/\*

Covid 19 Data Exploration

Skills used: Joins, CTE's, Creating Tables, Windows Functions, Aggregate Functions, Creating Views, Converting Data Types.

\*/

Select \*

From `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

Where continent IS NOT NULL

order by Location, date;

# Selecting the Data that we are going to be starting with

SELECT Location, date, total\_cases, new\_cases, total\_deaths, population

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

/\*Is Not Null because the rows that = NULL are the continents \*/

ORDER BY Location, date;

# Looking at Total Cases VS Total Deaths

# Shows likelihood of dying if you contract COVID-19 within the United States

SELECT Location, date, total\_cases, total\_deaths, (total\_deaths/total\_cases)\*100 AS DeathPercentage

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE location = 'United States' AND continent IS NOT NULL

ORDER BY Location, date DESC LIMIT 1;

# Looking at Total Cases VS Population

# Shows percentage of population that got COVID-19

SELECT Location, date, Population, total\_cases, (total\_cases/Population\*100) AS percentage\_of\_population\_infected

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE location LIKE '%States%' AND continent IS NOT NULL

ORDER BY Location, date DESC LIMIT 1;

# Looking at Countries with Highest Infection Rate compared to Population

SELECT Location, population, MAX(total\_cases) AS HighestInfectionCount, MAX((total\_cases/Population))\*100 AS percentage\_of\_population\_infected

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

GROUP BY population, location

ORDER BY 4 desc;

# Showing countries with highest death count per Population

SELECT Location, MAX(total\_deaths) AS TotalDeathCount

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

GROUP BY location

ORDER BY 2 desc;

# Breaking it down by Continent

-- Showing contintents with the highest death count per population

SELECT continent, MAX(total\_deaths) AS TotalDeathCount

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

GROUP BY continent

ORDER BY 2 desc;

#Global Numbers:

-- By Date

SELECT date, SUM(new\_cases) AS TotalCases, SUM(new\_deaths) AS TotalDeaths, SUM(new\_deaths)/SUM(New\_Cases)\*100 AS DeathPercentage

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

GROUP BY date

ORDER BY 1,2;

--Overall

SELECT SUM(new\_cases) AS TotalCases, SUM(new\_deaths) AS TotalDeaths, SUM(new\_deaths)/SUM(New\_Cases)\*100 AS DeathPercentage

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths`

WHERE continent IS NOT NULL

ORDER BY 1,2;

# Looking at Total population VS Vaccinations

-- Shows Percentage of Population that has recieved at least one Covid-19 Vaccine

SELECT dea.continent, dea.location, dea.date, dea.population, vac.new\_vaccinations, SUM(vac.new\_vaccinations) OVER (Partition by dea.location ORDER BY dea.location, dea.date) AS RollingPeopleVaccinated

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths` AS dea

INNER JOIN `radiant-anchor-360514.Covid\_dataset.covid\_vaccinations` AS vac

ON dea.location = vac.location

AND dea.date = vac.date

WHERE dea.continent IS NOT NULL

ORDER BY 2,3;

# USE CTE (common table expression) to perform Calculation on Partition By in the previous query

With PopvsVac\_cte AS (

SELECT dea.continent, dea.location, dea.date, dea.population, vac.new\_vaccinations, SUM(vac.new\_vaccinations) OVER (Partition by dea.location ORDER BY dea.location, dea.date) AS RollingPeopleVaccinated

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths` AS dea

INNER JOIN `radiant-anchor-360514.Covid\_dataset.covid\_vaccinations` AS vac

ON dea.location = vac.location

AND dea.date = vac.date

WHERE dea.continent IS NOT NULL)

SELECT \*, (RollingPeopleVaccinated/Population)\*100 AS PercentofPopulationVaccinated

FROM PopvsVac\_cte;

# The CTE is useful because in the previous query we cant reference "rollingpeoplevaccinated" / by population \*100 since it will all be in the SELECT clause

# Creating Table to perform Calculation on Partition By in previous query

DROP TABLE IF EXISTS radiant-anchor-360514.Covid\_dataset.PercentPopulationVaccinated;

CREATE TABLE radiant-anchor-360514.Covid\_dataset.PercentPopulationVaccinated

(continent string(255),

location string(255),

date datetime,

population numeric,

new\_vaccinations numeric,

RollingPeopleVaccinated numeric)

AS

SELECT dea.continent, dea.location, dea.date, dea.population, vac.new\_vaccinations, SUM(vac.new\_vaccinations) OVER (Partition by dea.location ORDER BY dea.location, dea.date) AS RollingPeopleVaccinated

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths` AS dea

INNER JOIN `radiant-anchor-360514.Covid\_dataset.covid\_vaccinations` AS vac

ON dea.location = vac.location

AND dea.date = vac.date

WHERE dea.continent IS NOT NULL;

--Querying the table created above

SELECT \*, (RollingPeopleVaccinated/Population)\*100 AS PercentofPopulationVaccinated

FROM radiant-anchor-360514.Covid\_dataset.PercentPopulationVaccinated

ORDER BY 2,3;

# Creating View to store data for visualizations

CREATE VIEW radiant-anchor-360514.Covid\_dataset.PercentPopulationVaccinated2 AS

SELECT dea.continent, dea.location, dea.date, dea.population, vac.new\_vaccinations, SUM(vac.new\_vaccinations) OVER (Partition by dea.location ORDER BY dea.location, dea.date) AS RollingPeopleVaccinated

FROM `radiant-anchor-360514.Covid\_dataset.covid\_deaths` AS dea

INNER JOIN `radiant-anchor-360514.Covid\_dataset.covid\_vaccinations` AS vac

ON dea.location = vac.location

AND dea.date = vac.date

WHERE dea.continent IS NOT NULL;