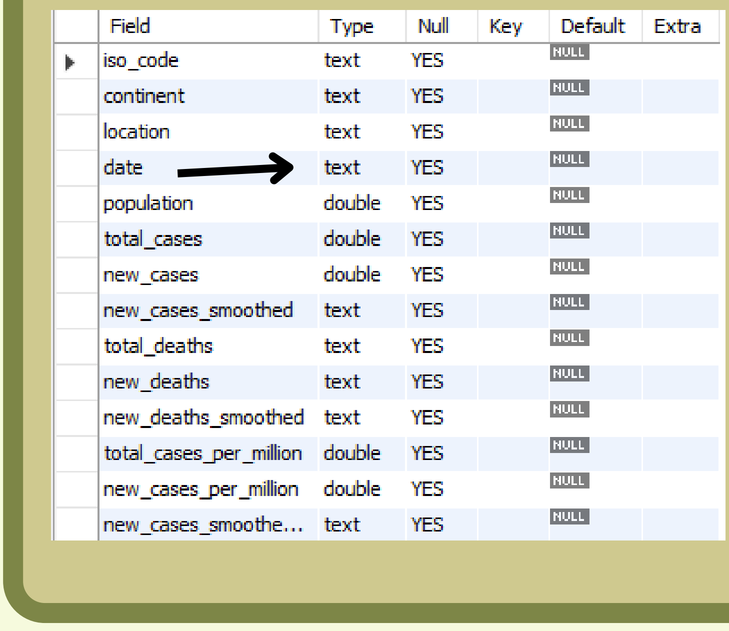
Data Exploration

Sharpened skills: Joins, Windows Functions, Aggregate Functions, CTE, Temporary Table, Creating Views, & Converting Data Types

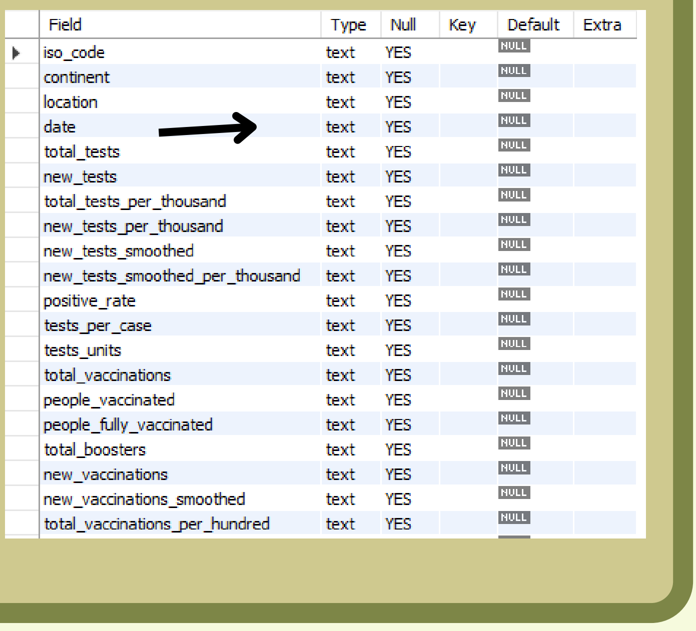
The data we want to explore comes from 'Our World in Data' about COVID-19. Regardless of the slowing trend of COVID-19 (although it is still in some places), the purpose of this project is to quickly resume the progress and the impact that we get by this time. In the first place, let us separate the data into two types: CovidDeath and CovidVaccine. Since the focus of this project is exploration, we separate the column in MS. Excel. The fields of each table are given below

```
DESCRIBE coviddeath;
```



Field	Type	Null	Key	Default	Extra
iso_code	text	YES		NULL	
continent	text	YES		NULL	
location	text	YES		NULL	
date	text	YES		NULL	
population	double	YES		NULL	
total_cases	double	YES		NULL	
new_cases	double	YES		NULL	
new_cases_smoothed	text	YES		NULL	
total_deaths	text	YES		NULL	
new_deaths	text	YES		NULL	
new_deaths_smoothed	text	YES		NULL	
total_cases_per_million	double	YES		NULL	
new_cases_per_million	double	YES		NULL	
new_cases_smoothe...	text	YES		NULL	

```
DESCRIBE covidvacine;
```

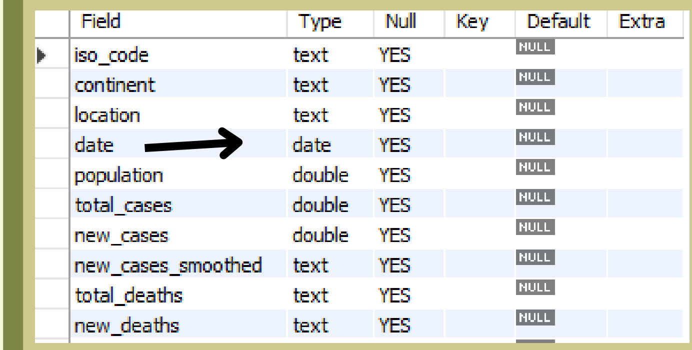


Field	Type	Null	Key	Default	Extra
iso_code	text	YES		NULL	
continent	text	YES		NULL	
location	text	YES		NULL	
date	text	YES		NULL	
total_tests	text	YES		NULL	
new_tests	text	YES		NULL	
total_tests_per_thousand	text	YES		NULL	
new_tests_per_thousand	text	YES		NULL	
new_tests_smoothed	text	YES		NULL	
new_tests_smoothed_per_thousand	text	YES		NULL	
positive_rate	text	YES		NULL	
tests_per_case	text	YES		NULL	
tests_units	text	YES		NULL	
total_vaccinations	text	YES		NULL	
people_vaccinated	text	YES		NULL	
people_fully_vaccinated	text	YES		NULL	
total_boosters	text	YES		NULL	
new_vaccinations	text	YES		NULL	
new_vaccinations_smoothed	text	YES		NULL	
total_vaccinations_per_hundred	text	YES		NULL	

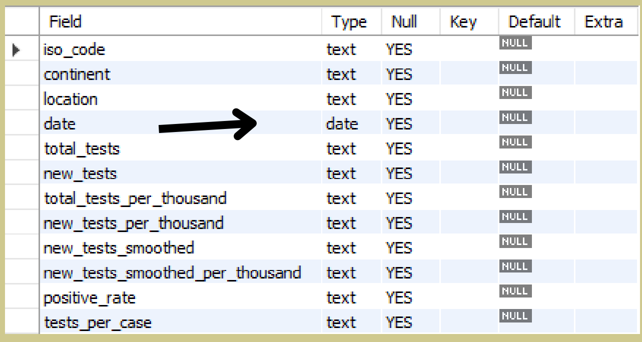
In the table above, we know the type of date is a text which non-desirable type for the 'date column'. Changing the type directly using definition and manipulation languages.

```
UPDATE coviddeath
SET date = STR_TO_DATE(date,'%d/%m/%Y');
ALTER TABLE coviddeath
MODIFY COLUMN date Date;
DESCRIBE coviddeath;
```

```
UPDATE covidvacine
SET date = STR_TO_DATE(date,'%d/%m/%Y');
ALTER TABLE covidvacine
MODIFY COLUMN date Date;
DESCRIBE covidvacine;
```



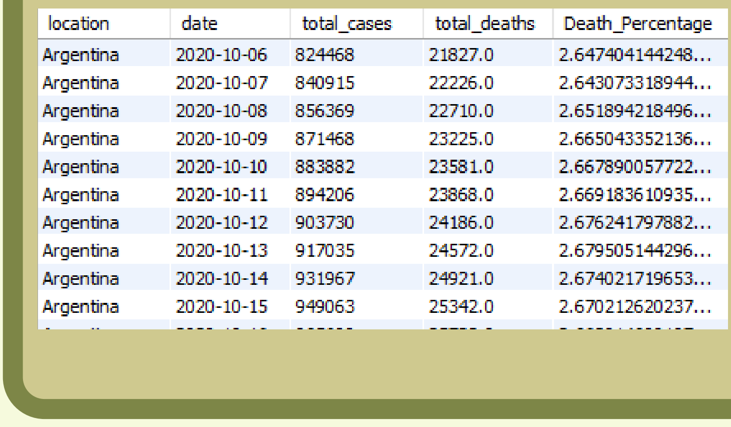
Field	Type	Null	Key	Default	Extra
iso_code	text	YES		NULL	
continent	text	YES		NULL	
location	text	YES		NULL	
date	date	YES		NULL	
population	double	YES		NULL	
total_cases	double	YES		NULL	
new_cases	double	YES		NULL	
new_cases_smoothed	text	YES		NULL	
total_deaths	text	YES		NULL	
new_deaths	text	YES		NULL	



Field	Type	Null	Key	Default	Extra
iso_code	text	YES		NULL	
continent	text	YES		NULL	
location	text	YES		NULL	
date	date	YES		NULL	
total_tests	text	YES		NULL	
new_tests	text	YES		NULL	
total_tests_per_thousand	text	YES		NULL	
new_tests_per_thousand	text	YES		NULL	
new_tests_smoothed	text	YES		NULL	
new_tests_smoothed_per_thousand	text	YES		NULL	
positive_rate	text	YES		NULL	
tests_per_case	text	YES		NULL	

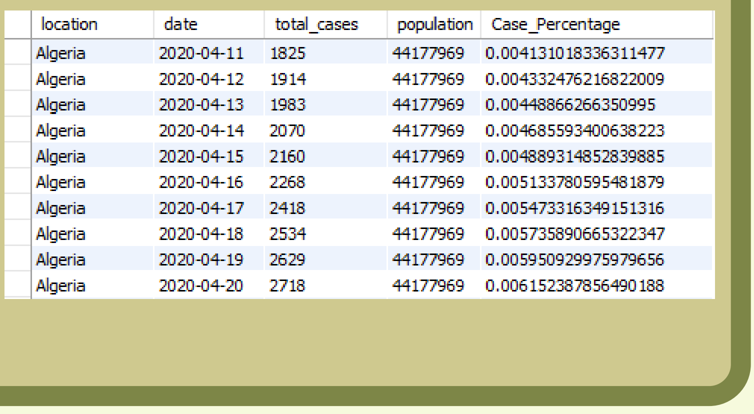
Let us focus on the Covid-Death table. First of all, we want to know the likelihood of dying (Death_Percentage) and the percentage of the population infected(Case_Percentage) in each country over time. In order to get the information, we select the case, death, and population from the table. We set the condition for the continent column to avoid incorrect values in the location column.

```
SELECT
 location,
    date,
    total_cases,
    total_deaths,
    (total_deaths/total_cases)*100 AS Death_Percentage
FROM coviddeath
WHERE continent != ''
ORDER BY location, date;
```



location	date	total_cases	total_deaths	Death_Percentage
Argentina	2020-10-06	824468	21827.0	2.647404144248...
Argentina	2020-10-07	840915	22226.0	2.643073318944...
Argentina	2020-10-08	856369	22710.0	2.651894218496...
Argentina	2020-10-09	871468	23225.0	2.665043352136...
Argentina	2020-10-10	883882	23581.0	2.667890057722...
Argentina	2020-10-11	894206	23868.0	2.669183610935...
Argentina	2020-10-12	903730	24186.0	2.676241797882...
Argentina	2020-10-13	917035	24572.0	2.679505144296...
Argentina	2020-10-14	931967	24921.0	2.674021719653...
Argentina	2020-10-15	949063	25342.0	2.670212620237...

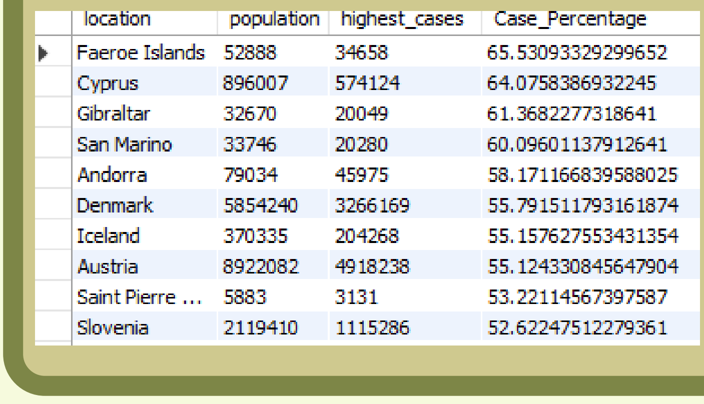
```
SELECT
 location,
    date,
    total_cases,
    population,
    (total_cases/population)*100 AS Case_Percentage
FROM coviddeath
WHERE continent != ''
ORDER BY location, date;
```



location	date	total_cases	population	Case_Percentage
Algeria	2020-04-11	1825	44177969	0.004131018336311477
Algeria	2020-04-12	1914	44177969	0.004332476216822009
Algeria	2020-04-13	1983	44177969	0.004488662663509995
Algeria	2020-04-14	2070	44177969	0.004685593400638223
Algeria	2020-04-15	2160	44177969	0.004889314852839885
Algeria	2020-04-16	2268	44177969	0.005133780595481879
Algeria	2020-04-17	2418	44177969	0.005473316349151316
Algeria	2020-04-18	2534	44177969	0.005735890665322347
Algeria	2020-04-19	2629	44177969	0.005950929975979656
Algeria	2020-04-20	2718	44177969	0.006152387856490188

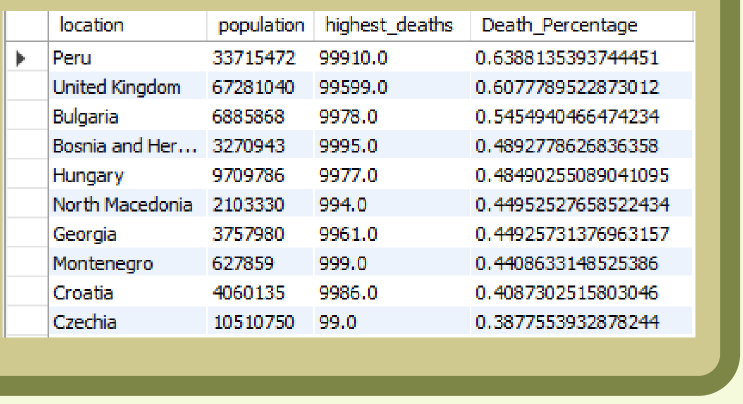
Secondly, we are looking for the countries with the highest infection rates (Case_Percentage) and deaths (Death_Percentage) according to the population of each country.

```
SELECT
 location,
    population,
    MAX(total_cases) AS highest_cases,
    MAX((total_cases/population))*100 AS
Case_Percentage
FROM coviddeath
GROUP BY location, population
ORDER BY Case_Percentage DESC;
```



	location	population	highest_cases	Case_Percentage
▶	Faeroe Islands	52888	34658	65.5309332929652
	Cyprus	896007	574124	64.0758386932245
	Gibraltar	32670	20049	61.3682277318641
	San Marino	33746	20280	60.0960113791264‍1
	Andorra	79034	45975	58.171166839588025
	Denmark	5854240	3266169	55.791511793161874
	Iceland	370335	204268	55.157627553431354
	Austria	8922082	4918238	55.124330845647904
	Saint Pierre ...	5883	3131	53.2211456739758‍7
	Slovenia	2119410	1115286	52.62247512279361

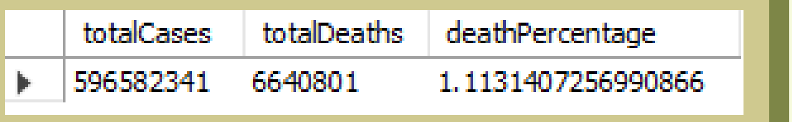
```
SELECT
 location,
    population,
    MAX(total_deaths) AS highest_deaths,
    MAX((total_deaths/population))*100 AS
Death_Percentage
FROM coviddeath
GROUP BY location, population
ORDER BY Death_Percentage DESC;
```



	location	population	highest_deaths	Death_Percentage
▶	Peru	33715472	99910.0	0.6388135393744451
	United Kingdom	67281040	99599.0	0.6077789522873012
	Bulgaria	6885868	9978.0	0.5454940466474234
	Bosnia and Her...	3270943	9995.0	0.4892778626836358
	Hungary	9709786	9977.0	0.48490255089041095
	North Macedonia	2103330	994.0	0.44952527658522434
	Georgia	3757980	9961.0	0.44925731376963157
	Montenegro	627859	999.0	0.4408633148525386
	Croatia	4060135	9986.0	0.4087302515803046
	Czechia	10510750	99.0	0.3877553932878244

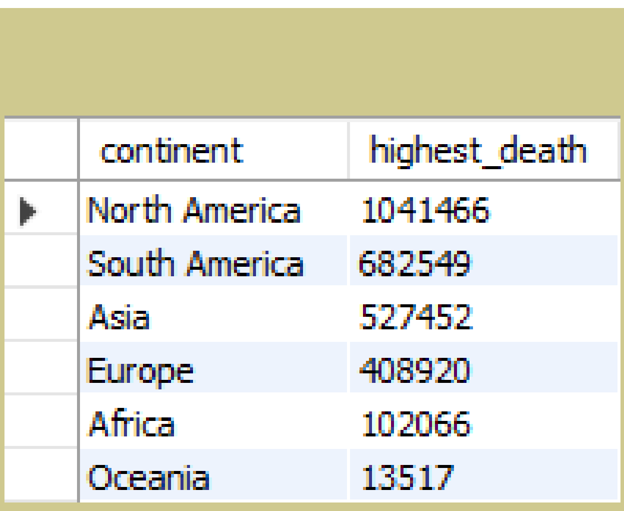
From the data above, Faeroe Islands and Peru are countries with the highest infection and death rates respectively. Next step, we are looking for the global number, which means the total cases and total deaths over the entire location, continent and country. For this purpose, we select new cases and new deaths columns, but we have to change the type of new deaths column from text to number first.

```
 SELECT
 SUM(new_cases) AS totalCases,
    SUM(cast(new_deaths AS UNSIGNED)) AS
totalDeaths,
    ((SUM(cast(new_deaths AS
UNSIGNED))/(SUM(new_cases))))*100 AS
deathPercentage
 FROM coviddeath
 WHERE continent !='';
```



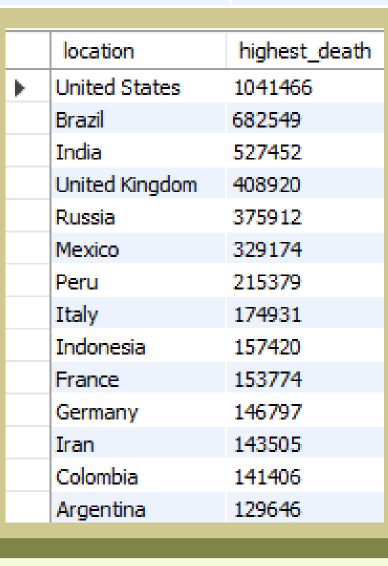
	totalCases	totalDeaths	deathPercentage
▶	596582341	6640801	1.1131407256990866

```
SELECT
 continent,
    MAX(cast(total_deaths AS UNSIGNED)) AS
highest_death
FROM coviddeath
WHERE continent !=''
GROUP BY continent
ORDER BY highest_death DESC;
```



	continent	highest_death
▶	North America	1041466
	South America	682549
	Asia	527452
	Europe	408920
	Africa	102066
	Oceania	13517

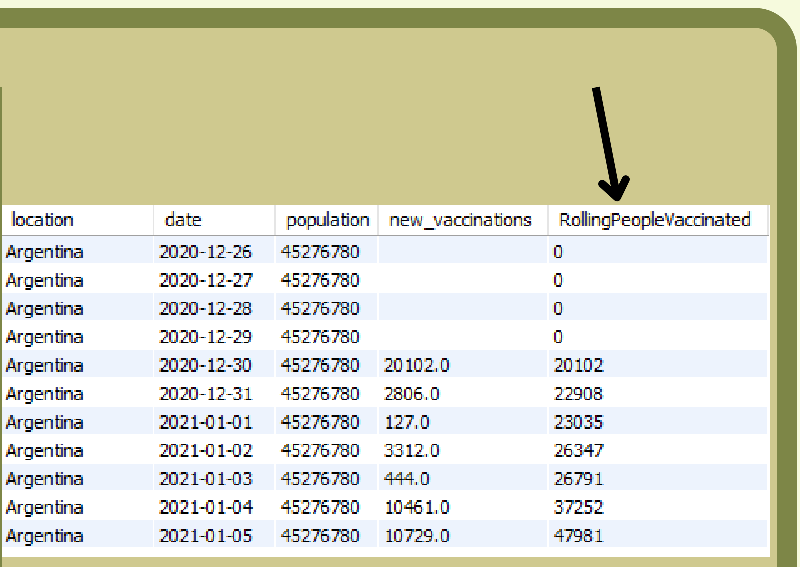
```
SELECT
 location,
    MAX(cast(total_deaths AS UNSIGNED)) AS
highest_death
FROM coviddeath
WHERE continent !=''
GROUP BY location
ORDER BY highest_death DESC;
```



	location	highest_death
▶	United States	1041466
	Brazil	682549
	India	527452
	United Kingdom	408920
	Russia	375912
	Mexico	329174
	Peru	215379
	Italy	174931
	Indonesia	157420
	France	153774
	Germany	146797
	Iran	143505
	Colombia	141406
	Argentina	129646

Now we are interested in the number of people who have been vaccinated at least on the first doses. To get the information, we join covid deaths and covid vaccine table on dates and locations. We also want to know the number of people got vaccines every day in different countries, so we use the window function.

```
SELECT
 dea.continent,
    dea.location,
    dea.date,
    dea.population,
    vac.new_vaccinations,
    SUM(CAST(vac.new_vaccinations AS
UNSIGNED)) OVER
   (PARTITION BY dea.location ORDER BY
dea.location, dea.date ) AS
RollingPeopleVaccinated
FROM coviddeath as dea
INNER JOIN  covidvacine as vac
 ON dea.date=vac.date AND
dea.location=vac.location
WHERE dea.continent !=''
ORDER BY dea.location, dea.date;
```



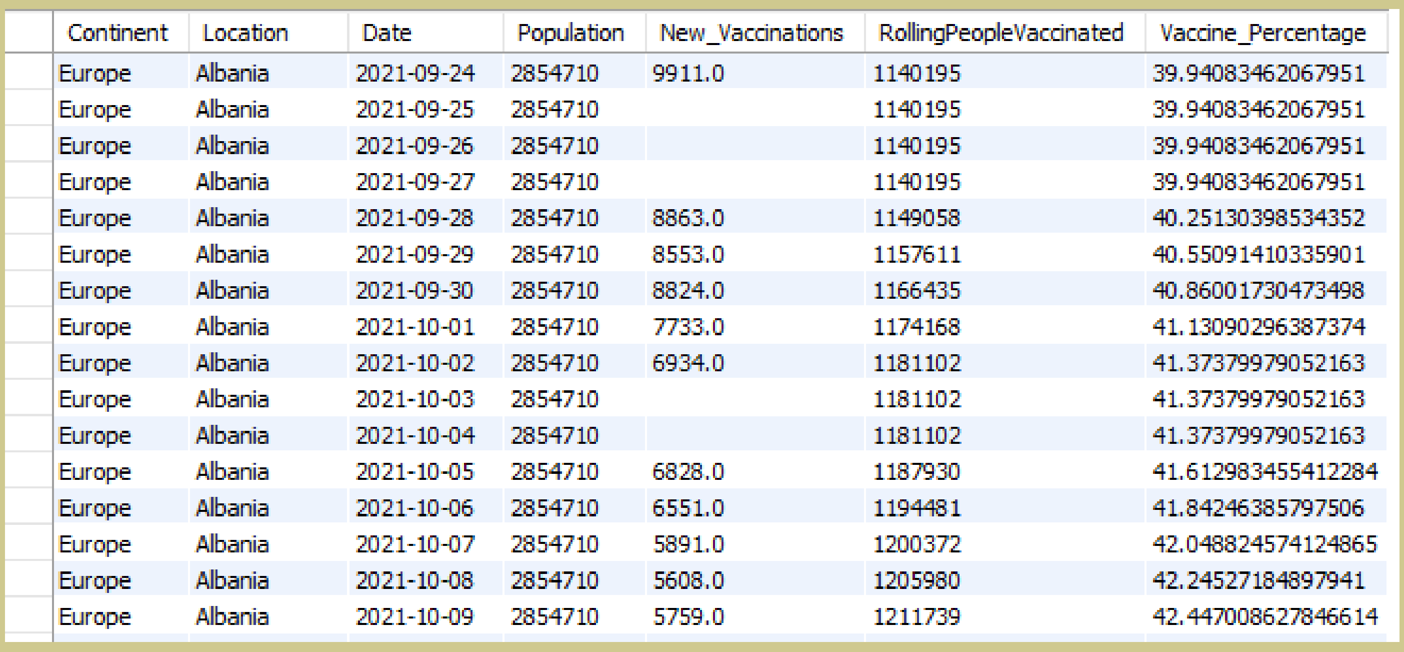
location	date	population	new_vaccinations	RollingPeopleVaccinated
Argentina	2020-12-26	45276780		0
Argentina	2020-12-27	45276780		0
Argentina	2020-12-28	45276780		0
Argentina	2020-12-29	45276780		0
Argentina	2020-12-30	45276780	20102.0	20102
Argentina	2020-12-31	45276780	2806.0	22908
Argentina	2021-01-01	45276780	127.0	23035
Argentina	2021-01-02	45276780	3312.0	26347
Argentina	2021-01-03	45276780	444.0	26791
Argentina	2021-01-04	45276780	10461.0	37252
Argentina	2021-01-05	45276780	10729.0	47981

Next step, we want to explore the new column RollingPeopleVaccinated. Since we can not select aliases column, we perform CTE and the temporary table to generate the same table.

```
WITH PopVsVac(Continent,Location, Date,
Population, New_Vaccinations,
RollingPeopleVaccinated)
AS (
SELECT
 dea.continent,
    dea.location,
    dea.date,
    dea.population,
    vac.new_vaccinations,
    SUM(CAST(vac.new_vaccinations AS
UNSIGNED)) OVER
  (PARTITION BY dea.location ORDER BY
dea.location, dea.date) AS
RollingPeopleVaccinated
FROM coviddeath as dea
INNER JOIN  covidvacine as vac
 ON dea.date=vac.date AND
dea.location=vac.location
WHERE dea.continent !=''
ORDER BY dea.location, date
)
SELECT *,
(RollingPeopleVaccinated/Population)*100 AS
Vaccine_Percentage
FROM PopVsVac;
```

```
DROP TEMPORARY TABLE  IF EXISTS
Vaccine_Percentage;
CREATE TEMPORARY TABLE Vaccine_Percentage (
Continent nvarchar(255),
Location nvarchar(255),
Date datetime,
Population numeric,
New_vaccination numeric,
RollingPeopleVaccinated numeric
);
INSERT INTO  Vaccine_Percentage
SELECT
 dea.continent,
    dea.location,
    dea.date,
    dea.population,
    CAST(vac.new_vaccinations AS DOUBLE) AS
New_Vaccinations,
    SUM(CAST(vac.new_vaccinations AS DOUBLE)) OVER
 (PARTITION BY dea.location ORDER BY dea.location,
dea.date) AS RollingPeopleVaccinated
FROM coviddeath as dea
INNER JOIN  covidvacine as vac
 ON dea.date=vac.date AND dea.location=vac.location
WHERE dea.continent !=''
ORDER BY dea.location, date;

SELECT *, (RollingPeopleVaccinated/Population)*100 AS
Vaccine_Percentages
FROM Vaccine_Percentage;
```

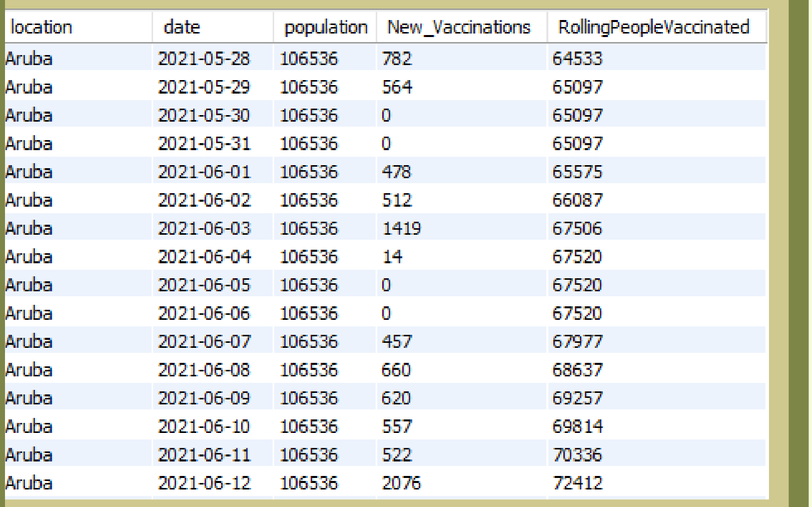


Continent	Location	Date	Population	New_Vaccinations	RollingPeopleVaccinated	Vaccine_Percentage
Europe	Albania	2021-09-24	2854710	9911.0	1140195	39.94083462067951
Europe	Albania	2021-09-25	2854710		1140195	39.94083462067951
Europe	Albania	2021-09-26	2854710		1140195	39.94083462067951
Europe	Albania	2021-09-27	2854710		1140195	39.94083462067951
Europe	Albania	2021-09-28	2854710	8863.0	1149058	40.25130398534352
Europe	Albania	2021-09-29	2854710	8553.0	1157611	40.55091410335901
Europe	Albania	2021-09-30	2854710	8824.0	1166435	40.86001730473498
Europe	Albania	2021-10-01	2854710	7733.0	1174168	41.13090296387374
Europe	Albania	2021-10-02	2854710	6934.0	1181102	41.37379979052163
Europe	Albania	2021-10-03	2854710		1181102	41.37379979052163
Europe	Albania	2021-10-04	2854710		1181102	41.37379979052163
Europe	Albania	2021-10-05	2854710	6828.0	1187930	41.612983455412284
Europe	Albania	2021-10-06	2854710	6551.0	1194481	41.84246385797506
Europe	Albania	2021-10-07	2854710	5891.0	1200372	42.048824574124865
Europe	Albania	2021-10-08	2854710	5608.0	1205980	42.24527184897941
Europe	Albania	2021-10-09	2854710	5759.0	1211739	42.447008627846614

Finally, we create our view to store the data for later visualizations.

```
DROP VIEW IF EXISTS VaccinePercentage;
CREATE VIEW  VaccinePercentage AS
SELECT
 dea.continent,
    dea.location,
    dea.date,
    dea.population,
    CAST(vac.new_vaccinations AS DOUBLE) AS
New_Vaccinations,
    SUM(CAST(vac.new_vaccinations AS
DOUBLE)) OVER
 (PARTITION BY dea.location ORDER BY
dea.location, dea.date) AS
RollingPeopleVaccinated
FROM coviddeath as dea
INNER JOIN  covidvacine as vac
 ON dea.date=vac.date AND
dea.location=vac.location
WHERE dea.continent !=''
ORDER BY dea.location, date;

SELECT * FROM vaccinePercentage;
```



location	date	population	New_Vaccinations	RollingPeopleVaccinated
Aruba	2021-05-28	106536	782	64533
Aruba	2021-05-29	106536	564	65097
Aruba	2021-05-30	106536	0	65097
Aruba	2021-05-31	106536	0	65097
Aruba	2021-06-01	106536	478	65575
Aruba	2021-06-02	106536	512	66087
Aruba	2021-06-03	106536	1419	67506
Aruba	2021-06-04	106536	14	67520
Aruba	2021-06-05	106536	0	67520
Aruba	2021-06-06	106536	0	67520
Aruba	2021-06-07	106536	457	67977
Aruba	2021-06-08	106536	660	68637
Aruba	2021-06-09	106536	620	69257
Aruba	2021-06-10	106536	557	69814
Aruba	2021-06-11	106536	522	70336
Aruba	2021-06-12	106536	2076	72412