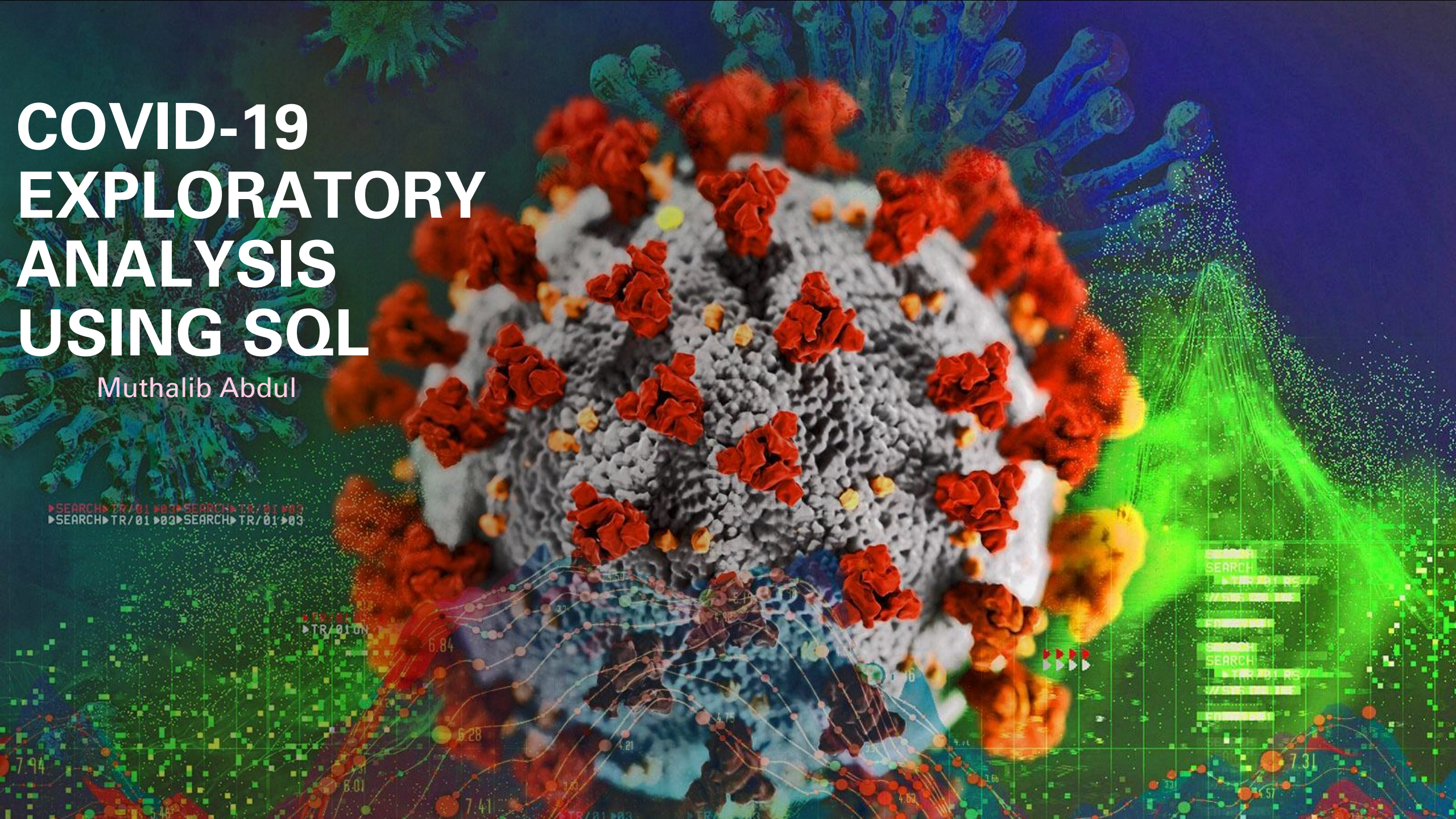


COVID-19 EXPLORATORY ANALYSIS USING SQL

Muthalib Abdul

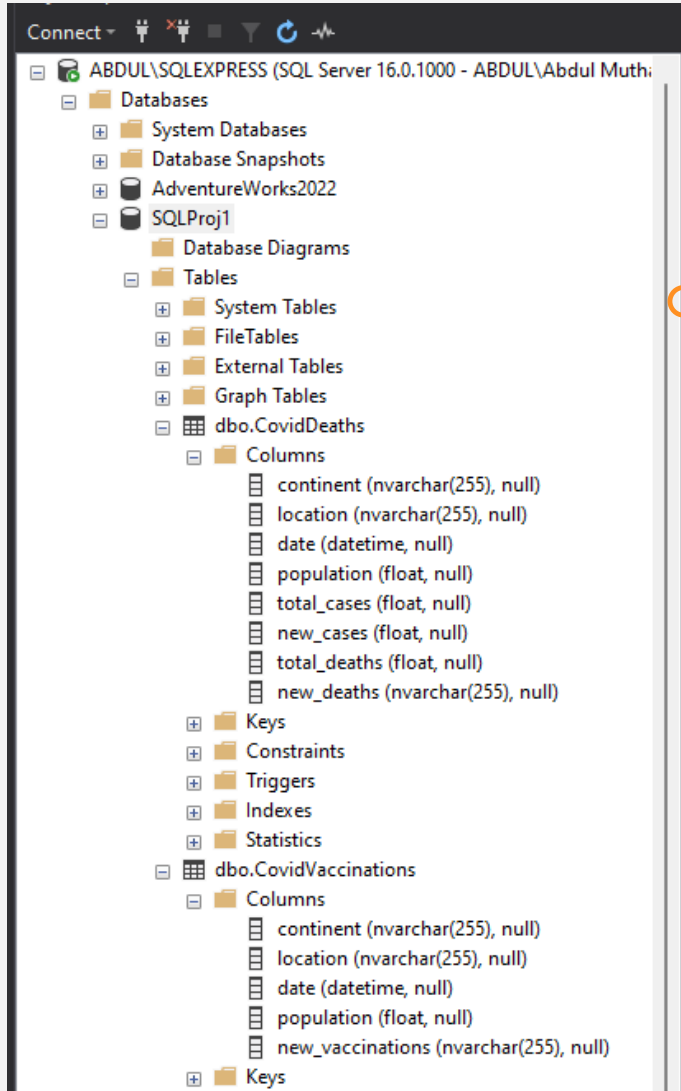
►SEARCH►TR/01►03►SEARCH►TR/01►03
►SEARCH►TR/01►03►SEARCH►TR/01►03

SEARCH
SEARCH
►TR/01►RS/7
W/SYS-000-000
SEARCH
SEARCH
►TR/01►RS/7
W/SYS-000-000
SEARCH



INTRODUCTION

- This project involves the use Covid-19 data and performing some basic data cleaning, data manipulation, and lastly data analysis to interpret the outcome
- Dataset used: <https://github.com/owid/covid-19-data/tree/master/public/data>
- Steps:
 - Download Covid Deaths & Vaccinations data
 - Performing data cleaning in excel (removing columns)
 - Importing data into MS SQL server
 - Performing data manipulation and data analysis
 - Interpreting Results
- This presentation will showcase the code with the output in each slide
- If you are interested in viewing just the syntax, [Click Here](#)
- Link to this project repository (GitHub) - [Click Here](#)
- Link to main portfolio [Click Here](#)
- Questions :
 1. Which country has the highest infection count/percentage
 2. Which country has the highest death count/percentage
 3. Produce a rolling vaccination count for each location and date



DATABASE OVERVIEW

- Here we have the database name "SQLProj1" that has 2 datasets
 - CovidDeaths
 - CovidVaccinations
- Each dataset has set number of columns that we will use to do our analysis
- We can also see the type of data and if null values are allowed within the columns

DATA EXPLORATION

```
-- Exploring both CovidDeaths & CovidVaccinations tables by ordering them by Location & Date
select * from SQLProj1..CovidDeaths
order by 2,3;

select * from SQLProj1..CovidVaccinations
order by 2,3;
```

- We use the simple select statement to select all variables from both the tables and then order the table by 2(location) and 3(date).
- Initially, null values are expected for all countries as we had no means to confirm a positive case

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

	continent	location	date	population	new_vaccinations
1	Asia	Afghanistan	2020-01-03 00:00:00.000	41128772	NULL
2	Asia	Afghanistan	2020-01-04 00:00:00.000	41128772	NULL
3	Asia	Afghanistan	2020-01-05 00:00:00.000	41128772	NULL
4	Asia	Afghanistan	2020-01-06 00:00:00.000	41128772	NULL
5	Asia	Afghanistan	2020-01-07 00:00:00.000	41128772	NULL
6	Asia	Afghanistan	2020-01-08 00:00:00.000	41128772	NULL
7	Asia	Afghanistan	2020-01-09 00:00:00.000	41128772	NULL
8	Asia	Afghanistan	2020-01-10 00:00:00.000	41128772	NULL
9	Asia	Afghanistan	2020-01-11 00:00:00.000	41128772	NULL
10	Asia	Afghanistan	2020-01-12 00:00:00.000	41128772	NULL
11	Asia	Afghanistan	2020-01-13 00:00:00.000	41128772	NULL
12	Asia	Afghanistan	2020-01-14 00:00:00.000	41128772	NULL
13	Asia	Afghanistan	2020-01-15 00:00:00.000	41128772	NULL
14	Asia	Afghanistan	2020-01-16 00:00:00.000	41128772	NULL

DATA MANIPULATION

```
-- Calculating the percentage of deaths vs cases
select location, date, total_cases, total_deaths, (total_deaths/total_cases)*100 as Daily_Mortality_Percentage
from SQLProj1..CovidDeaths
where continent is not null
order by 1,2
```

- Here we analyze the percentage of people that succumb to the disease vs total case.
- The new column Daily_Mortality_Percentage is a calculated field

	location	date	total_cases	total_deaths	Daily_Mortality_Percentage
76	Afghanistan	2020-03-18 00:00:00.000	24	NULL	NULL
77	Afghanistan	2020-03-19 00:00:00.000	24	NULL	NULL
78	Afghanistan	2020-03-20 00:00:00.000	24	NULL	NULL
79	Afghanistan	2020-03-21 00:00:00.000	24	NULL	NULL
80	Afghanistan	2020-03-22 00:00:00.000	34	NULL	NULL
81	Afghanistan	2020-03-23 00:00:00.000	40	1	2.5
82	Afghanistan	2020-03-24 00:00:00.000	42	1	2.38095238095238
83	Afghanistan	2020-03-25 00:00:00.000	74	1	1.35135135135135
84	Afghanistan	2020-03-26 00:00:00.000	80	2	2.5
85	Afghanistan	2020-03-27 00:00:00.000	91	2	2.1978021978022
86	Afghanistan	2020-03-28 00:00:00.000	106	2	1.88679245283019
87	Afghanistan	2020-03-29 00:00:00.000	114	4	3.50877192982456

DATA MANIPULATION Cont...

```
-- -- Calculating the percentage of total deaths vs total cases in the United States
select location, date, total_deaths, total_cases, (total_deaths/total_cases)*100 as Daily_Mortality_Percentage
from SQLProj1..CovidDeaths
where location = 'United States' and continent is not null
order by 1,2
```

- In this part, we check the mortality percentage in the "United States" by adding a simple where clause.
- We can observe initially we had null values and gradually number started to rise consistently as more and more people were diagnosed with the illness as well as succumbed.

	location	date	total_deaths	total_cases	Daily_Mortality_Percentage
51	United States	2020-02-22 00:00:00.000	NULL	35	NULL
52	United States	2020-02-23 00:00:00.000	NULL	40	NULL
53	United States	2020-02-24 00:00:00.000	NULL	48	NULL
54	United States	2020-02-25 00:00:00.000	NULL	48	NULL
55	United States	2020-02-26 00:00:00.000	NULL	52	NULL
56	United States	2020-02-27 00:00:00.000	NULL	56	NULL
57	United States	2020-02-28 00:00:00.000	NULL	64	NULL
58	United States	2020-02-29 00:00:00.000	1	69	1.44927536231884
59	United States	2020-03-01 00:00:00.000	1	73	1.36986301369863
60	United States	2020-03-02 00:00:00.000	2	82	2.4390243902439
61	United States	2020-03-03 00:00:00.000	2	100	2
62	United States	2020-03-04 00:00:00.000	8	135	5.92592592592593

DATA MANIPULATION Cont...

```
-- -- Calculating the percentage of total cases vs population overall
select location, date, total_cases, population, (total_cases/population)*100 as Daily_Infection_Rate
from SQLProj1..CovidDeaths
order by 1,2
```

- Next we calculate the infection rate in all "location" by using the total_cases and population and we see a similar trend of rising infectious rate

	location	date	total_cases	population	Daily_Infection_Rate
41	Afghanistan	2020-02-12 00:00:00.000	NULL	41128772	NULL
42	Afghanistan	2020-02-13 00:00:00.000	NULL	41128772	NULL
43	Afghanistan	2020-02-14 00:00:00.000	NULL	41128772	NULL
44	Afghanistan	2020-02-15 00:00:00.000	NULL	41128772	NULL
45	Afghanistan	2020-02-16 00:00:00.000	NULL	41128772	NULL
46	Afghanistan	2020-02-17 00:00:00.000	NULL	41128772	NULL
47	Afghanistan	2020-02-18 00:00:00.000	NULL	41128772	NULL
48	Afghanistan	2020-02-19 00:00:00.000	NULL	41128772	NULL
49	Afghanistan	2020-02-20 00:00:00.000	NULL	41128772	NULL
50	Afghanistan	2020-02-21 00:00:00.000	NULL	41128772	NULL
51	Afghanistan	2020-02-22 00:00:00.000	NULL	41128772	NULL
52	Afghanistan	2020-02-23 00:00:00.000	NULL	41128772	NULL
53	Afghanistan	2020-02-24 00:00:00.000	5	41128772	1.21569396723053E-05
54	Afghanistan	2020-02-25 00:00:00.000	5	41128772	1.21569396723053E-05
55	Afghanistan	2020-02-26 00:00:00.000	5	41128772	1.21569396723053E-05
56	Afghanistan	2020-02-27 00:00:00.000	5	41128772	1.21569396723053E-05

DAA MANIPULATION Cont...

```
-- -- Calculating the percentage of total cases vs population in the United States
select location, date, total_cases, population, (total_cases/population)*100 as Daily_Infection_Rate
from SQLProj1..CovidDeaths
where location = 'United States'
order by 1,2
```

- Next we calculate the infection rate in the "United States" by using the total_cases and population and we see a similar trend of rising infectious rate

	location	date	total_cases	population	Daily_Infection_Rate
11	United States	2020-01-13 00:00:00.000	NULL	338289856	NULL
12	United States	2020-01-14 00:00:00.000	NULL	338289856	NULL
13	United States	2020-01-15 00:00:00.000	NULL	338289856	NULL
14	United States	2020-01-16 00:00:00.000	NULL	338289856	NULL
15	United States	2020-01-17 00:00:00.000	NULL	338289856	NULL
16	United States	2020-01-18 00:00:00.000	NULL	338289856	NULL
17	United States	2020-01-19 00:00:00.000	NULL	338289856	NULL
18	United States	2020-01-20 00:00:00.000	1	338289856	2.95604488950446E-07
19	United States	2020-01-21 00:00:00.000	1	338289856	2.95604488950446E-07
20	United States	2020-01-22 00:00:00.000	1	338289856	2.95604488950446E-07
21	United States	2020-01-23 00:00:00.000	1	338289856	2.95604488950446E-07
22	United States	2020-01-24 00:00:00.000	1	338289856	2.95604488950446E-07
23	United States	2020-01-25 00:00:00.000	6	338289856	1.77362693370268E-06
24	United States	2020-01-26 00:00:00.000	7	338289856	2.06923142265312E-06

DATA MANIPULATION Cont...

```
-- Countries with highest infection rate
select location, max(total_cases) as Highest_Infection_Count, population, max(total_cases/population)*100 as Highest_Infection_Rate
from SQLProj1..CovidDeaths
where continent is not null
group by location, population
order by Daily_Infection_Rate desc
```

- Next, we use the max of total_cases and group by location & population to get the max infection rate for that location

	location	Highest_Infection_Count	population	Daily_Infection_Rate
1	Cyprus	655664	896007	73.1762140251136
2	San Marino	23873	33690	70.8607895517958
3	Austria	6046956	8939617	67.6422267307425
4	Faeroe Islands	34658	53117	65.2484138787959
5	Brunei	284632	449002	63.3921452465691
6	Slovenia	1342156	2119843	63.3139340979497
7	Gibraltar	20550	32677	62.8882700370291
8	Martinique	229479	367512	62.4412264089336
9	Andorra	47939	79843	60.0415816038977
10	Jersey	66391	110796	59.9218383335138
11	South Korea	30918060	51815808	59.6691650548034
12	Saint Pierre and Miquelon	3426	5885	58.2158028887001
13	Denmark	3409630	5882259	57.964635695232
14	Greece	5972760	10384972	57.5134916107622

DATA MANIPULATION Cont...

```
-- Countries with highest Death rate
select location, max(cast(total_deaths as int)) as Highest_Death_Count, max((total_deaths/population))*100 as Daily_Death_Rate
from SQLProj1..CovidDeaths
where continent is not null
group by location
order by Daily_Death_Rate desc
```

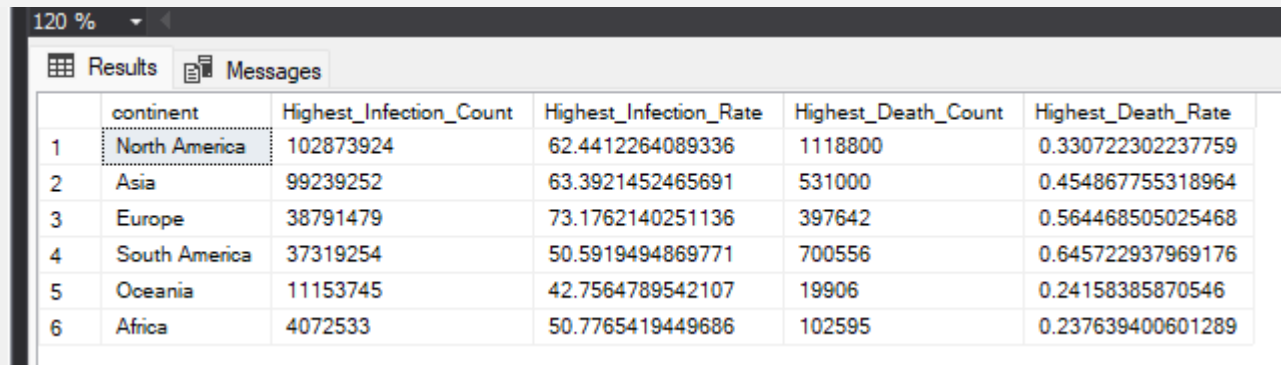
- Here we check the highest death count and death rate by every location.
- We also use the cast function to convert data type to int to make the calculation possible

Results		Messages	
	location	Highest_Death_Count	Daily_Death_Rate
1	Peru	219866	0.645722937969176
2	Bulgaria	38282	0.564468505025468
3	Bosnia and Herzegovina	16328	0.504958976722034
4	Hungary	48719	0.488788141708129
5	North Macedonia	9667	0.461739219318248
6	Georgia	17032	0.454867755318964
7	Croatia	18091	0.4488679798162
8	Montenegro	2808	0.447788327523354
9	Slovenia	9230	0.435409603447048
10	Czechia	42702	0.406918626756839
11	Slovakia	21123	0.374291989570219
12	Moldova	12086	0.369264462221581

FINAL ANALYSIS

```
-- Cases and Deaths by Continent/location
select continent,
       max(total_cases) as Highest_Infection_Count,
       max((total_cases/population))*100 as Highest_Infection_Rate,
       max(total_deaths) as Highest_Death_Count,
       max((total_deaths/population))*100 as Highest_Death_Rate
from SQLProj1..CovidDeaths
where continent is not null
group by continent
order by Highest_Infection_Count desc, Highest_Death_Count desc
```

- Here we take the max of total cases and its percentage, max of total deaths and its percentage and group the output by "continent"



	continent	Highest_Infection_Count	Highest_Infection_Rate	Highest_Death_Count	Highest_Death_Rate
1	North America	102873924	62.4412264089336	1118800	0.330722302237759
2	Asia	99239252	63.3921452465691	531000	0.454867755318964
3	Europe	38791479	73.1762140251136	397642	0.564468505025468
4	South America	37319254	50.5919494869771	700556	0.645722937969176
5	Oceania	11153745	42.7564789542107	19906	0.24158385870546
6	Africa	4072533	50.7765419449686	102595	0.237639400601289

DATA MANIPULATION Cont...

```
-- Joining both the Deaths and Vaccinations tables
select *
from SQLProj1..CovidDeaths cd join SQLProj1..CovidVaccinations cv
on cd.location=cv.location
and cd.date = cv.date
```

- In this step, we join both the data sets (Deaths & Vaccinations) on two variables, location and date.
- We also make sure we only keep the variables that we are interested in

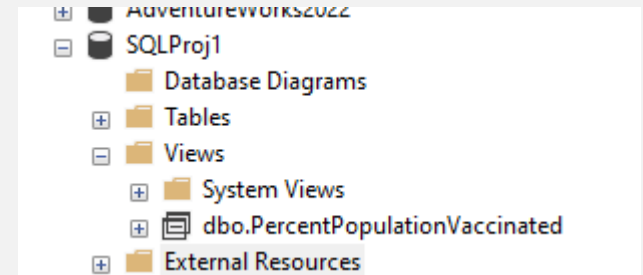
Results													
	continent	location	date	population	total_cases	new_cases	total_deaths	new_deaths	continent	location	date	population	new_vaccinations
1	Europe	Albania	2021-09-02 00:00:00.000	2842318	146387	1054	2498	3	Europe	Albania	2021-09-02 00:00:00.000	2842318	14648
2	Europe	Albania	2021-09-03 00:00:00.000	2842318	147369	982	2501	3	Europe	Albania	2021-09-03 00:00:00.000	2842318	NULL
3	Europe	Albania	2021-09-04 00:00:00.000	2842318	148222	853	2505	4	Europe	Albania	2021-09-04 00:00:00.000	2842318	NULL
4	Europe	Albania	2021-09-05 00:00:00.000	2842318	149117	895	2508	3	Europe	Albania	2021-09-05 00:00:00.000	2842318	NULL
5	Europe	Albania	2021-09-06 00:00:00.000	2842318	150101	984	2512	4	Europe	Albania	2021-09-06 00:00:00.000	2842318	NULL
6	Europe	Albania	2021-09-07 00:00:00.000	2842318	150997	896	2515	3	Europe	Albania	2021-09-07 00:00:00.000	2842318	14394
7	Europe	Albania	2021-09-08 00:00:00.000	2842318	151499	502	2519	4	Europe	Albania	2021-09-08 00:00:00.000	2842318	13687
8	Europe	Albania	2021-09-09 00:00:00.000	2842318	152239	740	2523	4	Europe	Albania	2021-09-09 00:00:00.000	2842318	12089
9	Europe	Albania	2021-09-10 00:00:00.000	2842318	153318	1079	2528	5	Europe	Albania	2021-09-10 00:00:00.000	2842318	10173
10	Europe	Albania	2021-09-11 00:00:00.000	2842318	154316	998	2531	3	Europe	Albania	2021-09-11 00:00:00.000	2842318	8621
11	Europe	Albania	2021-09-12 00:00:00.000	2842318	155293	977	2535	4	Europe	Albania	2021-09-12 00:00:00.000	2842318	3356
12	Europe	Albania	2021-09-13 00:00:00.000	2842318	156162	869	2539	4	Europe	Albania	2021-09-13 00:00:00.000	2842318	8605
13	Europe	Albania	2021-09-14 00:00:00.000	2842318	157026	864	2543	4	Europe	Albania	2021-09-14 00:00:00.000	2842318	7954
14	Europe	Albania	2021-09-15 00:00:00.000	2842318	157436	410	2548	5	Europe	Albania	2021-09-15 00:00:00.000	2842318	7525
15	Europe	Albania	2021-09-16 00:00:00.000	2842318	158431	995	2553	5	Europe	Albania	2021-09-16 00:00:00.000	2842318	7350
16	Europe	Albania	2021-09-17 00:00:00.000	2842318	159423	992	2557	4	Europe	Albania	2021-09-17 00:00:00.000	2842318	7253
17	Europe	Albania	2021-09-18 00:00:00.000	2842318	160385	942	2562	6	Europe	Albania	2021-09-18 00:00:00.000	2842318	NULL

FINAL ANALYSIS

```
-- Creating a View
use SQLProj1
go
create view
PercentPopulationVaccinated1 as
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 Rolling_Vaccination_Count
from SQLProj1..CovidDeaths cd
join SQLProj1..CovidVaccinations cv
on cd.location=cv.location
and cd.date = cv.date
where cd.continent is not null
```

- Lastly we produce the vaccination rolling count for each location and date and save this data set as a view by using simple create view function

	continent	location	date	population	new_vaccinations	Rolling_Vaccination_Count
751	Asia	Afghanistan	2022-01-22 00:00:00.000	41128772	NULL	6874
752	Asia	Afghanistan	2022-01-23 00:00:00.000	41128772	NULL	6874
753	Asia	Afghanistan	2022-01-24 00:00:00.000	41128772	NULL	6874
754	Asia	Afghanistan	2022-01-25 00:00:00.000	41128772	NULL	6874
755	Asia	Afghanistan	2022-01-26 00:00:00.000	41128772	NULL	6874
756	Asia	Afghanistan	2022-01-27 00:00:00.000	41128772	6868	13742
757	Asia	Afghanistan	2022-01-28 00:00:00.000	41128772	NULL	13742
758	Asia	Afghanistan	2022-01-29 00:00:00.000	41128772	NULL	13742
759	Asia	Afghanistan	2022-01-30 00:00:00.000	41128772	NULL	13742
760	Asia	Afghanistan	2022-01-31 00:00:00.000	41128772	NULL	13742
761	Asia	Afghanistan	2022-02-01 00:00:00.000	41128772	NULL	13742
762	Asia	Afghanistan	2022-02-02 00:00:00.000	41128772	NULL	13742
763	Asia	Afghanistan	2022-02-03 00:00:00.000	41128772	NULL	13742



- The view is saved within the views tab and we can directly call the table by using select statement

```
--calling the view
select * from
SQLProj1..PercentPopulationVaccinated
```



THANK YOU

Muthalib Abdul

