	NULL
16	United States	2020-01-18 00:00:00.000	1893	NULL	NULL
17	United States	2020-01-19 00:00:00.000	1893	NULL	NULL
18	United States	2020-01-20 00:00:00.000	1893	1	0.0528262017960909
19	United States	2020-01-21 00:00:00.000	1893	1	0.0528262017960909
20	United States	2020-01-22 00:00:00.000	1893	1	0.0528262017960909
21	United States	2020-01-23 00:00:00.000	1893	1	0.0528262017960909
22	United States	2020-01-24 00:00:00.000	1893	1	0.0528262017960909

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:00 2,896 rows

6. To find the countries with the highest COVID infection per location and population, I looked for the maximum value of the total cases and renamed that field as Highest Infection Count. Likewise, to calculate the Percentage Population Infected I divided the maximum value of total cases by the maximum value of the population and multiplied that by 100:

```
SELECT location,
       population,
       MAX(total_cases) AS HighestInfectionCount,
       MAX((total_cases/population))*100 AS PercentPopulationInfected
FROM CovidDeaths
GROUP BY location, Population
ORDER by location, HighestInfectionCount, PercentPopulationInfected DESC
```

	location	population	HighestInfectionCount	PercentPopulationInfected
1	Afghanistan	41128772	996	0.558455769114624
2	Africa	41128772	98	0.000763455811420774
3	Africa	1426736614	9973075	0.919809576009101
4	Albania	2842318	9967	11.7541386994699
5	Albania	1426736614	998	7.03703816211168E-05
6	Algeria	2842318	9935	1.2016600535197
7	Algeria	44903228	99897	0.605769366959542
8	American Samoa	44903228	NULL	NULL
9	American Samoa	44295	84	18.8712044248787
10	Andorra	44295	997	18.1713511683034
11	Andorra	79843	9972	60.1367684080007
12	Angola	79843	9871	26.7775509437271
13	Angola	35588996	99839	0.298696259933829
14	Anguilla	15877	984	24.5890281539334
15	Anguilla	35588996	99	0.000306274445055994
16	Antigua and Barbuda	93772	9106	9.71078786844687
17	Antigua and Barbuda	15877	992	8.3643005605593
18	Argentina	93772	985992	5697.79998293734
19	Argentina	45510324	9963697	22.135937770955
20	Armenia	2780472	451272	16.2300501497587
21	Armenia	45510324	99563	0.75586585584405
22	Aruba	106459	44224	41.540874890803

7. Next, I identified the data type in each column name by typing ‘EXEC sp_help’, followed by the table for Covid Deaths:

```
EXEC sp_help CovidDeaths
```

	Name	Owner	Type	Created_datetime
1	CovidDeaths	dbo	user table	2024-03-14 21:51:30.213

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
1	iso_code	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
2	continent	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
3	location	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
4	date	datetime	no	8			yes	(n/a)	(n/a)	NULL
5	Population	float	no	8	53	NULL	yes	(n/a)	(n/a)	NULL
6	population_d...	float	no	8	53	NULL	yes	(n/a)	(n/a)	NULL
7	total_cases	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
8	new_cases	float	no	8	53	NULL	yes	(n/a)	(n/a)	NULL

	Identity	Seed	Increment	Not For Replication
1	No identity column defined.	NULL	NULL	NULL

8. Doing this I was able to find out that the ‘total deaths’ field contains a data type of ‘nvarchar’. Therefore, to get better results when calculating the maximum value of total deaths I used the CAST function to convert values as integer:

```
select location, MAX(cast(total_deaths as int)) as Total_Death_Count
from CovidDeaths
GROUP BY location
ORDER BY Total_Death_Count DESC

EXEC sp_help CovidDeaths
```

	location	Total_Death_Count
1	World	6988666
2	High income	2930636
3	Upper middle income	2665233
4	Europe	2087378
5	Asia	1635825
6	North America	1621332
7	South America	1354165
8	Lower middle income	1340822
9	European Union	1250472
10	United States	1144877
11	Brazil	702116
12	India	533316
13	Russia	400771
14	Mexico	334938
15	Africa	259057
16	United Kingdom	232112
17	Peru	221564
18	Italy	193419
19	Germany	174979
20	France	167985
21	Indonesia	161930
22	Iran	146741

Query executed successfully. | DESKTOP-UPPIM79\SQLEXPRESS ... | DESKTOP-UPPIM79\aleja ... | CovidData | 00:00:00 | 255 rows

9. Similarly, to find out the continents that have the highest death count with no null values:

```
select continent, MAX(cast(total_deaths as int)) as Total_Death_Count
from CovidDeaths
WHERE continent IS NOT NULL
GROUP BY continent
ORDER BY Total_Death_Count DESC
```

	continent	Total_Death_Count
1	North America	1144877
2	South America	702116
3	Asia	533316
4	Europe	400771
5	Africa	102595
6	Oceania	23915

10. Next, to find the total covid cases, total covid deaths and total death percentages in the world by date and continent, I used the CONVERT and NULLIF functions (to convert the datatype in float and to return 0 if a value was null) and finally to ignore the division in Null values:

```
SELECT date,
       continent,
       SUM(CAST(new_cases AS int)) AS total_new_cases,
       SUM(CAST(new_deaths AS int)) AS total_new_deaths,
       SUM(CONVERT(float, new_deaths)/NULLIF(CONVERT(float,new_cases),0))*100 AS total_new_death_percentage
FROM CovidDeaths
WHERE continent IS NOT NULL
GROUP BY date, continent
ORDER BY 1, 2
```

	date	continent	total_new_cases	total_new_deaths	total_new_death_percentage
1	2020-01-01 00:00:00.000	North America	NULL	NULL	NULL
2	2020-01-01 00:00:00.000	South America	NULL	NULL	NULL
3	2020-01-02 00:00:00.000	North America	NULL	NULL	NULL
4	2020-01-02 00:00:00.000	South America	NULL	NULL	NULL
5	2020-01-03 00:00:00.000	Africa	0	0	NULL
6	2020-01-03 00:00:00.000	Asia	0	0	NULL
7	2020-01-03 00:00:00.000	Europe	0	0	NULL
8	2020-01-03 00:00:00.000	North America	0	0	NULL
9	2020-01-03 00:00:00.000	Oceania	0	0	NULL
10	2020-01-03 00:00:00.000	South America	0	0	NULL
11	2020-01-04 00:00:00.000	Africa	0	0	NULL
12	2020-01-04 00:00:00.000	Asia	1	0	0
13	2020-01-04 00:00:00.000	Europe	2	0	0
14	2020-01-04 00:00:00.000	North America	0	0	NULL
15	2020-01-04 00:00:00.000	Oceania	0	0	NULL
16	2020-01-04 00:00:00.000	South America	0	0	NULL
17	2020-01-05 00:00:00.000	Africa	0	0	NULL
18	2020-01-05 00:00:00.000	Asia	0	0	NULL
19	2020-01-05 00:00:00.000	Europe	0	3	NULL
20	2020-01-05 00:00:00.000	North America	0	0	NULL
21	2020-01-05 00:00:00.000	Oceania	0	0	NULL
22	2020-01-05 00:00:00.000	South America	0	0	NULL

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:00 8,735 rows

11. Finally, to find out the number of people in the world that has received at least one Covid Vaccine as the day goes by (by country), I joined the Covid Deaths and Covid Vaccinations by location and date. I also used the Window functions OVER and PARTITION BY to define a specified set of rows and to divide them into partitions.

```
SELECT cd.continent,
       cd.location,
       cd.date,
       cd.Population,
       cv.new_vaccinations,
       SUM(CONVERT(BIGINT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER BY cd.location, cd.date) AS RollingPeopleVaccinated
FROM CovidDeaths AS cd

JOIN CovidVaccinations AS cv
     ON cd.location = cv.location
     AND cd.date = cv.date

WHERE cd.continent IS NOT NULL AND cv.new_vaccinations IS NOT NULL
ORDER BY 2, 3
```

	continent	location	date	Population	new_vaccinations	RollingPeopleVaccinated
1	Asia	Afghanistan	2021-05-27 00:00:00.000	41128772	2859	2859
2	Asia	Afghanistan	2021-06-03 00:00:00.000	41128772	4015	6874
3	Asia	Afghanistan	2022-01-27 00:00:00.000	41128772	6868	13742
4	Asia	Afghanistan	2022-04-27 00:00:00.000	41128772	383	14125
5	Asia	Afghanistan	2022-09-12 00:00:00.000	41128772	9447	23572
6	Asia	Afghanistan	2022-11-02 00:00:00.000	41128772	36587	60159
7	Asia	Afghanistan	2022-11-16 00:00:00.000	41128772	14800	74959
8	Asia	Afghanistan	2023-04-25 00:00:00.000	41128772	3316	78275
9	Europe	Albania	2021-01-13 00:00:00.000	2842318	60	60
10	Europe	Albania	2021-01-14 00:00:00.000	2842318	78	138
11	Europe	Albania	2021-01-15 00:00:00.000	2842318	42	180
12	Europe	Albania	2021-01-16 00:00:00.000	2842318	61	241
13	Europe	Albania	2021-01-17 00:00:00.000	2842318	36	277
14	Europe	Albania	2021-01-18 00:00:00.000	2842318	42	319
15	Europe	Albania	2021-01-19 00:00:00.000	2842318	36	355
16	Europe	Albania	2021-01-20 00:00:00.000	2842318	36	391
17	Europe	Albania	2021-01-21 00:00:00.000	2842318	30	421
18	Europe	Albania	2021-02-18 00:00:00.000	2842318	1348	1769
19	Europe	Albania	2021-02-19 00:00:00.000	2842318	1128	2897
20	Europe	Albania	2021-03-23 00:00:00.000	2842318	3461	6358
21	Europe	Albania	2021-03-24 00:00:00.000	2842318	2302	8660
22	Europe	Albania	2021-03-25 00:00:00.000	2842318	5356	14016

Query executed successfully. DESKTOP-UPPIM79\SQLEXPRESS ... DESKTOP-UPPIM79\aleja ... CovidData 00:00:01 53,778 rows

12. Before going to the next steps, I noticed a few errors (location included high income, upper middle income, lower middle income and low income) in the table. To fix this I copied the entire data from SQL Server (for Covid Deaths and Covid Vaccinations) into Excel, and deleted those rows that were not needed (as I was going to import these into SQL later).
13. I then dropped the tables in SQL Server:

```
USE CovidData
DROP TABLE
  CovidDeaths, CovidVaccinations
```

14. In order to create visualizations for the next results using Tableau, I copied the data in SQL Server into MS Excel for it to be imported (A shortcut I found out to copy query result was to press Ctrl + Shift + C), and saved them under different names (CovidData-Table1, CovidData-Table2, CovidData-Table3, CovidData-Table4):

    - Total cases, total deaths, and total death percentage:

```
SELECT SUM(new_cases) AS total_cases,
       SUM(CAST(new_deaths AS INT)) AS total_deaths,
       SUM(CAST(new_deaths AS INT))/SUM(new_cases)*100 AS death_percentage
FROM CovidDeaths
```

	total_cases	total_deaths	death_percentage
1	2503616392	22185349	0.886132119556757

	A	B	C	D
1	total_cases	total_deaths	death_percentage	
2	2503616392	22185349	0.88613212	

    - Location and total death count:

```
SELECT location, SUM(CAST(new_deaths AS INT)) AS total_deaths
FROM CovidDeaths
WHERE continent IS NULL AND location NOT IN ('World', 'European Union')
GROUP BY location
ORDER BY total_deaths DESC
```

	location	total_deaths
1	Europe	2087384
2	Asia	1635991
3	North America	1607152
4	South America	1357686
5	Africa	259064
6	Oceania	31010

	A	B
1	location	total_deaths
2	Europe	2087384
3	Asia	1635991
4	North America	1607152
5	South America	1357686
6	Africa	259064
7	Oceania	31010
8		

- Location, population, highest infection count, and percentage population infected (where I replaced the NULL values with zero in Excel for future use):

```
SELECT location,
       Population,
       MAX(total_cases) AS highest_infection_count,
       MAX(total_cases/population)*100 AS Percentage_of_Population_Infected
FROM CovidDeaths
GROUP BY location, Population
ORDER BY Percentage_of_Population_Infected DESC
```

	A	B	C	D
1	location	Population	highest_infection_count	Percentage_of_Population_Infected
2	United States	1893	9920253	1392718.806
3	United Kingdom	1893	9962069	1310754.464
4	France	3801	38997490	1025979.742
5	Oceania	1952	14628086	749389.6516
6	High income	63329	9991472	454827.2971
7	Rwanda	47	133208	283421.2766
8	Netherlands	4413	8623210	195404.7133
9	Asia	106459	99515909	141656.3766
10	United States	106867	99883410	96790.24301
11	France	53117	9850650	54859.352
12	Poland	18084	6590705	36444.95134
13	South Korea	107135	99839	32269.44789
14	Portugal	18084	998289	26693.55784
15	Oman	1952	99	20437.55123
16	Europe	1326064	99658696	18923.872
17	Saudi Arabia	5401	841469	15579.8741
18	European Union	1201680	99984699	15383.70473
19	Germany	306292	99971	12549.38294
20	Saint Barthelemy	47	994	11717.02128
21	Serbia	31816	2583470	8120.033945
22	Argentina	93772	985992	5697.799983
23	Slovakia	33690	99304	5542.143069
24	Saudi Arabia	10994	98869	4948.189922
25	Venezuela	11335	99835	4625.41685
26	Lower middle income	2305826	9955791	4226.519954
27	Bulgaria	31332	993255	3720.818971
28	Upper middle income	1531043	9914005	3552.697801
29	Croatia	17032	9861	3522.645608
30	High income	13859349	427084445	3081.562092
31	Finland	53117	1499712	2823.412467
32	French Guiana	3801	9968	2466.061563
33	Czechia	191173	4713739	2465.692854
34	European Union	1326064	995954	2426.742601
35	Guatemala	56494	997980	2213.16246
36	Singapore	103959	996914	2183.586799
37	Nigeria	12691	267173	2105.216295

- Location, population, date, highest infection count, and percentage population infected:

```
SELECT location,
       Population,
       date,
       MAX(total_cases) AS highest_infection_count,
       MAX(total_cases/population)*100 AS Percentage_of_Population_Infected
FROM CovidDeaths
GROUP BY location, Population, date
ORDER BY Percentage_of_Population_Infected DESC
```

	location	Population	date	highest_infection_count	Percentage_of_Population_Infected
1	United States	1893	2021-02-04 00:00:00.000	26364167	1392718.80612784
2	United States	1893	2021-02-03 00:00:00.000	26242837	1386309.40306392
3	United States	1893	2021-02-02 00:00:00.000	26113794	1379492.55150555
4	United States	1893	2021-02-01 00:00:00.000	26001856	1373579.2921289
5	United States	1893	2021-01-31 00:00:00.000	25863033	1366245.80031696
6	United States	1893	2021-01-30 00:00:00.000	25698795	1357569.73058637
7	United States	1893	2021-01-29 00:00:00.000	25541973	1349285.4199683
8	United States	1893	2021-01-28 00:00:00.000	25383008	1340887.90279979
9	United States	1893	2021-01-27 00:00:00.000	25237065	1333178.28843106
10	United States	1893	2021-01-26 00:00:00.000	25101474	1326015.53090333
11	United States	1893	2021-01-25 00:00:00.000	24959920	1318537.77073428
12	United Kingdom	1893	2023-11-16 00:00:00.000	24812582	1310754.46381405
13	United Kingdom	1893	2023-11-17 00:00:00.000	24812582	1310754.46381405
14	United Kingdom	1893	2023-11-18 00:00:00.000	24812582	1310754.46381405
15	United Kingdom	1893	2023-11-19 00:00:00.000	24812582	1310754.46381405
16	United Kingdom	1893	2023-11-20 00:00:00.000	24812582	1310754.46381405
17	United Kingdom	1893	2023-11-21 00:00:00.000	24812582	1310754.46381405
18	United Kingdom	1893	2023-11-22 00:00:00.000	24812582	1310754.46381405
19	United Kingdom	1893	2023-11-23 00:00:00.000	24812582	1310754.46381405
20	United Kingdom	1893	2023-11-24 00:00:00.000	24812582	1310754.46381405
21	United Kingdom	1893	2023-11-25 00:00:00.000	24812582	1310754.46381405
22	United Kingdom	1893	2023-11-26 00:00:00.000	24812582	1310754.46381405

location	Population	date	highest_infection_count	Percentage_of_Population_Infected
United States	1893	2/4/2021	26364167	1392718.806
United States	1893	2/3/2021	26242837	1386309.403
United States	1893	2/2/2021	26113794	1379492.552
United States	1893	2/1/2021	26001856	1373579.292
United States	1893	1/31/2021	25863033	1366245.8
United States	1893	1/30/2021	25698795	1357569.731
United States	1893	1/29/2021	25541973	1349285.42
United States	1893	1/28/2021	25383008	1340887.903
United States	1893	1/27/2021	25237065	1333178.288
United States	1893	1/26/2021	25101474	1326015.531
United States	1893	1/25/2021	24959920	1318537.771
United Kingdom	1893	11/16/2023	24812582	1310754.464
United Kingdom	1893	11/17/2023	24812582	1310754.464
United Kingdom	1893	11/18/2023	24812582	1310754.464
United Kingdom	1893	11/19/2023	24812582	1310754.464
United Kingdom	1893	11/20/2023	24812582	1310754.464
United Kingdom	1893	11/21/2023	24812582	1310754.464
United Kingdom	1893	11/22/2023	24812582	1310754.464
United Kingdom	1893	11/23/2023	24812582	1310754.464
United Kingdom	1893	11/24/2023	24812582	1310754.464
United Kingdom	1893	11/25/2023	24812582	1310754.464
United Kingdom	1893	11/26/2023	24812582	1310754.464
United Kingdom	1893	11/27/2023	24812582	1310754.464
United Kingdom	1893	11/28/2023	24812582	1310754.464
United Kingdom	1893	11/29/2023	24812582	1310754.464
United Kingdom	1893	11/30/2023	24812582	1310754.464
United Kingdom	1893	12/1/2023	24812582	1310754.464
United Kingdom	1893	12/2/2023	24812582	1310754.464
United Kingdom	1893	12/3/2023	24812582	1310754.464
United Kingdom	1893	12/4/2023	24812582	1310754.464
United Kingdom	1893	12/5/2023	24812582	1310754.464
United Kingdom	1893	12/6/2023	24812582	1310754.464
United Kingdom	1893	12/7/2023	24812582	1310754.464
United Kingdom	1893	12/8/2023	24812582	1310754.464
United Kingdom	1893	12/9/2023	24812582	1310754.464
United Kingdom	1893	12/10/2023	24812582	1310754.464

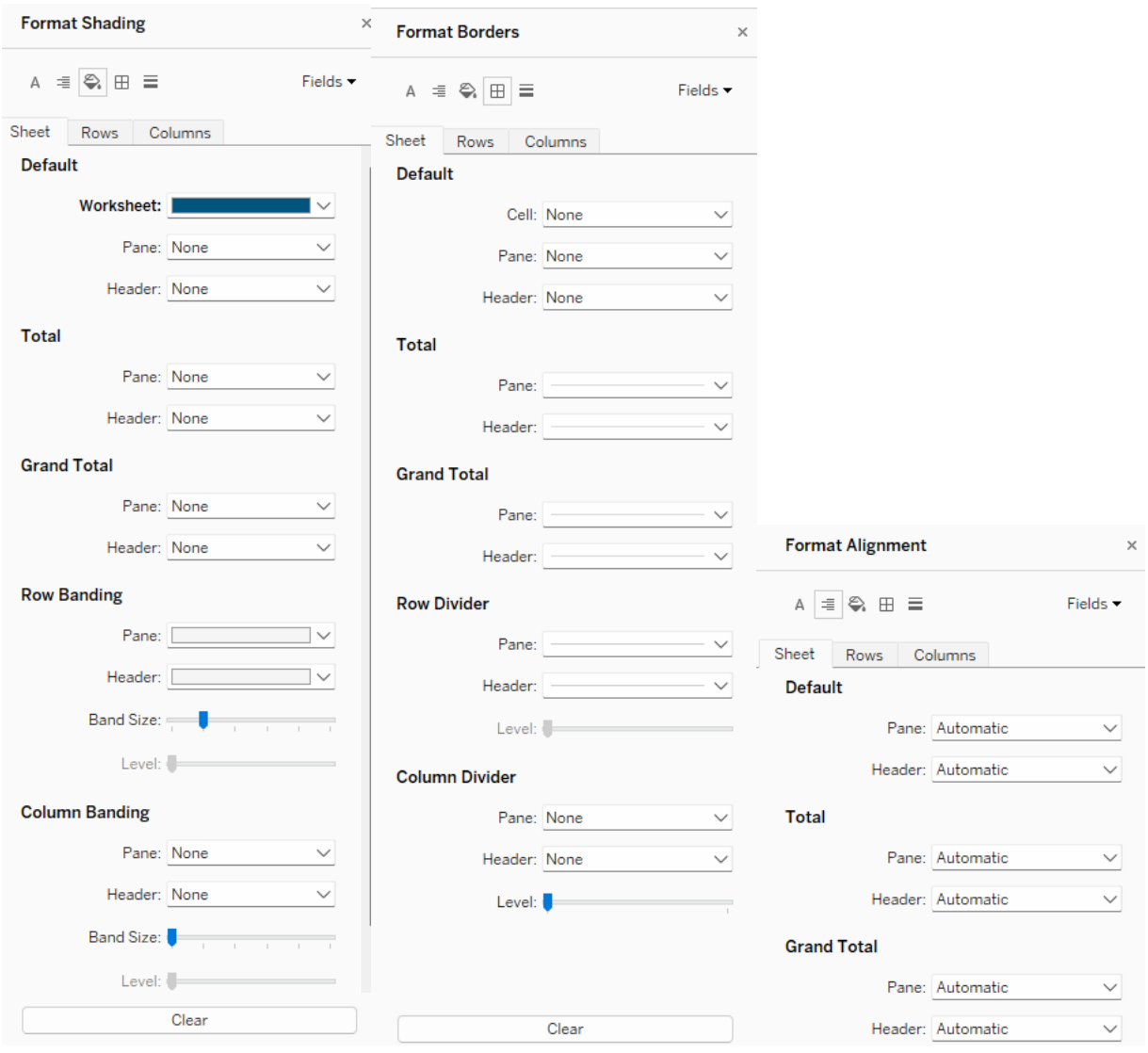
15. Next I opened Tableau and imported the sheets saved in Excel, in four sheets.

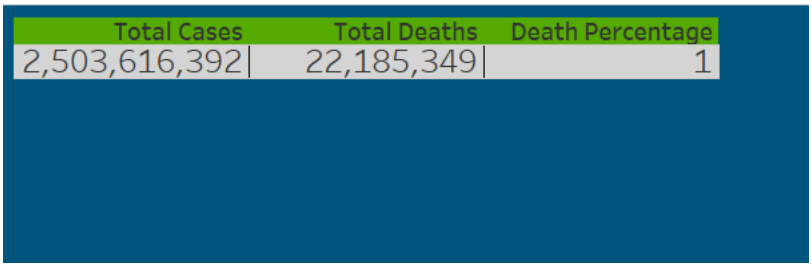
16. In the first sheet, I dragged table1 (the total cases, total deaths and total death percentage) into the columns section, and changed the chart to text tables:

Sheet 1

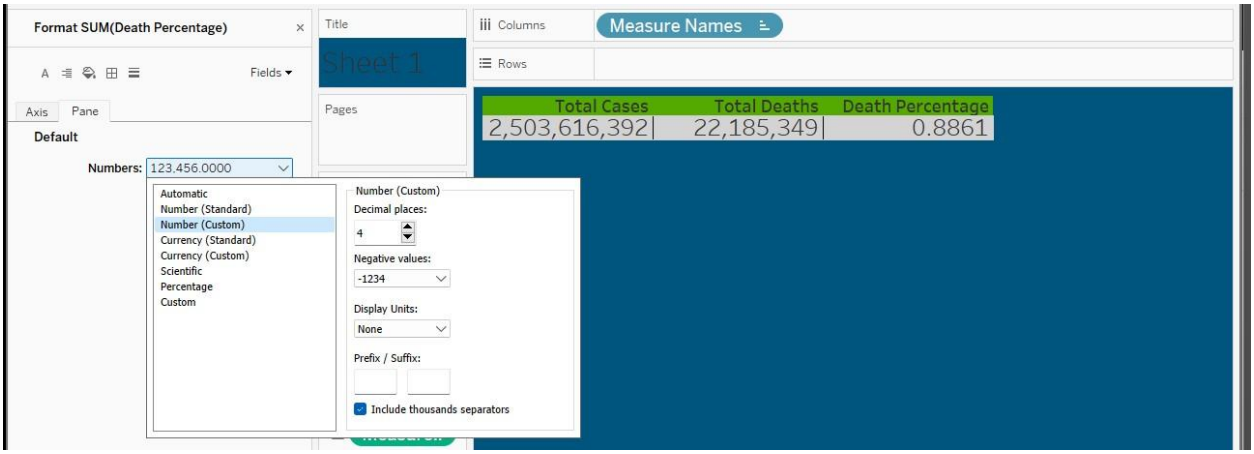
Death Percentage	1
Total Cases	2,503,616,392
Total Deaths	22,185,349

17. I dragged the Measure Names from the Rows section into Columns, and organized it to my liking (changing the color and formatting) By selecting the Format →(Shading, Borders, Alignment, etc):

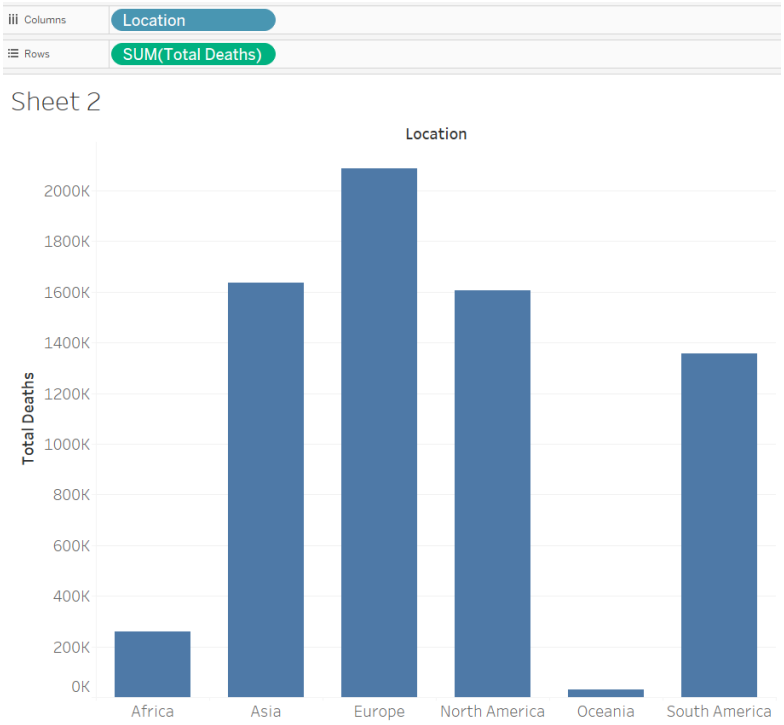




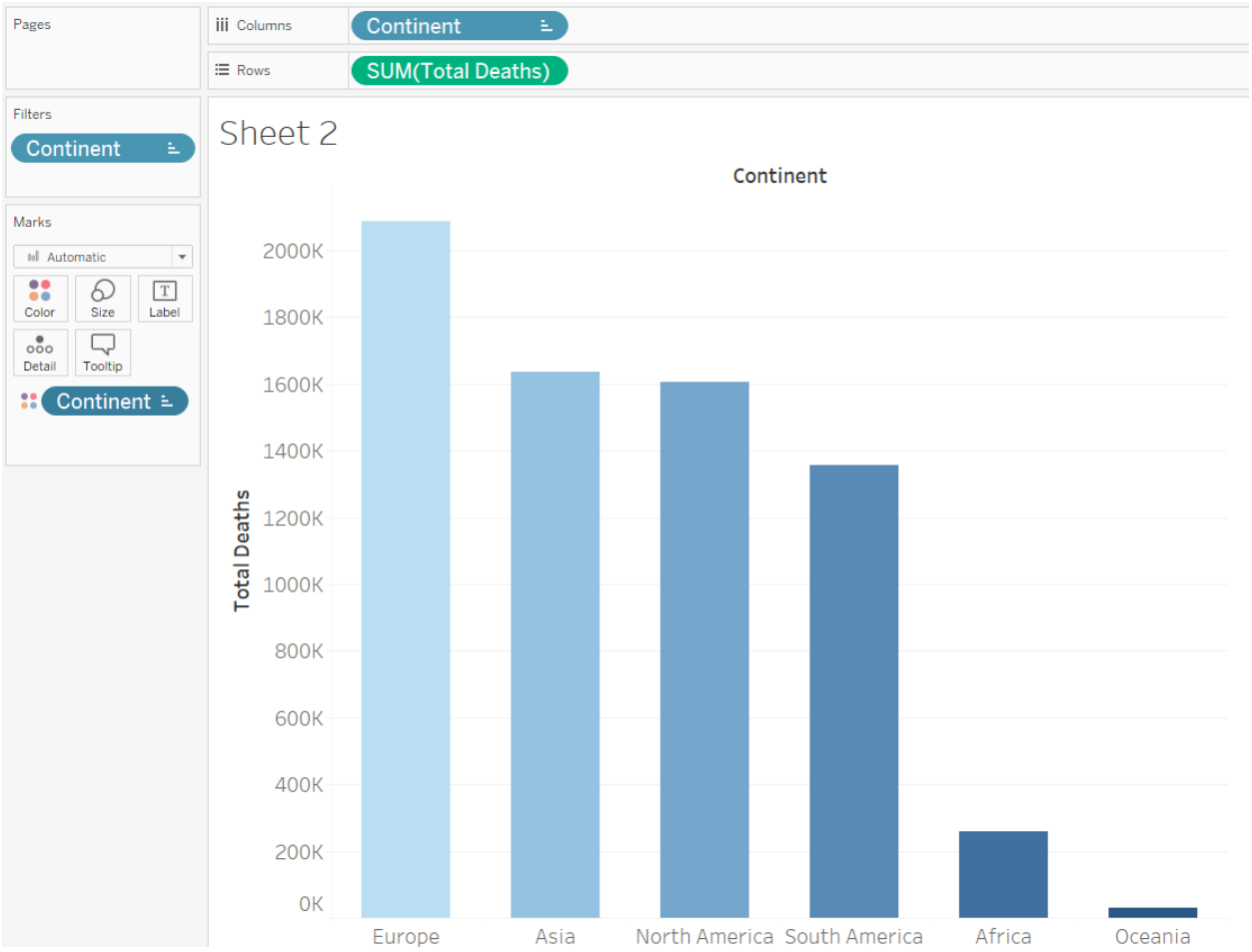
18. To change the death percentage to the nearest decimal, in the 'Marks' Box select Measure →Format. Then select the field you want to update (SUM(Death Percentage)):



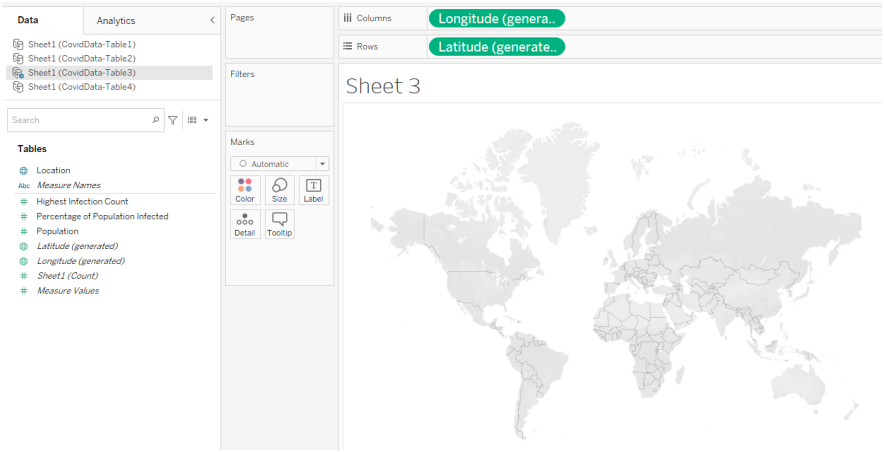
19. In the second sheet I dragged the location (from table 2) (into the column canvas) and the total deaths into the Rows Canvas, to create a bar chart:



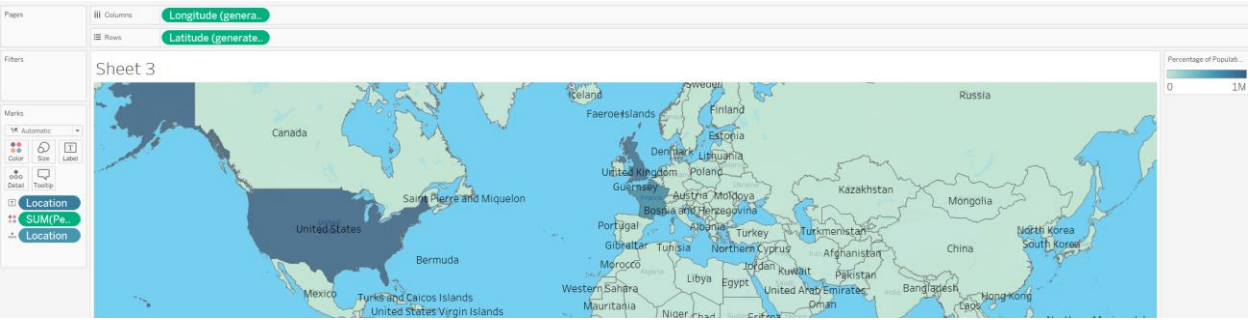
20. I then sorted the bars to my choice by clicking Location →Sort → By Manual (& then dragging locations to my preferred order). I also changed the color bars to my choice by first dragging continent (I changed the name from locations) into the marks table:



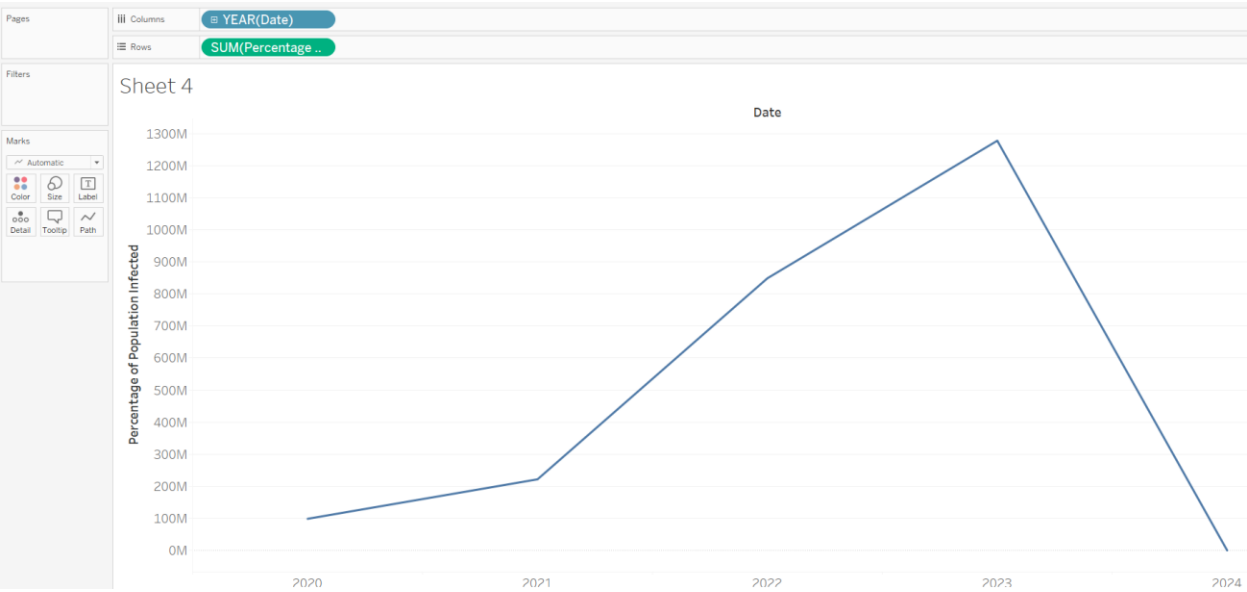
21. In the third sheet, to create a symbol map for location (in table 3) I clicked it's dropdown →Geographic Role →Country/Region (This in turn generates new 'latitude' and 'longitude' tables). I then dragged those new tables in the column and rows canvas:



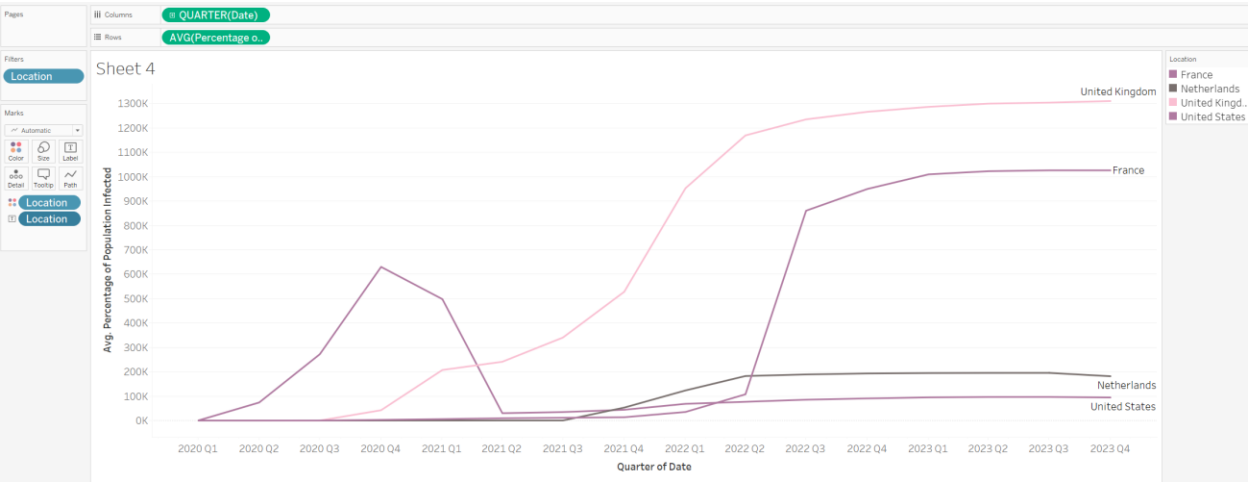
22. Next I dragged the ‘Location’ and ‘Percent Population Infected’ tables in the Marks box and changed the color by selecting the color option right next to Sum(percent population infected). I also updated the background of my choice by selecting Map → Background Maps → (Outdoors, Normal, Dark, etc). Including a label also helped in identifying the countries:



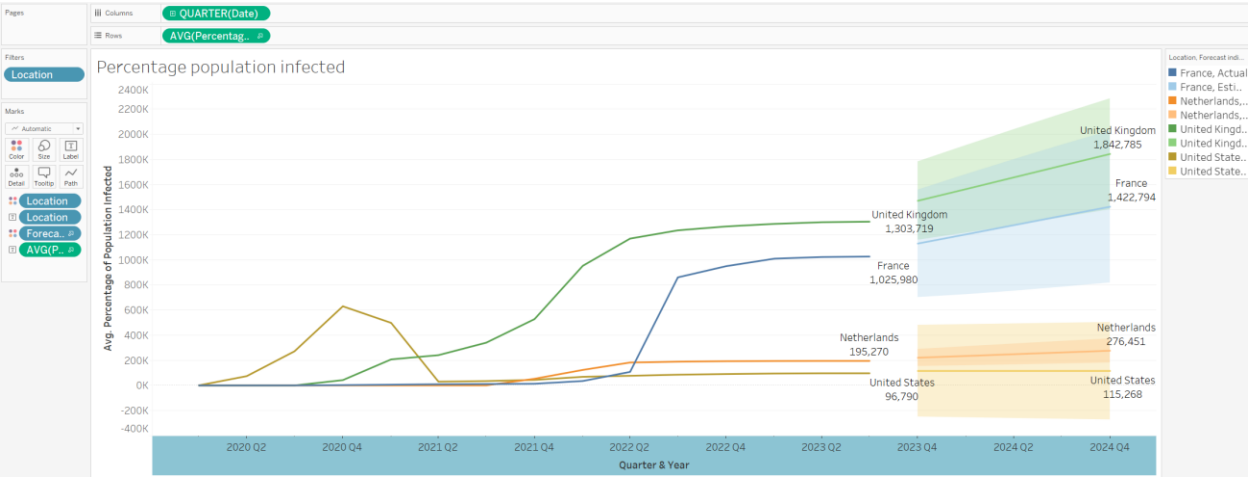
23. Finally, in sheet 4, I have used table 4 to do some time series and dragged the date table in columns and percentage of population infected into rows. The result being a trendline:



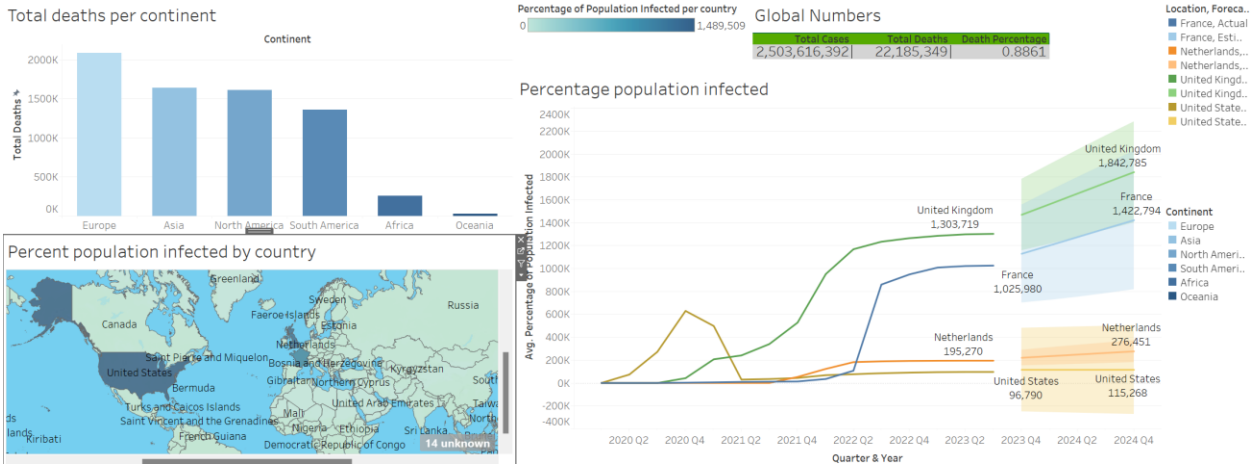
24. After dragging the location box into the Marks box to filter data by color and label (for UK, France, USA and Netherlands) to find the average percentage population infected:



25. To add some predictive analysis and forecasting, I selected Analysis → Forecast → Show forecast. The end result (plus by adding additional labels and renaming the axis):



26. Finally, I created a dashboard (by selecting new dashboard (rather than new worksheet or story) and selected the size to automatic (to get the largest dashboard/image). I dragged the finished worksheets as desired and renamed their titles as desired:



Conclusion: By performing an exploratory data analysis (DEA) I was able to find out:

- The dataset from the COVID Deaths table where there are no null values in the continent field,
- A list of unique continents in the dataset,
- The probability of dying if contracting COVID in the United States,
- The percentage of population infected with COVID in the United States,
- The countries with the highest COVID infection per location and population,
- The data type in each column name,
- The maximum value of total deaths,
- The continents that have the highest death count with no null values,
- The total covid cases, total covid deaths and total death percentages in the world by date and continent,
- The number of people in the world that has received at least one Covid Vaccine as the day goes by (by country).

References:

- https://ourworldindata.org/covid-deaths
- https://www.youtube.com/watch?v=qfyynHBFOsM
- https://www.udemy.com/course/sql-power-bi-data-analyst-ms-sql-ssrs-ssas-power-bi/learn/lecture/32120828?start=15#content
- https://public.tableau.com/app/profile/alejandro.castaneda2167/viz/CovidDatasetExamplefromSQLServer1tutorial/Dashboard1