**--selecting data to be used**

**SELECT location, date, total\_cases, new\_cases, total\_deaths, population,**

**FROM `PortfolioProject.CovidDeaths`**

**ORDER BY location, date**

**--calculating at Total Cases vs Total Deaths**

**--displays likelyhood of dying if contract covid**

**SELECT location, date, total\_cases, population, (total\_cases/population)\*100 AS Infection\_rate**

**FROM `PortfolioProject.CovidDeaths`**

**WHERE location = 'United States'**

**ORDER BY location, date**

**--looking at countries with highest infection rates**

**--MAX function needs to be applied to both**

**SELECT location, MAX(total\_cases) AS HighestInfectionRate, population, MAX((total\_cases/population))\*100 AS Infection\_rate**

**FROM `PortfolioProject.CovidDeaths`**

**GROUP BY location, population**

**ORDER BY Infection\_rate DESC**

**--looking at countries with highest death counts**

**--USE CAST function to convert total\_deaths to integer format (no longer needed in 2023)**

**-- but helpful if a column with numbers is not coded as integers but rather strings**

**SELECT location, MAX(CAST(total\_deaths AS INT)) AS DeathCount**

**FROM `PortfolioProject.CovidDeaths`**

**GROUP BY location**

**ORDER BY DeathCount DESC**

**--looking at countries with highest death counts**

**--the above gives groupings of countries as location b/c that's how it was put into the database,**

**--such as "Europe/High Income" b/c "continent" column is null= add is not null statement**

**SELECT location, MAX(CAST(total\_deaths AS INT)) AS DeathCount**

**FROM `PortfolioProject.CovidDeaths`**

**WHERE continent is NOT NULL**

**GROUP BY location**

**ORDER BY DeathCount DESC**

**-----------------LETS BREAK THINGS DOWN BY CONTINENT NOW-----------------------------**

**SELECT continent, MAX(CAST(total\_deaths AS INT)) AS DeathCount**

**FROM `PortfolioProject.CovidDeaths`**

**WHERE continent is NOT NULL**

**GROUP BY continent**

**ORDER BY DeathCount DESC**

**--with this query, North America only includes US**

**--now see a higher # for North Am which indicates its pulling from Canada & MX**

**SELECT continent, MAX(CAST(total\_deaths AS INT)) AS DeathCount**

**FROM `PortfolioProject.CovidDeaths`**

**WHERE continent is NOT NULL**

**GROUP BY continent**

**ORDER BY DeathCount DESC**

**--GLOBAL NUMBERS**

**SELECT SUM(new\_cases) AS total\_cases, SUM(new\_deaths) AS total\_deaths, SUM(new\_deaths)/SUM(new\_cases)\*100 AS DeathPercent**

**FROM `PortfolioProject.CovidDeaths`**

**WHERE continent IS NOT NULL**

**--GROUP BY date**

**ORDER BY 1,2**

**------CHECKING OUT THE OTHER TABLE------**

**--Looking at total populations vs vaccination**

**--must specify what table we want in SELECT Clause**

**SELECT dea.continent, dea.location, dea.date, dea.population, vac.new\_vaccinations**

**FROM `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date**

**WHERE dea.continent IS NOT NULL**

**ORDER BY 2,3**

**--Advanced Aggregation of above query**

**--Partition by= breaking SUM function up by location so it starts --over at each new country**

**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 `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date**

**WHERE dea.continent IS NOT NULL**

**ORDER BY 2,3**

**--adds up vaccinations by country by date= next row down is sum of prev total + new\_vaccinations = Rolling Count**

**--Checking out vaccination RATE**

**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, (RollingPeopleVaccinated/Population)\*100 --cant use column that you just created to do a calculation =see next query for creating a temporary table**

**FROM `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date**

**WHERE dea.continent IS NOT NULL**

**ORDER BY 2,3**

**--USING A CTE- virtual table to simplify queries**

**--use WITH to create CTE, and add RollingPeopleVaccinated as a --column, then AS, then subquery from above between parenthesis**

**--# of columns in CTE must be same as # columns in SELECT subquery**

**WITH PopVsVac --naming your query**

**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 `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date**

**WHERE dea.continent IS NOT NULL**

**-- ORDER BY 2,3 clause can't be in there will get an error**

**)**

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

**FROM PopVsVac**

**--ANOTHER TEMP TABLE---**

**--unable to create table in BigQuery but this is the correct syntax**

**CREATE TABLE PortfolioProject.PercentPopulationVaccinated**

**(**

**continent STRING,**

**--must specify type of data when creating new chart**

**location STRING,**

**date datetime,**

**population INTEGER,**

**new\_vaccinations INTEGER,**

**RollingPeopleVaccinated INTEGER,**

**);**

**INSERT INTO PortfolioProject.PercentPopulationVaccinated**

**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 `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date;**

**--WHERE dea.continent IS NOT NULL**

**--ORDER BY 2,3**

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

**FROM PortfolioProject.PercentPopulationVaccinated**

**--CREATING VIEWS to be able to store data for later, use in Tablaeu**

**--also limited in bigquery**

**--can create a view of anything and all calculations**

**--can now query off of that view**

**CREATE VIEW PortfolioProject.PercentPopulationVaccinated 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 `PortfolioProject.CovidDeaths` AS dea**

**JOIN `PortfolioProject.vaccinationsCovid` AS vac**

**ON dea.location=vac.location AND dea.date=vac.date**

**WHERE dea.continent IS NOT NULL**

**--ORDER BY 2,3**