

COVID-19 Situation in Europe Report Project

I. Concept of the Project

This project depicts COVID-19 Situation in Europe using dataset from the ECDC website. Using the reference source data and multiple cloud service from Microsoft Azure to gain some hand-on experiences about ETL process (extract, transform, load) and also to get familiar with various tools/platforms provided by Azure e.g Azure Data Factory, Azure Databricks,... The primary objective is visualising the key point of the whole acquired dataset, in order to comprehensively understand the influence of COVID-19 on the entirety of European Region throughout the year 2020.

II. Task

1. Ingest data from multiple source, clean it up, make sensible transformation to be suitable for the goal.
2. Load the processed data into central repository, such as data warehouse and datalake.
3. Using Power BI to access data storage and make a representation about confirmed cases and deaths, hospital and ICU occupancy rate

III. Resource

1. Sources data
 - The resources for the dataset and the main concept of these project is based on “Udemy Course – Azure Data Factory For Data Engineers – Project on Covid-19 by Ramesh Retnasamy” under the following link:
<https://github.com/cloudboxacademy/covid19>
 - The acquisitive dataset is categorized like following:
 - eurostat_data: contain population

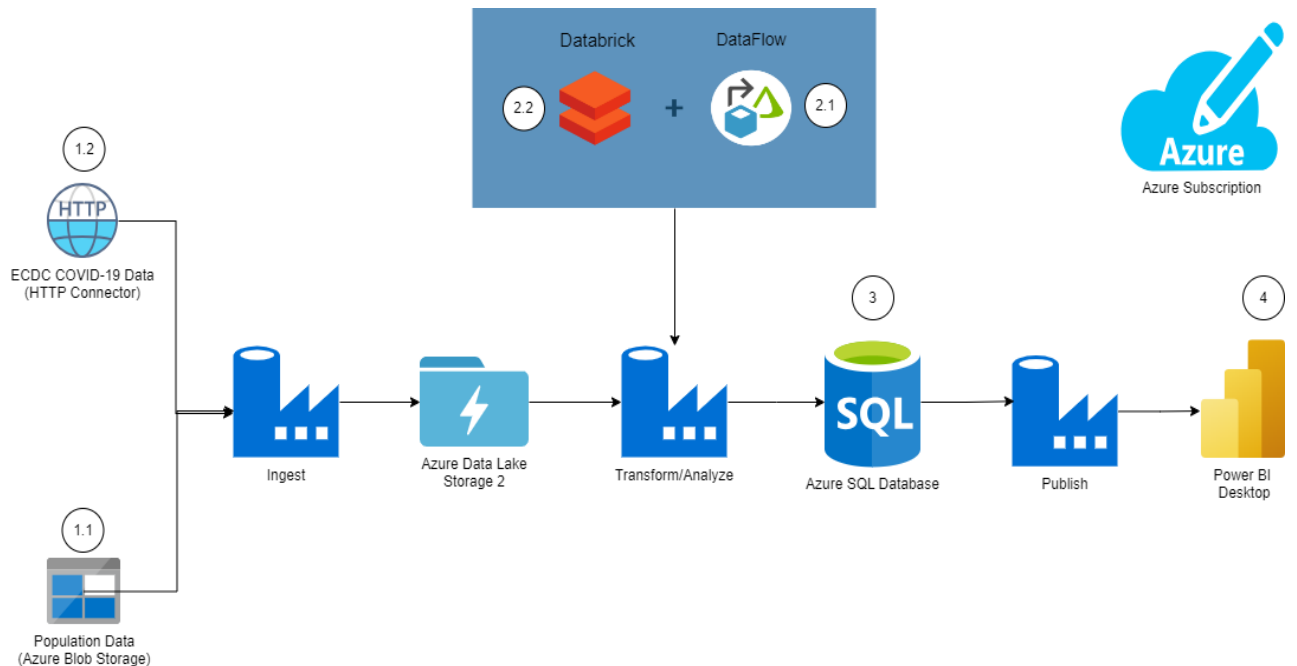
- ecdc
 - cases_deaths.csv: timeline of number of case and death in each country
 - country_response.csv: timeline of how each country response to the situation
 - hospital_admissions.csv: timeline of daily and weekly hospital occupancy in each country
 - testing.csv: COVID test-situation in each country
- look_up:
 - country_lookup.csv: geometric information of each country
 - dim_data.csv: derived information of reported date

2. Tools

- Environment Setup
 - Azure Subscription
- Data Integration/Ingestion
 - ADF Data Flows within Data Factory
- Transformation
 - Data Flows within Data Factory
 - Azure Databricks Cluster
- Data Storage
 - Azure SQL Database
 - Azure Blob Storage Account
 - Azure Data Lake Storage Gen 2
- Visualization
 - Power BI Desktop

IV. Approach

Solution Architecture Overview



1. Data Extraction/Ingestion

Four different datasets were ingested from the link mention in Section III.1.

These are:

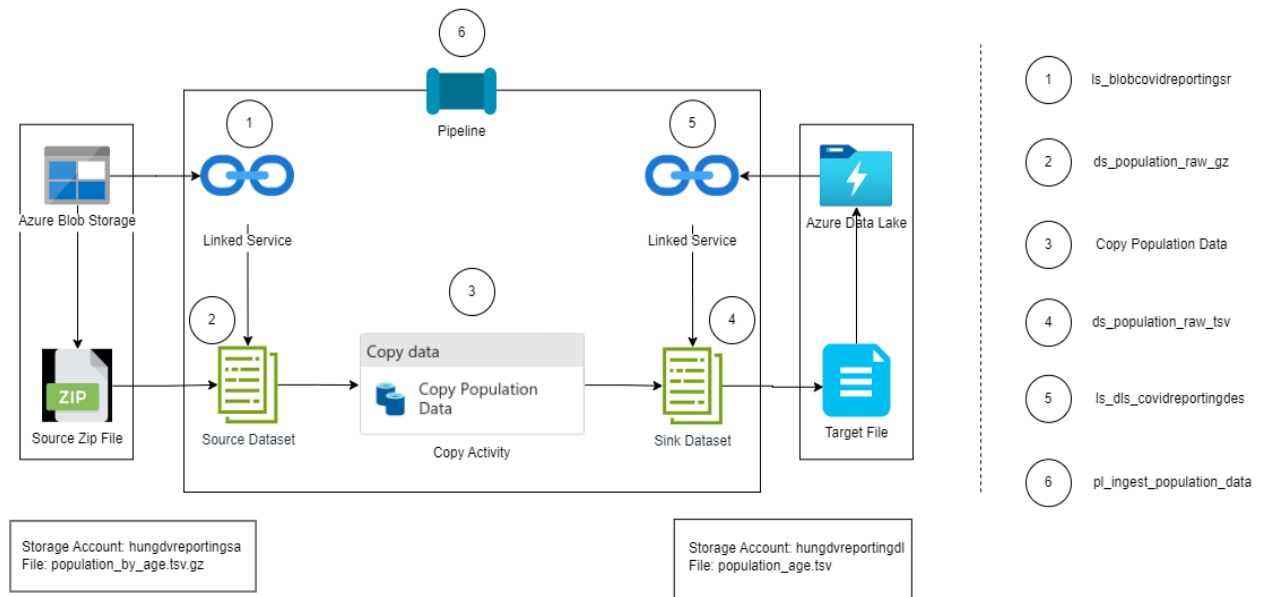
- Cases and Deaths Data
- Hospital Admissions Data
- Population Data
- Test Conducted Data

To gain some hand-on experiences, I used various components of ADF Pipeline activities to ingest data both from HTTP Data Source and Azure Storage Account to Azure DataLake. These are

- Validation Activity
- Get Metadata Activity
- Copy Activity

Step 1.1 Population Data: Load into Storage Account and move it to Destination Data Lake

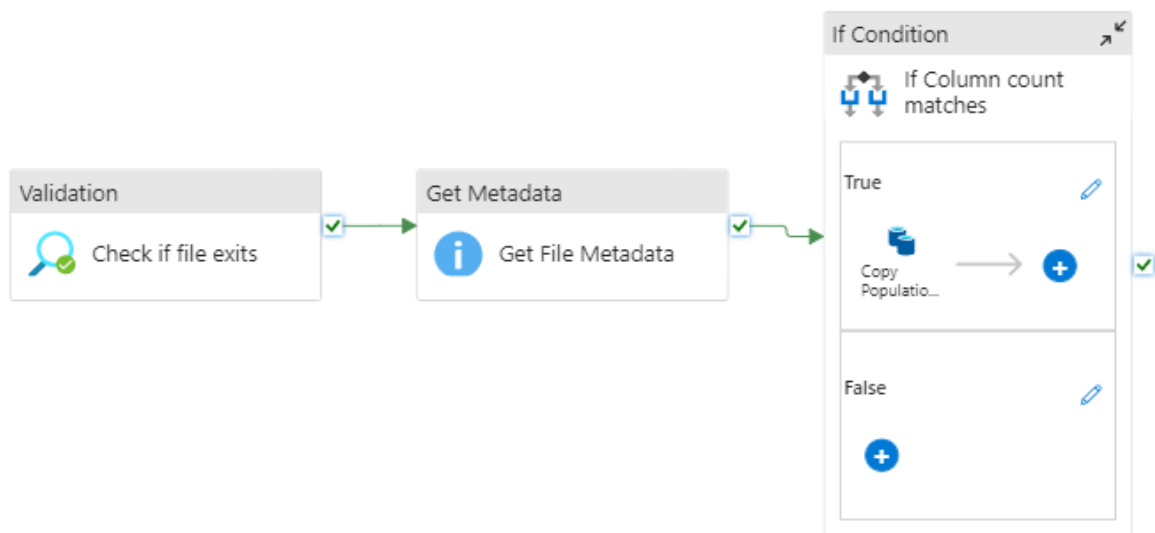
Solution Flow



Process

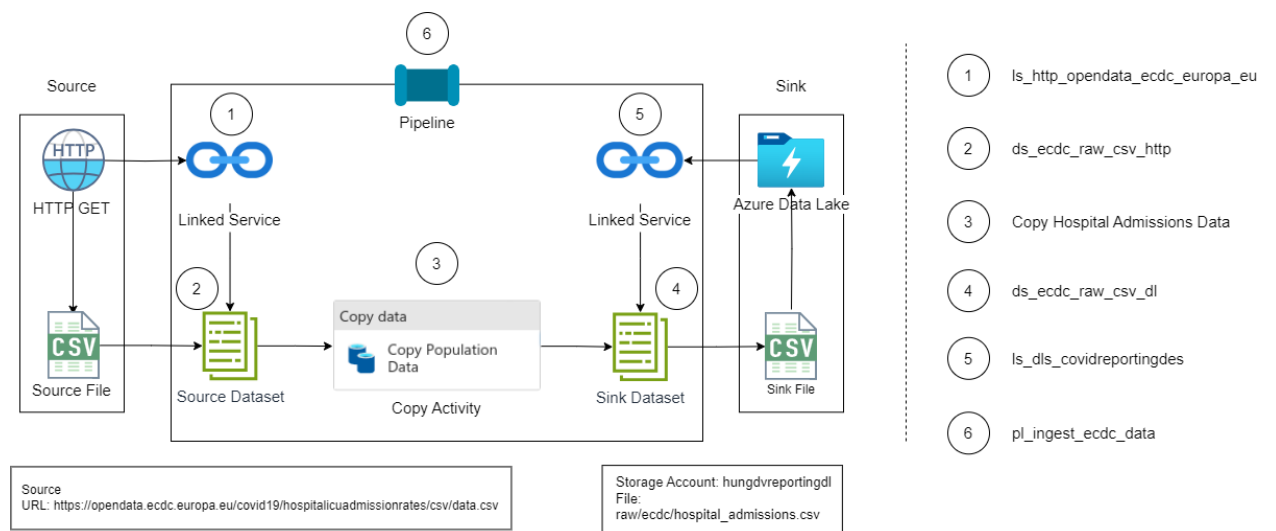
- (1) Create Linked Service to Azure Blob Storage
- (2) Create Source Data Set
- (3) Execute Copy Activity when file becomes available -> Check Metadata about number of column before loading data using IF condition -> Load Data into Target Destination
- (4) Create Linked Service to Azure Lake Storage (GEN2)
- (5) Create Sink Data Set
- (6) Create Pipeline and Schedule Trigger

Pipeline Design



Step 1.2 ECDC Data from Web to Destination Data Lake

Solution Flow

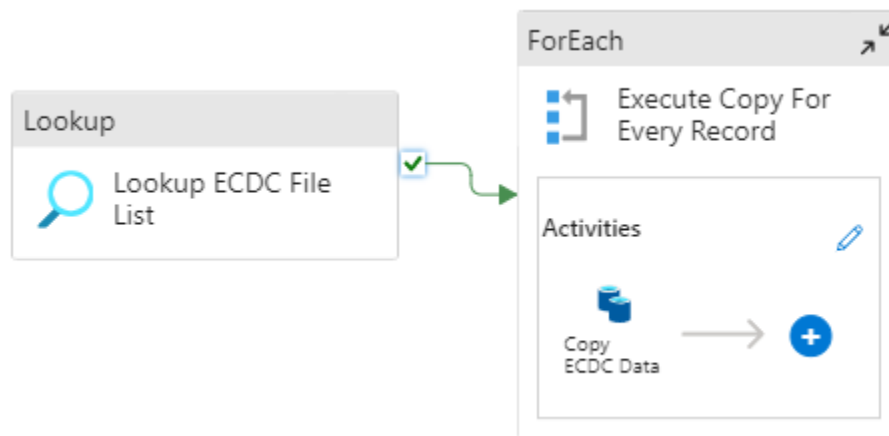


Process

- (1) Create Linked Service using HTTP GET method
- (2) Create Source Data Set

- (3) Look up to get all the parameters from json config file to get 4 files of ECDC Data Content
- (4) Create Sink Dataset
- (5) Create Linked Service to Azure Data Lake Storage (GEN2)
- (6) Create Pipeline and schedule Trigger to recursive trigger pipeline every 24 hours

Pipeline Design



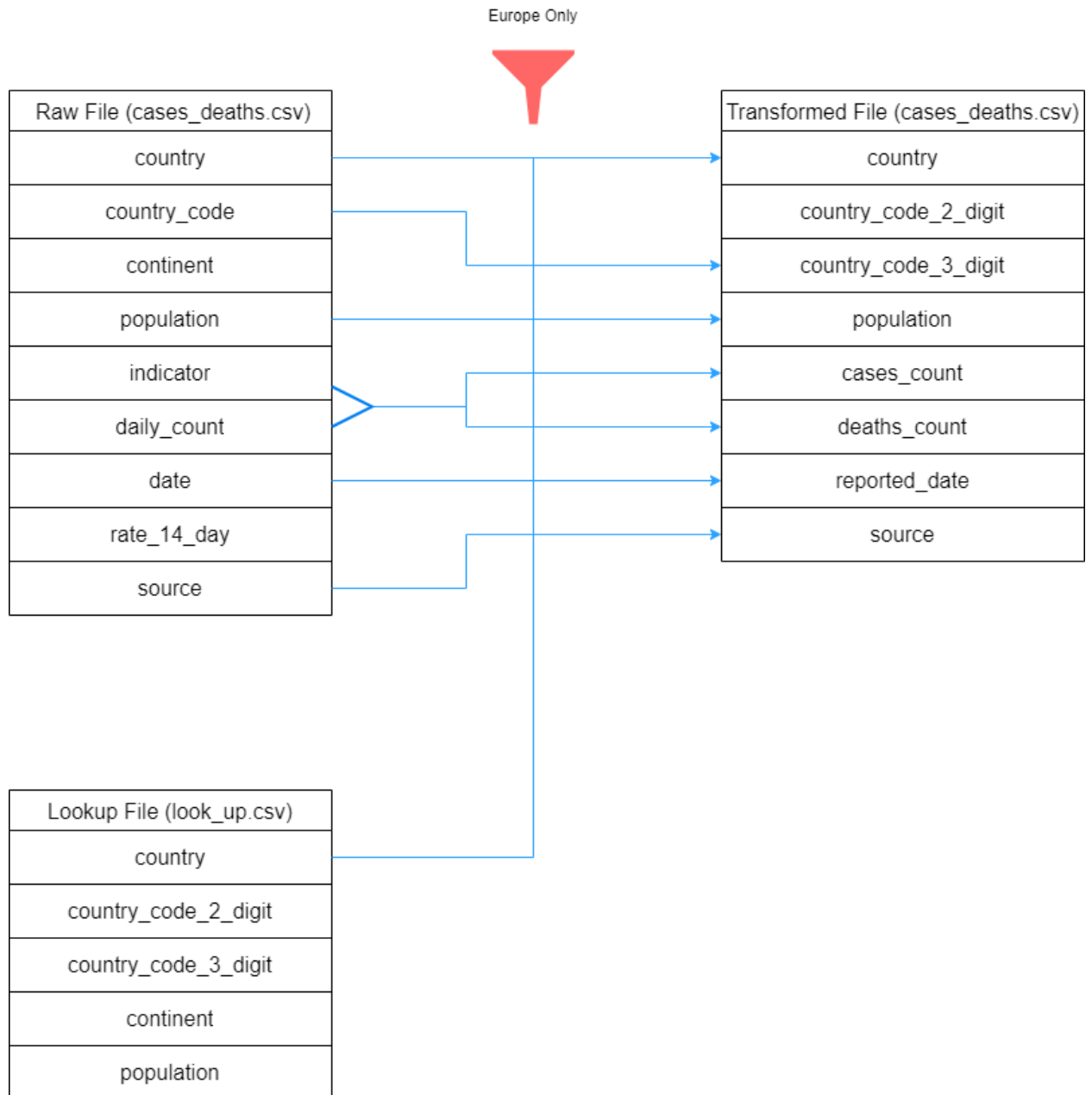
2. Data Transformation

Step 2.2 “Cases and Deaths” and “Hospital Admissions” data were transformed using ADF Data Flows

The transformation used on both dataset include following method: select, lookup, filter, join, sort, conditional split, derived columns.

Data Flows Transformation Cases & Deaths Data

Solution Flow



Process

- Ingest Cases and Deaths Source (Azure Data Lake Storage Gen2)
- Filter data from Europe-Only country
- Filter only selected columns
- Pivot Count only "indicator" column and get the sum of daily confirmed cases and death count

- Lookup Country to get country_code_2_digit (Country Code in 2 digits) and country_code_3_digit (Country Code in 3 digits) (this steps is unnecessary, just to practice on several transformation method)
- Filter only selected columns for the Sink
- Create Sink Dataset (Azure Data Lake Storage Gen2)
- Create Pipeline and schedule Trigger to recursive trigger pipeline for every 24 hours

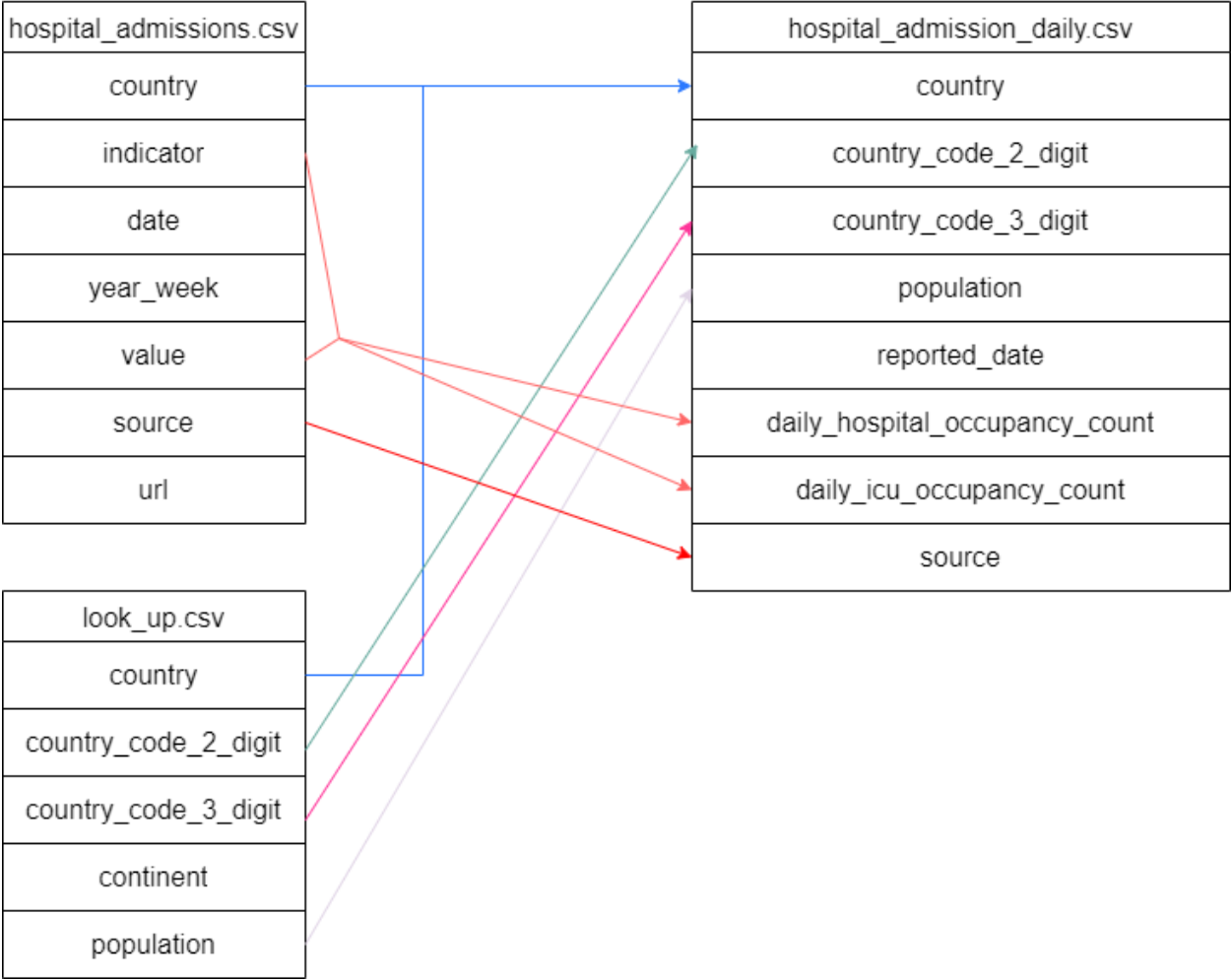
Data Flows Transformation Hospital Admissions Data

Solution Flow

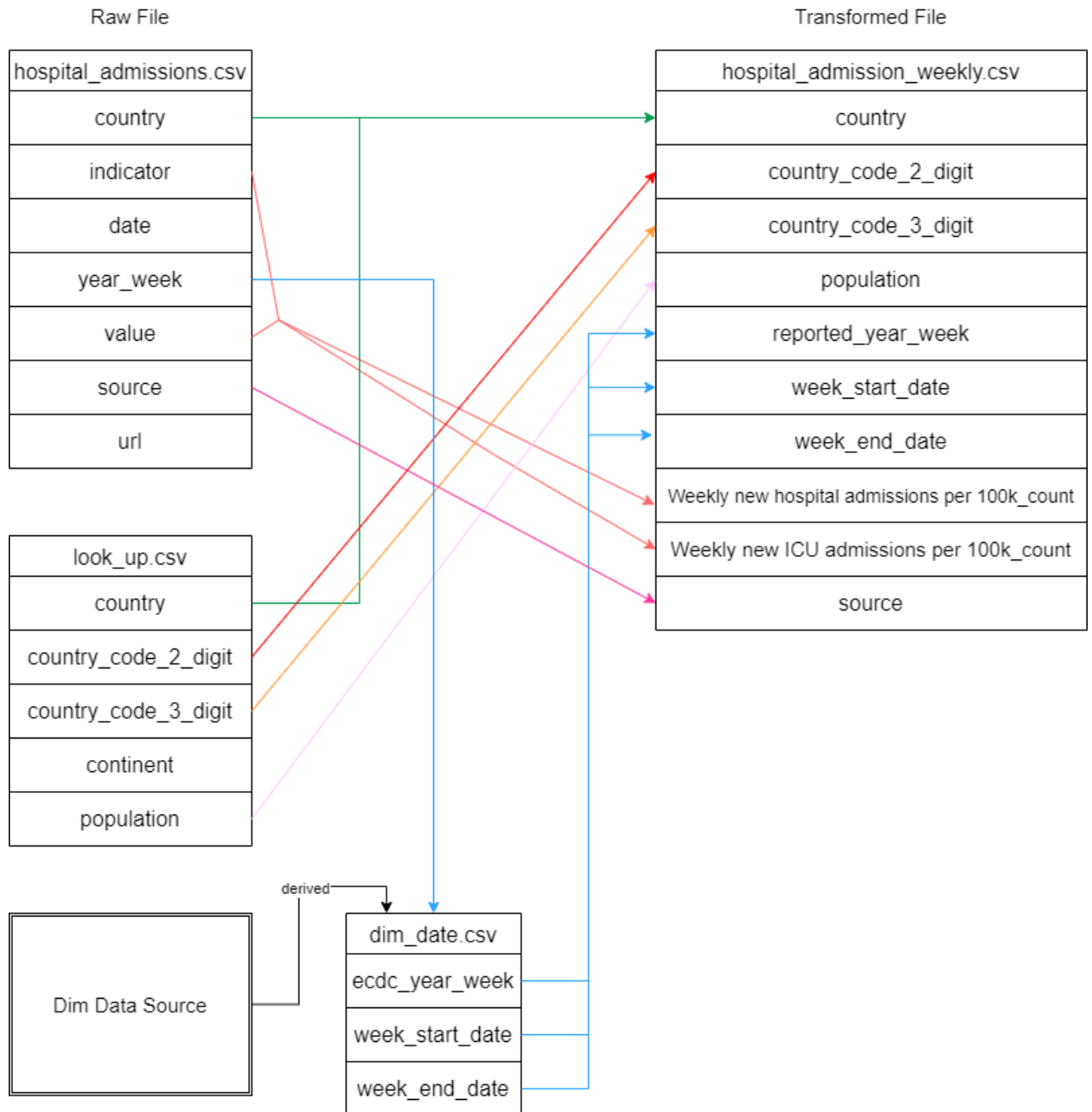
Hospital Admissions Daily Data

Raw File

Transformed File



Hospital Admissions Weekly Data



Process

- Hospital Admissions Source (Azure Data Lake Storage Gen2)
- Filter only selected columns
- Lookup Country to get "country_code_2_digit" (Country code in 2 digits), "country_code_3_digit" (Country code in 3 digits) columns

- Filter out duplicated columns
- Filter rows using Condition Split Weekly, Daily condition
 - `indicator=="Weekly new hospital admissions per 100k" || indicator=="Weekly new ICU admissions per 100k"`
 - `indicator=="Daily hospital occupancy" || indicator=="Daily ICU occupancy"`

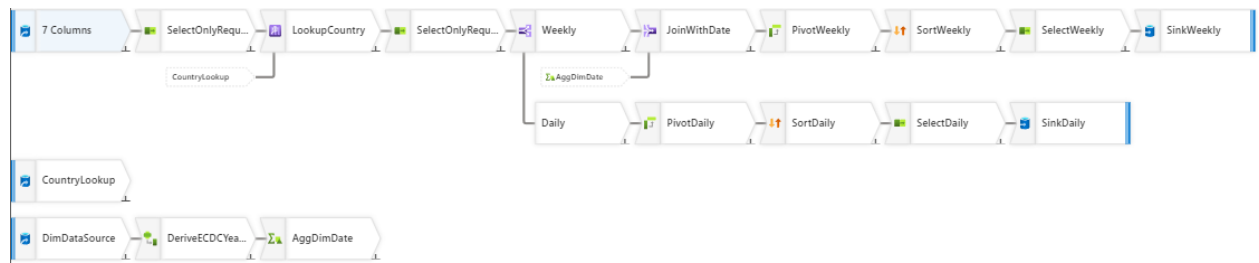
For Weekly Path

- Join with Date to get "ecdc_year_week", "week_start_date", "week_end_date"
- Pivot Count only "indicator" column and get the sum of weekly hospital & ICU admissions per 100k
- Sort data using "reported_year_week" and "country" in respectively ascending and descending order
- Filter selected columns for Sink
- Create Sink Dataset (Azure Data Lake Storage Gen 2)
- Schedule Trigger to recursive trigger pipeline every 24 hours

For Daily Path

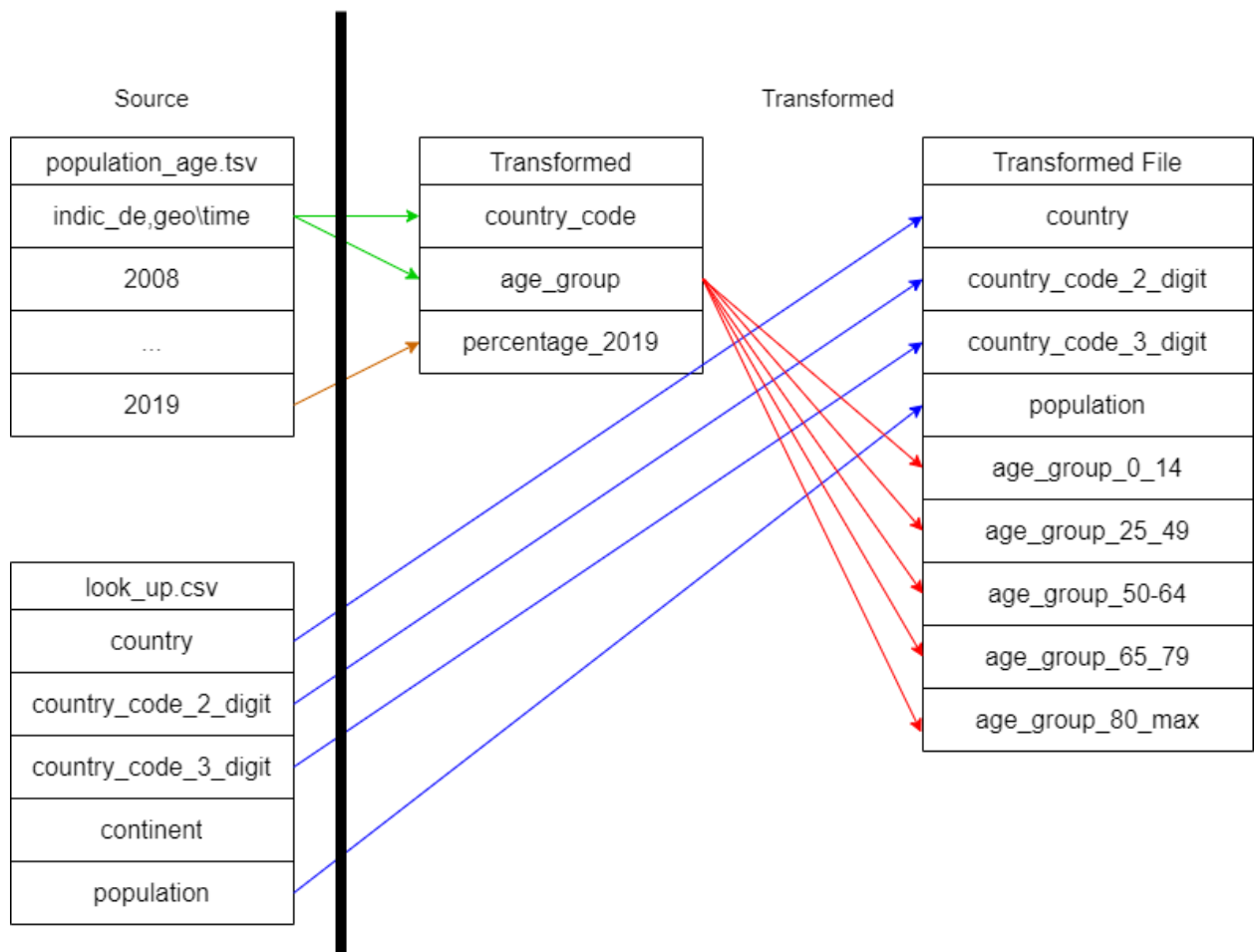
- Pivot Count only "indicator" column and get the sum of daily hospital & ICU admissions per 100k
- Sort data using "reported_date" and "country" in respectively descending and ascending order
- Filter selected columns for Sink
- Create Sink Dataset (Azure Data Lake Storage Gen 2)
- Schedule Trigger to recursive trigger pipeline every 24 hours

Data Flows Transformation



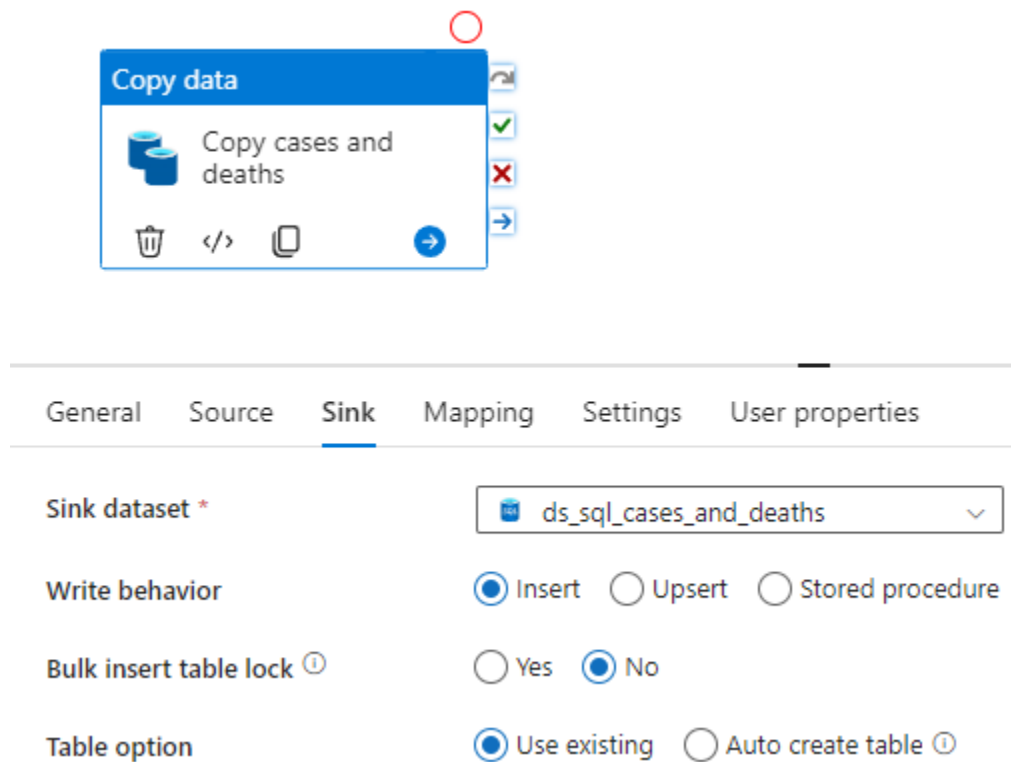
Step 2.3 Databricks Transformation on “Population by Age” File

Solution Flow



3. Store Data in Azure SQL Database

Create pipeline to copy Cases & Deaths Data to SQL Database



Copy data

Copy cases and deaths

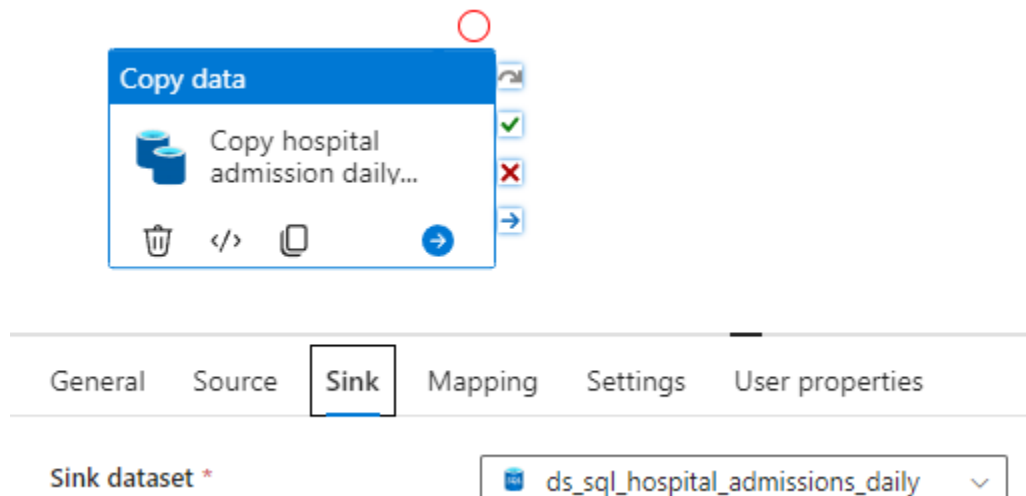
Sink dataset * ds_sql_cases_and_deaths

Write behavior ☒ Insert ☐ Upsert ☐ Stored procedure

Bulk insert table lock ☐ Yes ☒ No

Table option ☒ Use existing ☐ Auto create table

Create pipeline to copy Hospital Admissions Daily Data to SQL Database



Copy data

Copy hospital admission daily...

Sink dataset * ds_sql_hospital_admissions_daily

General Source **Sink** Mapping Settings User properties

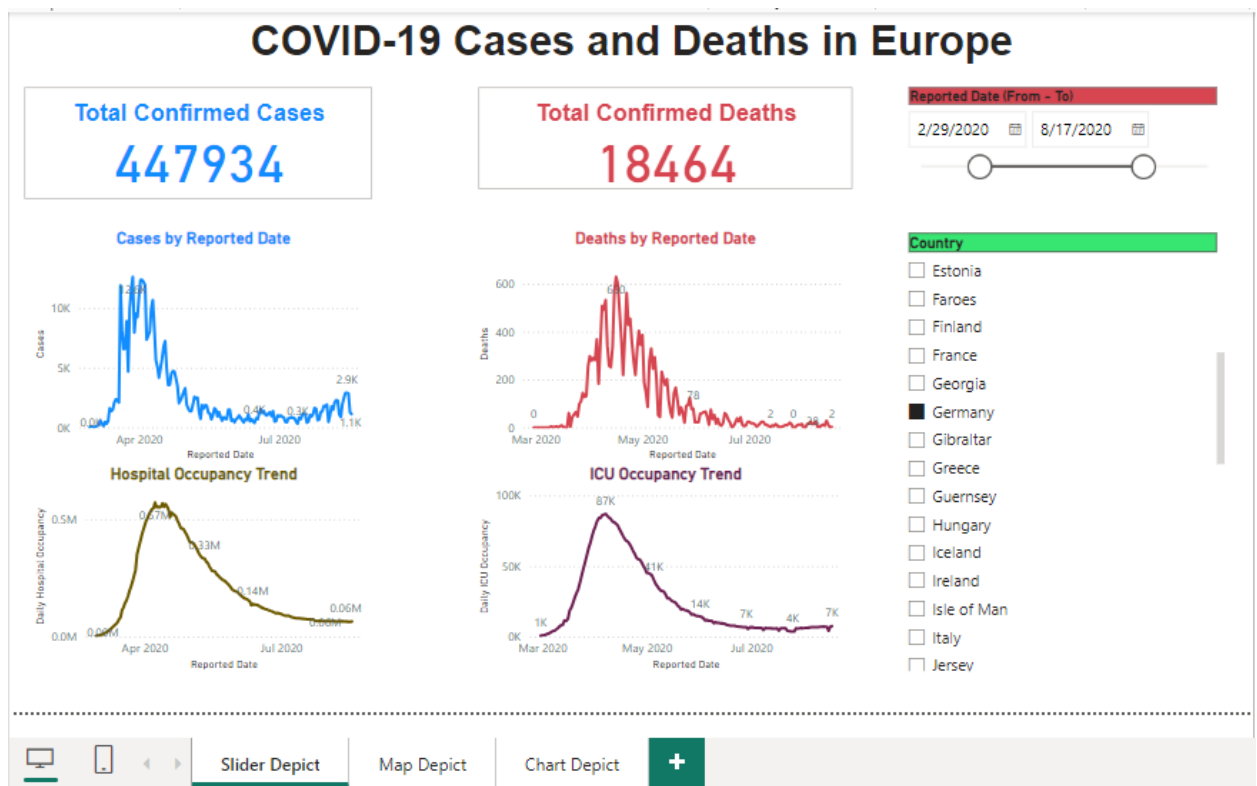
The same goes for Hospital Admissions Weekly Data

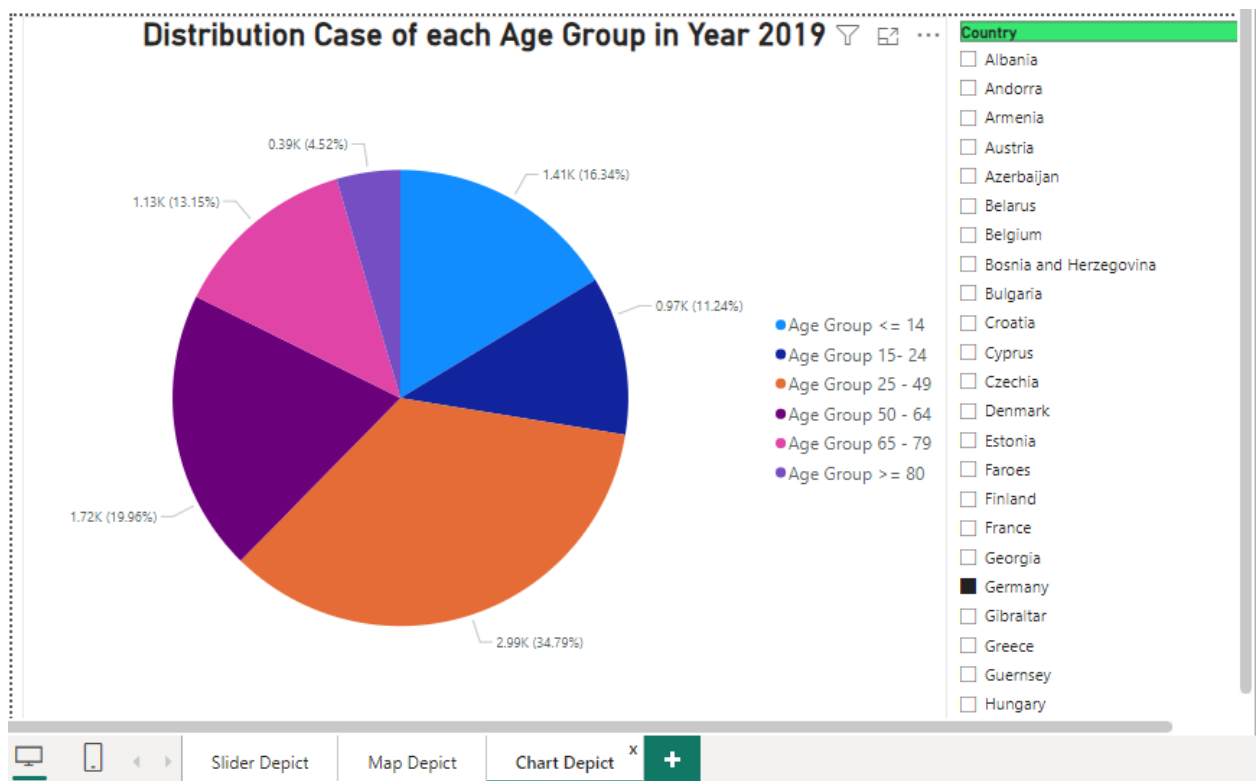
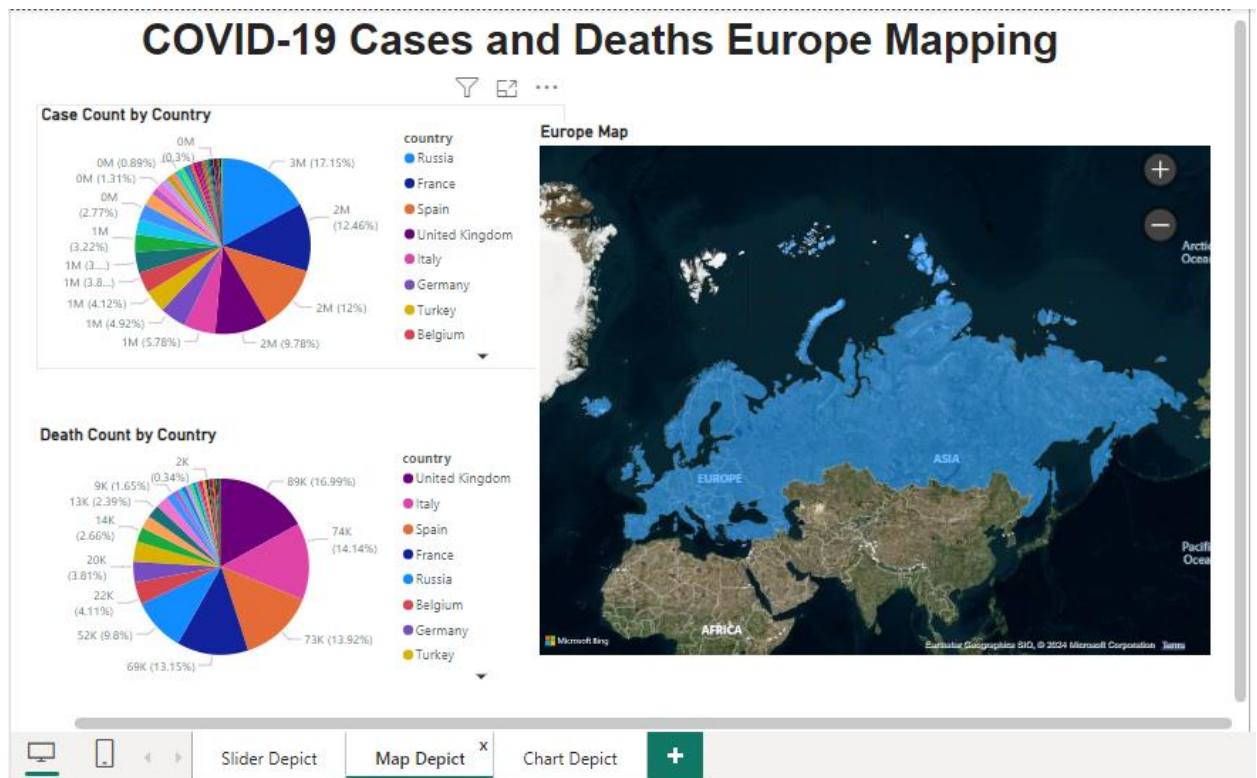
4. Visualize Presentation using PowerBI Desktop

Solution Flow

- Create a connection from Azure SQL Database to PowerBI and load the data
- Visualize the total confirm Cases and Deaths Year 2020 in Slider Depict
- Visualize the total confirm Cases and Deaths Year 2020 in Map Depict
- Visualize the distribution Case of each Age Group Year 2019 in Chart Depict

Process





All depicts can be interactive (e.g: click on specific country to get corresponding data)

V. Conclusion

This project contains almost every basic step of ETL progress and also covers some use-cases of Azure Technology Stack, including: Azure Data Factory (with the using of Blob and Data Lake Storage, Data Flows for various transformation methods), Azure Databricks and visualization tool like PowerBI.