```sql
/*
NAME: Tayo Obafaiye
SQL Data Exploration Portforlio Project
COVID-19 Data: Global data on COVID Deaths and Vaccinations
*/

SELECT *
FROM PortfolioProject..CovidDeaths
ORDER BY 3, 4

SELECT *
FROM PortfolioProject..CovidVaccinations
ORDER BY 3, 4

-- Columns/Attributes for data exploration
SELECT location, date, total_cases, new_cases, total_deaths, population
FROM PortfolioProject..CovidDeaths
ORDER BY 1, 2

-- Total Cases vs. Total Deaths
SELECT location, date, total_cases, total_deaths, (total_deaths/total_cases)*100 AS
Death_Percentage
FROM PortfolioProject..CovidDeaths
ORDER BY 1, 2

-- In the U.S.
SELECT location, date, total_cases, total_deaths, (total_deaths/total_cases)*100 AS
Death_Percentage
FROM PortfolioProject..CovidDeaths
WHERE location LIKE '%states%'
ORDER BY 1, 2

-- Total Cases vs. Population in the U.S.
-- Shows percentage of population with covid
SELECT location, date, total_cases, population, (total_cases/population)*100 AS
Case_Percentage
FROM PortfolioProject..CovidDeaths
WHERE location LIKE '%states%'
ORDER BY 1, 2

-- Countries with the highest infection rate compared to population
SELECT location, population, MAX(total_cases) AS Highest_Infection_Count,
MAX((total_cases/population))*100 AS Percent_of_Population_Infected
FROM PortfolioProject..CovidDeaths
GROUP BY location, population
ORDER BY 1, 2

SELECT location, population, MAX(total_cases) AS Highest_Infection_Count,
MAX((total_cases/population))*100 AS Percent_of_Population_Infected
FROM PortfolioProject..CovidDeaths
GROUP BY location, population
ORDER BY Percent_of_Population_Infected DESC

--Countries with the highest death count per population
SELECT location, MAX(total_deaths) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
GROUP BY location
ORDER BY Total_Death_Count DESC

SELECT location, MAX(cast(total_deaths AS INT)) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
GROUP BY location
```

```sql
ORDER BY Total_Death_Count DESC

SELECT *
FROM PortfolioProject..CovidDeaths
WHERE continent IS NOT NULL
ORDER BY 3, 4

-- with UPDATED location, by country
SELECT location, MAX(cast(total_deaths AS INT)) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
WHERE continent IS NOT NULL
GROUP BY location
ORDER BY Total_Death_Count DESC

-- Continent/Continents with the highest death counts
SELECT continent, MAX(cast(total_deaths AS INT)) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
WHERE continent IS NOT NULL
GROUP BY continent
ORDER BY Total_Death_Count DESC

SELECT location, MAX(cast(total_deaths AS INT)) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
WHERE continent IS NULL
GROUP BY location
ORDER BY Total_Death_Count DESC

SELECT location, MAX(cast(total_deaths AS INT)) AS Total_Death_Count
FROM PortfolioProject..CovidDeaths
WHERE continent IS NULL AND location = 'North America'
GROUP BY location
ORDER BY Total_Death_Count DESC

-- Global
SELECT date, SUM(new_cases) AS Total_Cases, SUM(cast(new_deaths AS INT)) AS
Total_Deaths,
          SUM(cast(new_deaths AS INT))/SUM(new_cases)*100 AS Death_Percentage
FROM PortfolioProject..CovidDeaths
WHERE continent IS NOT NULL
GROUP BY date
ORDER BY 1, 2

SELECT SUM(new_cases) AS Total_Cases, SUM(cast(new_deaths AS INT)) AS Total_Deaths,
          SUM(cast(new_deaths AS INT))/SUM(new_cases)*100 AS Death_Percentage
FROM PortfolioProject..CovidDeaths
WHERE continent IS NOT NULL
ORDER BY 1, 2

SELECT *
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date

-- Total Population vs. Vaccinations
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL
ORDER BY 2, 3
```

```sql
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location)
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL
ORDER BY 2, 3

SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER
BY cd.location, cd.date) AS Rolling_People_Vaccinated
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL
ORDER BY 2, 3

-- USING CTE
WITH PopvsVac (Continent, Location, Date, Population, New_Vaccinations,
RollingPeopleVaccinated) AS
(
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER
BY cd.location, cd.date) AS Rolling_People_Vaccinated
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL
)
SELECT *, (RollingPeopleVaccinated/Population)*100
FROM PopvsVac

-- TEMP TABLE
CREATE TABLE #PercentPopulationVaccinated
(
Continent NVARCHAR(255),
Location NVARCHAR(255),
Date DATETIME,
Population NUMERIC,
New_Vaccinations NUMERIC,
RollingPeopleVaccinated NUMERIC
)

INSERT INTO #PercentPopulationVaccinated
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER
BY cd.location, cd.date) AS Rolling_People_Vaccinated
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL

SELECT *, (RollingPeopleVaccinated/Population)*100
FROM #PercentPopulationVaccinated

DROP TABLE IF EXISTS #PercentPopulationVaccinated
CREATE TABLE #PercentPopulationVaccinated
```

```sql
(
Continent NVARCHAR(255),
Location NVARCHAR(255),
Date DATETIME,
Population NUMERIC,
New_Vaccinations NUMERIC,
RollingPeopleVaccinated NUMERIC
)

INSERT INTO #PercentPopulationVaccinated
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER
BY cd.location, cd.date) AS Rolling_People_Vaccinated
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
SELECT *, (RollingPeopleVaccinated/Population)*100
FROM #PercentPopulationVaccinated

-- Creating View to store date for visualizations
CREATE VIEW Percent_Population_Vaccinated AS
SELECT cd.continent, cd.location, cd.date, cd.population, cv.new_vaccinations,
          SUM(CONVERT(INT, cv.new_vaccinations)) OVER (PARTITION BY cd.location ORDER
BY cd.location, cd.date) AS Rolling_People_Vaccinated
FROM PortfolioProject..CovidDeaths cd
JOIN PortfolioProject..CovidVaccinations cv
       ON cd.location = cv.location
       AND cd.date = cv.date
WHERE cd.continent IS NOT NULL

SELECT *
FROM Percent_Population_Vaccinated

/*
Credit to Alex The Analyst https://www.youtube.com/watch?v=qfyynHBFOsM&t=3s
*/
```