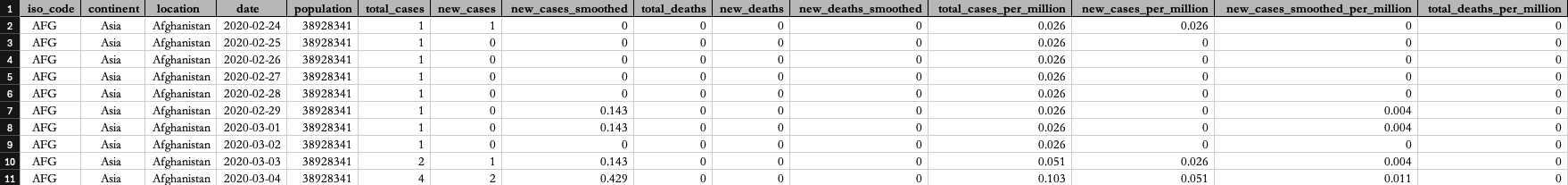
**COVID-19 Portfolio Project- Data exploration and Data Manipulation using SQL**

**Let’s have an overview of the Covid-19 Data:**

**Select all the data**

MySQL> select \* from CovidDeaths;

****

**Total cases vs Total Deaths**

MySQL> select location, date, total\_cases, total\_deaths, (total\_deaths/total\_cases) \* 100 as DeathRatio from CovidDeaths where location like '%states%' order by DeathRatio desc;

**A table with numbers and numbers

AI-generated content may be incorrect.**

**Likelihood of contracting COVID-19**

MySQL> Select Location, date, total\_cases,total\_deaths, (total\_deaths/total\_cases)\*100 as DeathRatio from CovidDeaths Where location like '%states%' and continent != '' order by date desc, DeathRatio desc;

**From the image below,**

**#1:**

**The initial peak covid data shows that the total cases skyrocketed from 55 to 107.**

**#2:**

**The total cases peaked from November 8, 2020 with over 10M+ cases with a DeathRatio of 2.37%.**

**#3:**

**The DeathRatio also plummeted to 1.7% in the later stage.**

**This shows that you have 1.7% chance of contracting COVID-19**

**A screenshot of a table

AI-generated content may be incorrect.**

**Shows the percentage of population got covid**

MySQL> select location, date, population, total\_cases, (total\_cases/population)

\* 100 as DeathRatio from CovidDeaths where location like '%states%' order by DeathRatio desc;

A table with numbers and a number on it

AI-generated content may be incorrect.

**Looking at countries with highest infection rate vs population**

**MySQL> select location, population, Max(total\_cases) as HighestInfectionCount, Max((total\_cases/population)) \* 100 as PercentPopulationInfected from CovidDeaths group by location, population order by PercentPopulationInfected desc;**

**A table with numbers and text

AI-generated content may be incorrect.**

**Showing the countries with highest death count per population**

MySQL> select location, max(cast(total\_deaths as UNSIGNED)) as TotalDeathCount from CovidDeaths where continent != '' group by location order by TotalDeathCount desc;

**Breaking things down by continent**

**showing deaths count by continent**

MySQL> select date, sum(new\_cases) as TotalCases,sum(new\_deaths) as TotalDeaths, sum(new\_deaths) / sum(new\_cases) \* 100 as DeathPercent from CovidDeaths where continent != '' group by date order by 1,2; — globally

MySQL> select continent, max(cast(total\_deaths as UNSIGNED)) as TotalDeathCount from CovidDeaths where continent != '' group by continent order by TotalDeathCount desc;— continent

MySQL> select sum(new\_cases) as TotalCases, sum(new\_deaths) as TotalDeaths, sum(new\_deaths) / sum(new\_cases) \* 100 as DeathPercent from CovidDeaths where continent!= '' order by 1,2;— across the world

**Looking for total pop vs Vacc**

MySQL> With PopvsVacc (continent, location, date, population, new\_vaccinations, RVacc) as (select d.continent, d.location, d.date, d.population, v.new\_vaccinations, sum(cast(v.new\_vaccinations as unsigned)) over (partition by d.location order by d.location, d.date) as RVacc from CovidDeaths as d JOIN CovidVaccinations as v on d.location = v.location and d.date = v.date where d.continent != '') select \*, (RVacc/population) \* 100 from PopvsVacc limit 1000;

**Creating a temp table**

MySQL> Drop temporary table if exists PercentPopVaccinated; create temporary table PercentPopVaccinated (continent varchar(50), location varchar(100), date DATE, population bigint, new\_vaccinations bigint, RVacc bigint); insert into PercentPopVaccinated select d.continent, d.location, d.date, d.population, v.new\_vaccinations, sum(cast(v.new\_vaccinations as unsigned)) over (partition by d.location order by d.location, d.date) as RVacc from CovidDeaths as d JOIN CovidVaccinations as v on d.location = v.location and d.date = v.date; select \*, RVacc/population \* 100 as PercentVaccinated from PercentPopVaccinated limit 1000;

**Creating a view**

MySQL> create view PercentPopVaccinated as select d.continent, d.location, d.date, d.population, v.new\_vaccinations, sum(cast(v.new\_vaccinations as unsigned)) over (partition by d.location order by d.location, d.date) as RVacc from CovidDeaths as d JOIN CovidVaccinations as v on d.location =

v.location and d.date = v.date where d.continent != '' limit 1000;