Day 3

SQL Data Exploration Project

<https://ourworldindata.org/covid-deaths>

Broke data set down into two relevant tables as to not JOIN every query.

A screenshot of a computer file

Description automatically generated

As to be expected, things did not go as they were supposed to so now, we TROUBLESHOOT!

Problem 1. Files were CSV's, and we needed them to be Excel Worksheet files, simple solution convert them.

A screenshot of a computer

Description automatically generated

Problem 2. They won't import the way described to me because where Alex has them sent to the SQL Server Native Client 11.0, I do not have that as Microsoft removed that in SQL 2022. Instead, it recommends that I use Microsoft OLE DB Driver, which also didn't work.

[](https://learn.microsoft.com/en-us/sql/relational-databases/native-client/applications/installing-sql-server-native-client?view=sql-server-ver16)A screenshot of a computer error

Description automatically generated

 Problem then solved by testing another similar named destination which happened to work. A screenshot of a computer

Description automatically generated

The project can continue, thank you Microsoft documentation for being slightly incorrect, I move on with my life.

The Project

Note: Most of the project will use data from the US, as I am from there.

Covid Deaths

Make sure both data sets are viewable and accurate.

A screenshot of a computer

Description automatically generated

Selecting workable data

A close-up of words

Description automatically generated

Total percentages of deaths

A close-up of words

Description automatically generated

A simple WHERE statement will allow you to filter by area.

A close up of a message

Description automatically generated

Total Cases vs Population

A close-up of a computer screen

Description automatically generated

Filtering by countries with the highest infection rate relative to population, ordered by highest percentage to lowest.

A screenshot of a computer

Description automatically generated

Showing countries with highest death count per population, didn’t work at first because it is a float, not an nvarchar.

A screenshot of a computer code

Description automatically generated

Changing “total\_deaths” to an integer with the CAST function mostly solves our problem. However, there are continents in our data.



Output.

A screenshot of a computer

Description automatically generated

Adding a WHERE statement that removes Locations that do not have a specified continent will fix this issue.



Ex.

A screenshot of a computer

Description automatically generated

Then showing continents, because why not.

A screenshot of a computer

Description automatically generated

Output.

A screenshot of a computer

Description automatically generated

Then global numbers by day, again using cast to turn a float variable into an integer.

A close up of a computer screen

Description automatically generated

The same data, but total death count and percentage based on total cases.

A close up of a computer screen

Description automatically generated

Output.

A screenshot of a computer

Description automatically generated

Covid Vaccinations

Begin by aggregating the data.

A screenshot of a computer

Description automatically generated

Clean the results to give only data you want.

A screen shot of a computer

Description automatically generated

Output.

A screenshot of a computer

Description automatically generated

Side note: Here I enabled Word Wrap because it was annoying me how long things were getting and having to scroll.

Same way to cast, but CONVERT command instead.



We narrow it down to how many people per location are vaccinated by day, and create a rolling total by using the two data sets.

A screenshot of a computer code

Description automatically generated

Because I won’t be returning to this particular set of data very often, I use CTE to aggregate information, as opposed to a temp table, to show the total percentage of vacinations/population.

A screenshot of a computer code

Description automatically generated

Output.

A screenshot of a computer

Description automatically generated

Temp table cause it’s not that hard.

A screenshot of a computer code

Description automatically generated

Output. (Same as above, which means it worked but isn’t organized the same way).

A screenshot of a table

Description automatically generated

Creating a VIEW of the data

A screenshot of a computer

Description automatically generated

Save our file to solidify the work for later when we learn Tableau.



Finish by uploading it all to Github!