### **BIA WORKSHOP – POWER BI USING DAX LAB 1 – Data preparation and DAX**

### Prepared exclusively for 2023 BIA Graduating Seniors (with an INFO 3300 prereq)

### May 2023

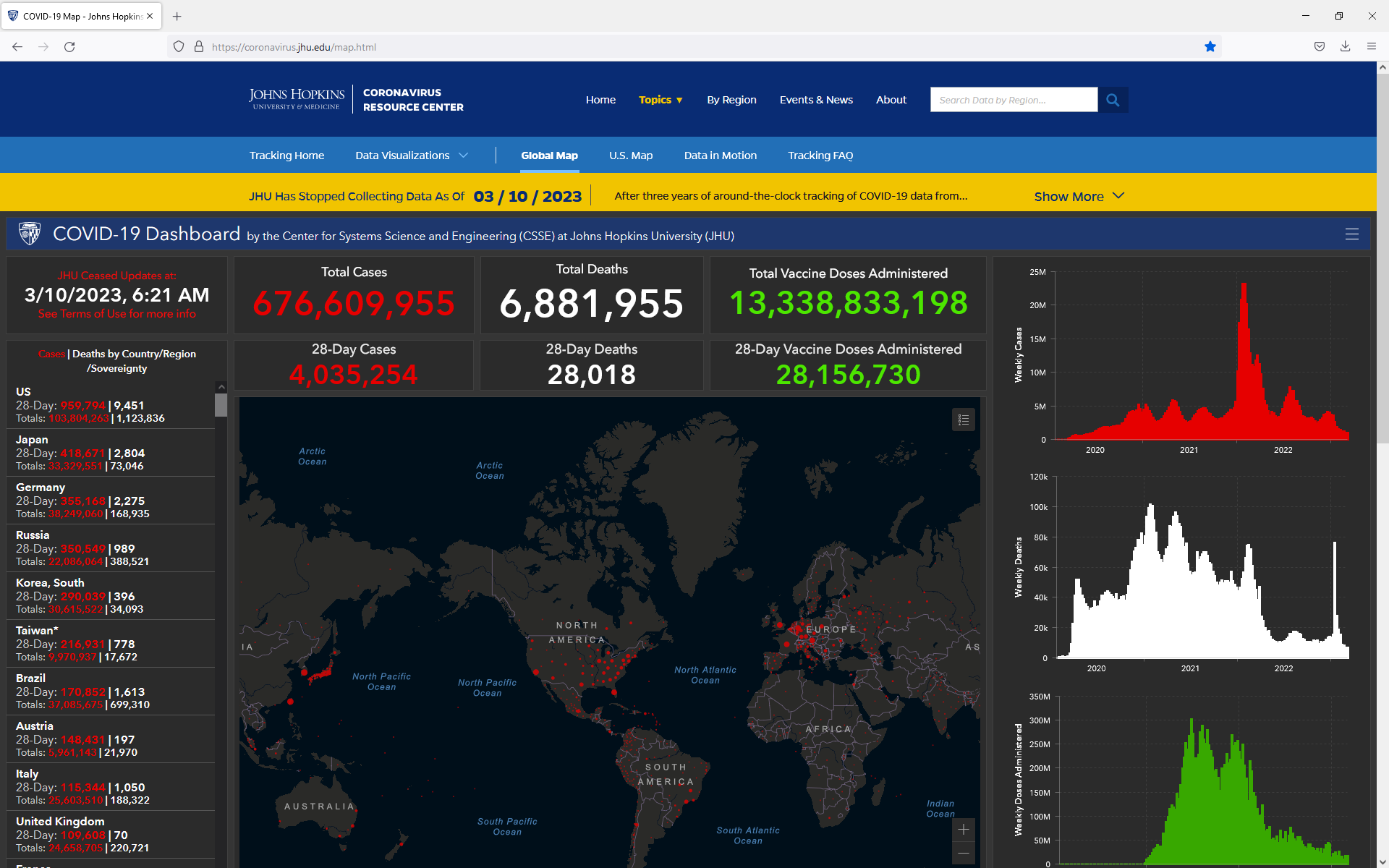
## **LAB RESOURCES**

|  |  |
| --- | --- |
|  | <https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv> |
|  | <https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv> |
|  | <https://www.worldometers.info/world-population/population-by-country/> |

**SITUATION:** On December 31, 2019, the World Health Organization (WHO) was informed of an outbreak of “pneumonia of unknown cause” detected in Wuhan City, Hubei Province, China – the seventh-largest city in China with 11 million residents. The Center for Systems Science and Engineering (CSSE) at Johns Hopkins University created a GitHub repository to share data on the location and number of confirmed COVID-19 cases, deaths and recoveries for all affected countries. It was developed to provide researchers, public health authorities and the general public with a user-friendly resource to track the outbreak as it unfolds. JHU data sources: [WHO](https://urldefense.proofpoint.com/v2/url?u=https-3A__www.who.int_emergencies_diseases_novel-2Dcoronavirus-2D2019_situation-2Dreports&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=Kqo9h3kYNxWfxMkTXGNLn2mEPiaPYMNXyv6MBhwWxrQ&e=), [CDC](https://urldefense.proofpoint.com/v2/url?u=https-3A__www.cdc.gov_coronavirus_2019-2Dncov_index.html&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=h6IhY9XL94FOKx6QJnNFE7oopNAn8i_x1i6glDHWTd0&e=), [ECDC](https://urldefense.proofpoint.com/v2/url?u=https-3A__www.ecdc.europa.eu_en_geographical-2Ddistribution-2D2019-2Dncov-2Dcases&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=Pm_Cqp389q7TdLx5wiHK3ipHSMPkxYlP4MpWVmiDaR8&e=), [NHC](https://urldefense.proofpoint.com/v2/url?u=http-3A__www.nhc.gov.cn_xcs_yqtb_list-5Fgzbd.shtml&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=e7LXmeShPUvDK1Ofl1o1uSo7fXYbKYRF-g18i34BTZg&e=), [DXY](https://urldefense.proofpoint.com/v2/url?u=https-3A__3g.dxy.cn_newh5_view_pneumonia-3Fscene-3D2-26clicktime-3D1579582238-26enterid-3D1579582238-26from-3Dsinglemessage-26isappinstalled-3D0&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=U6k1aMt14xAlMWB0Ya70Al4RPKqCr6MrmTtiZCiDwqw&e=), [1point3acres](https://urldefense.proofpoint.com/v2/url?u=https-3A__coronavirus.1point3acres.com_&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=WHqaU2zyfaAFzvVxkWAzehvRt3PQPYgmuyvv0KIdFI8&e=), [Worldometers.info](https://urldefense.proofpoint.com/v2/url?u=https-3A__www.worldometers.info_coronavirus_&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=OJzaqtdTPXQ1WXJvLilAILEXHaGadR0EojNXRvpOcOM&e=), [BNO](https://urldefense.proofpoint.com/v2/url?u=https-3A__bnonews.com_index.php_2020_02_the-2Dlatest-2Dcoronavirus-2Dcases_&d=DwMFaQ&c=n6-cguzQvX_tUIrZOS_4Og&r=ZNGvwmzlj8kMBY6JiOVCuA&m=I4tRzMk97krELxTlEBHo8DfBIHceLgA3njBSxpFaSkY&s=65T1oGXacxbDM8RByJHBeJiBRBuHaaqyCsuda5ko-yk&e=), [the COVID Tracking Project](https://covidtracking.com/data/) (testing and hospitalizations), state and national government health departments, and local media reports.

**JHU DASHBOARD**: COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU) <https://coronavirus.jhu.edu/map.html>. To navigate the university’s publicly accessible data resources, refer to <https://coronavirus.jhu.edu/about/how-to-use-our-data>.

**NOTE:** After three years of around-the-clock tracking of COVID-19 data from around the world, Johns Hopkins **has discontinued the Coronavirus Resource Center’s operations**. The site’s two raw data repositories will remain accessible for information collected from **1/22/20 to 3/10/23** on cases, deaths, vaccines, testing and demographics.



**LAB PURPOSE AND LEARNING OBJECTIVES:** For this lab, we focus on **importing data**, **applying transformations, creating a model**, and using **DAX** to design a **report** (dashboard) **using a multidimensional question and data**.

**The Exercise**

**LAB NARRATIVE:** Based on the situation as described above, our task is to query (ask questions) of the JHU data to better understand what has happened, what might happen and what could happen regarding the COVID pandemic (descriptive, predictive, and prescriptive analysis). Are cases (confirmed vs deaths) on the increase, decrease or have we plateaued?

**NOTE:** Before we begin… let’s review the data sources (there are two for this lab, using three files):

1. <https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series>

We will use the time\_series\_covid\_confirmed\_global and time\_series\_covid\_deaths\_global. JHU stopped updating the “recovered” dataset in 2021; therefore, we will not use this resource.

1. <https://www.worldometers.info/world-population/population-by-country/>

## **LESSON 1: EXTRACT, TRANSFORM AND LOAD (ETL) THE DATA**

### Create a new **Power BI file**, **Get Data**, **Web**, insert the URL <https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_confirmed_global.csv> **OK**, **Transform Data** [[1]](#footnote-1)

### **Rename** the query **COV2020\_CSSE**

### **Use First Row as Headers**, then click on the **Add Column** tab, select **Custom Column**, **Custom column formula** **=”Confirmed”, OK**

### 

### Select **Custom, Province/State, Country/Region, Lat, Long** then **Unpivot Other Columns**

### **Rename Custom** to **Item** and **Attribute** to **Date**

### **NOTE:** Match the case exactly as the append we will use later is case sensitive

### Change **datatype** for **Item** to **Text, Date** to **Date,** and **Value** to **Whole Number** (if this is not the default)

### Select **New Source** (Home tab), **Web,** insert URL<https://raw.githubusercontent.com/CSSEGISandData/COVID-19/master/csse_covid_19_data/csse_covid_19_time_series/time_series_covid19_deaths_global.csv> **OK**

### **Repeat steps 3-6** above, except for the **Custom Column** insert **Custom Column formula** **=”Deaths”**

### Select the **COV2020\_CSSE query**, click **Append Queries** (Home tab),select **Two tables,** in the **Table to Append** option, select **time\_series\_covid19\_deaths\_global** table,click **OK**

### 

### To confirm that the Append was successful, use the **Item Column** to filter the results (click on **Load more** to review all possible results… Confirmed and Deaths)

### 

### Under the Queries pane, right click **time\_series\_covid19\_deaths\_global** query, **unselect Enable load** (since we appended the data to the COV2020\_CSSE query, we don’t need to view this data in the Report view, but we can’t delete it since we need to refresh the data each class to import new records)

### Select **New Source** (Home tab), **Web**, insert the URL <https://www.worldometers.info/world-population/population-by-country/> , **OK**

1. Under the display options, there may be several selections, click on **Table 1** under **HTML Tables** (this choice should give us the column names as headers as well as the data), **OK**
2. **Rename** the query **World Population**
3. **Remove** the **Fert. Rate** and the **Med. Age** columns (both have errors, and we will not use the data)
4. **Rename** the column **Country (or dependency)** to **Country**

### In the **World Population** query some **Countries** are not documented the same as those in the COV2020\_CSSE query, such as United States, it’s US in the COV2020\_CSSE

### In the World Population query, **right click United States,** select **Replace Values… Replace With US,** click **OK**

### **Close and Apply**

### **Save** your Power BI file as **DIAD LAB 1**

## **LESSON 2: THE DATA MODEL AND (SOME) DAX**

### Go to the **Model** viewto create a relationship between **Country/Region (COV2020\_CSSE) and Country (World Population)**

### 

### Go to the **Data** view, under the **COV2020\_CSSE** table, select **Lat** change **Data category** (under Column tools tab) to **Latitude**, then select **Long** and change **Data category** to **Longitude** (we won’t use these fields for this lab, but in subsequent labs, so we need them to “behave” properly)

### Also, select **Lat** and **Long**, set **Summarization** to **Don’t Summarize** (we will never do “math” on these fields)

### Add several DAX measures (see the [Appendix](#_ADDENDUM) for DAX definition… more than what was provided in INFO 3300)

### **Right click** on the **COV2020\_CSSE** table, select **New measure**, in the formula bar above the dashboard below the ribbons, copy/paste the following **two DAX measures** (each one is a separate measure):

Confirmed Global = CALCULATE(SUM(COV2020\_CSSE[Value]), (COV2020\_CSSE[Item]="Confirmed"))

Confirmed Recent Value Global = CALCULATE(SUM(COV2020\_CSSE[Value]), (COV2020\_CSSE[Item]="Confirmed"), FILTER(COV2020\_CSSE, COV2020\_CSSE[Date] = MAX(COV2020\_CSSE[Date])))

**CALCULATE()** is the **most important** and **powerful function** in **DAX**. It is the only function that can modify the filter context coming from your visuals. CALCULATE() modifies an expression (which can be a measure or another DAX formula) by applying/removing/modifying filters.

**SIDEBAR:** We can write **DAX** as **formulas** or **expressions**. The main difference is that **expressions** typically use **variables** and formulas don’t. For example, the first DAX formulas above can be written as expressions as:

Confirmed Global v2 =

VAR ConfirmedValues =   
CALCULATETABLE(

    FILTER(COV2020\_CSSE, COV2020\_CSSE[Item] = "Confirmed"),

    COV2020\_CSSE[Item] = "Confirmed"

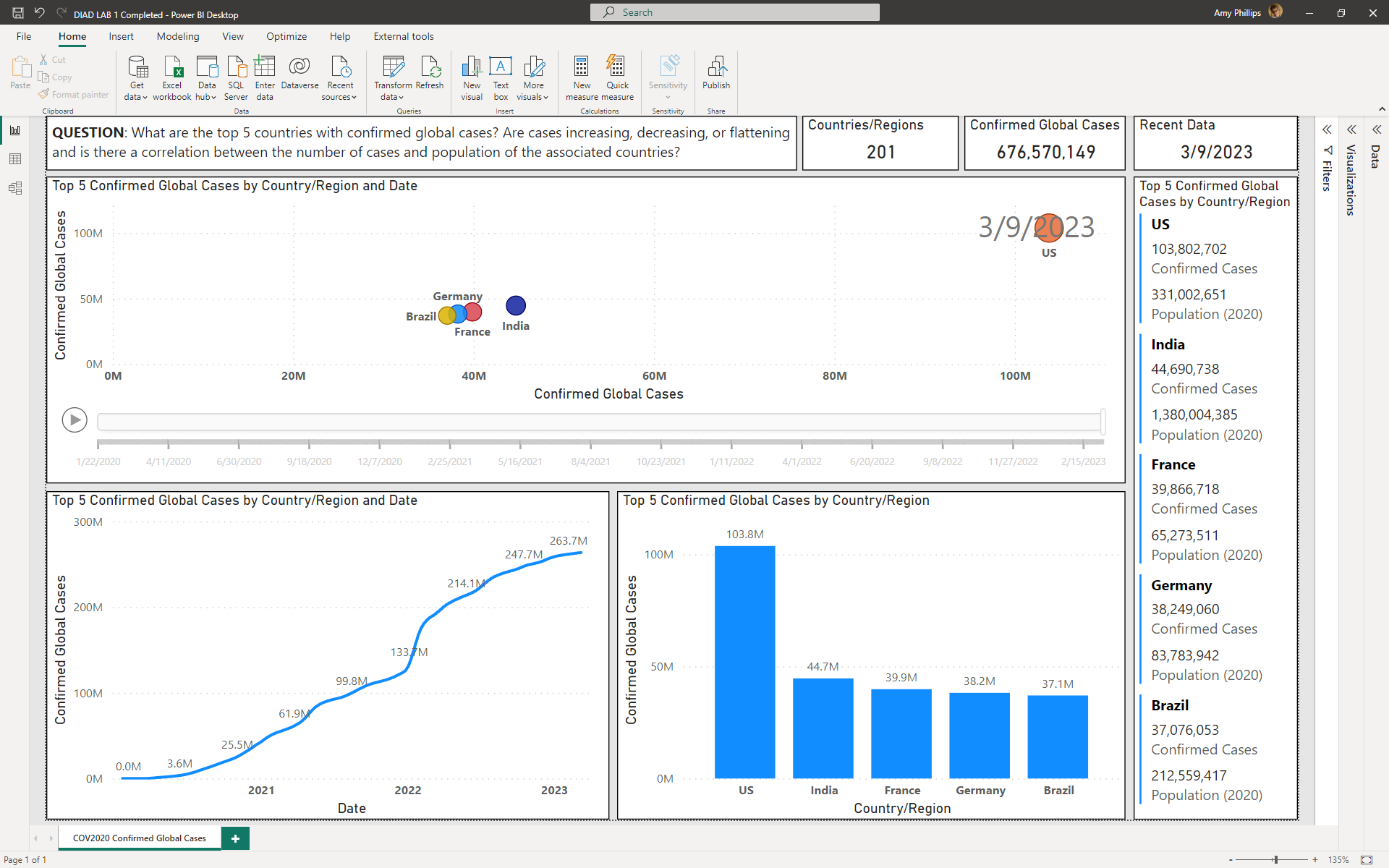
)

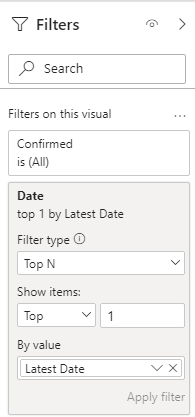
RETURN

    SUMX(ConfirmedValues, COV2020\_CSSE[Value])

## **LESSON 3: DESIGN A REPORT**

1. Go to the **Report** view, we will **try to replicate:**



1. **Start with a question** (include as a textbox at the top of the dashboard)
   1. **QUESTION**: What are the top 5 countries with confirmed global COVID cases? Are cases increasing, decreasing, or flattening and is there a correlation between the number of cases and population of the associated countries?
2. Create a **Card** (visualization) using **Country/Region**
   1. Make sure **Count(Distinct)** is selected in the Fields under the Visualizations pane for **Country/Region**
3. Create a **Card** (visualization) using **Confirmed Global** (DAX measure created above, located under the fields)
   1. Using the **Filters** pane: Drag **Date** to a Filter, set **Filter type to Top N**, set Show items to **Top** and **1**, and **By value** to **Latest Date** (see Figure on right)
   2. **NOTE:** We could have used Confirmed Recent Value Global instead to avoid using a filter, but this will not work for all visuals in the coming labs (good practice to use the filter option)
4. Create a **Card** (visualization) using **Date**
   1. Under the Visualization pane,in the **Fields** section**,** use the dropdown to **Date,** select **Latest Date**
5. Create a **Scatter chart** (visualization) using **Country/Region** (Legend), **Confirmed Global** (X Axis), **Confirmed Global** (Y Axis), **Confirmed Global** (Size) and **Date** (Play Axis)
   1. Using the **Filters** pane: Change **Country/Region** to use **Top 5** by value “**Confirmed Global**”
   2. Using the **Format** option (in the Visualizations pane): Visual tab, Turn the **Legend= Off**, **Category label = On**
6. Create a **Multi-row card** (visualization) using **Country/Region, Confirmed Recent Value Global,** and **Population 2020** (from World Population), put Fields in that order
   1. Using the **Filters** pane: Change **Country/Region** to use **Top 5** by value “**Confirmed Recent Value Global**”
   2. **Sort** the Multi-row card **by Confirmed Recent Value Global** in **descending** order
   3. Select **Confirmed Recent Value Global** under the fields pane, display using a “**,**” for thousands separator
7. Add a **Line chart** (visualizations) using **Date** (X-axis), **Confirmed Global** (Y-axis)
   1. Using the **Filters** pane: Drag **Country/Region** to **Filters on this visual**, to use **Top 5** by value “**Confirmed Global**”
8. Add a **Stacked Column Chart** (visualizations) using **Country/Region** (X-axis)**, Confirmed Recent Value** **Global** (Y-axis)
   1. Using the **Filters** pane: Change **Country/Region** to use **Top 5** by value “**Confirmed Recent Value Global**”
9. **NOTE:** Please pay special attention regarding **how visuals are positioned/sized using all available screen real estate** (review screenshot above regarding the expected dashboard layout)
10. **Rename** the report/page **COV2020 Confirmed Global Cases**
11. **Save** the Power BI file

## **APPENDIX**

## **What Is DAX?[[2]](#footnote-2)**

**NOTE: Data Analysis Expressions (DAX)** is both a query and functional language. It made its first appearance back in 2009 as part of an add-in to Microsoft Excel 2010 (for use with Power Pivot). The primary objective of DAX is to help organize, analyze, understand, and enhance data for analytics and reporting. DAX is not a full-blown programing language and does not provide some of the flow-control or state-persistence mechanisms you might expect from other programming languages, such as Java or Python. It has been designed to enhance data modeling, reporting, and analytics. DAX is constantly evolving with new functions being added on a regular basis.

Microsoft specifically developed DAX to support a large user base. DAX is much easier to learn than traditional technical languages, making it an ideal language for users who don’ t come from a technical background but want to do their own Self - Service Business Intelligence. DAX is often compared to an advanced version of Excel, having high-end capability of managing and manipulating data. Many DAX functions are similar to functions in Excel, which means new users can leverage their existing knowledge to make an easy transit ion to writing and authoring DAX formulas. While there are similarities between DAX and Excel, the two languages are not interchangeable.

DAX is described as a functional language, which means calculations primarily use functions to generate results. A wide variety of functions are provided to help with arithmetic, string manipulation, date and time handling, and more. Functions can be nested, but you cannot create your own.

There are over 200 functions in DAX. Every calculation you write will use one or more of these. Each function produces an output with some returning a single value and others returning a table. Functions use parameters as input. Functions can be nested so the output of one function can be used as input to another function.

The two **most common** uses for DAX are calculated columns and calculated measures. **Calculated columns** provide one value for each **row encountered; calculated** **measures** deal with **multiple rows** at the same time and often provide some form of aggregation.

## **Calculated Columns**

With calculated columns, the DAX expression used by the calculation is only executed when data for the table is refreshed. The expression is calculated once for every row in the table, and the single value generated by the calculation is stored as the value in the column for that row and it cannot be changed. It’s as if the value generated by the calculated column existed in the source data that was used to import for the table.

Calculated columns are row based, meaning they can quickly perform calculations if all the information required is contained within the same row. Calculations can still use data from rows other than the current row.

## **Calculated Measures**

With calculated measures, the DAX expression could be executed every time a page loads or when a user changes a filter or slicer selection. The DAX expression contained within the calculated measure is executed once for every value that uses it in a report. If a calculated measure is used in a table or matrix visual, it is executed as many times as there are cells in the grid that use it to show a value. When one is used on a line chart visual with 100 points in a series, it executes 100 times for each point, with each execution having a slightly different filter context.

Calculated measures are dynamic and respond to user interaction. They recalculate quickly and often but do not store output in the data model, so they have no impact on the physical size of the data model. Increasing the number of calculated measures in your data model has no impact on speed or size of the model at rest.

Calculated measures are quick to process calculations using a single column over many rows, but you can still write them in a way that allows row-based data processing using iterators.

**NOTE:** It is generally considered best practice to create new columns in your data model in the original source or in the Power Query editor, before the data is loaded in Power BI Desktop. This gives the user the best possible compression for the data.

|  |  |
| --- | --- |
| **COLUMNS vs MEASURES** | |
| The results of calculated columns are immediately viewable in the table. | The results of calculated measures are not stored in the table. |
| Calculated columns can be used in filters and slicers. | Calculated measures cannot be used in slicers. |
| Calculated columns are updated when the data model is refreshed. | Calculated measures are dynamic and calculated as filters are applied. |
| Calculated columns take up more space in the data model. | Calculated measures do not take up space in the model. |
| Calculated columns are not dynamic. | Calculated measures are dynamic and always changing based on the filters applied. |

1. **NOTE:** The data in the csv files aggregates for each date recorded; therefore, normal aggregates (e.g., SUM, MIN, MAX, AVG, etc. measures) will not work. Using links provided to the data will allow us to “refresh” data to pull in the “the most current dataset” (the GitHub update frequency is once a day around 11:59pm UTC). [↑](#footnote-ref-1)
2. DAX References: <https://dax.guide/> and <https://docs.microsoft.com/en-us/dax/> [↑](#footnote-ref-2)