



AZURE DATA FACTORY CAPSTONE – COVID USE CASE

Emp Id: 2319933
Name: Bandi Yuktha Reddy
Cohort Id: CSDAIA24AZ002

CONTENTS

Project Architecture Flow
Resource Requirements
 Project work flow
 Data Verification
 Project Outcome

INTRODUCTION

In the wake of the global COVID-19 pandemic, the need for reliable data integration and analysis has become paramount. This capstone project leverages Azure Data Factory (ADF), a cloud-based data integration service, to orchestrate and automate the movement and transformation of COVID-19 data. The project aims to provide insightful analytics that can aid in understanding the spread of the virus, its impact on healthcare systems, and the effectiveness of public health interventions.

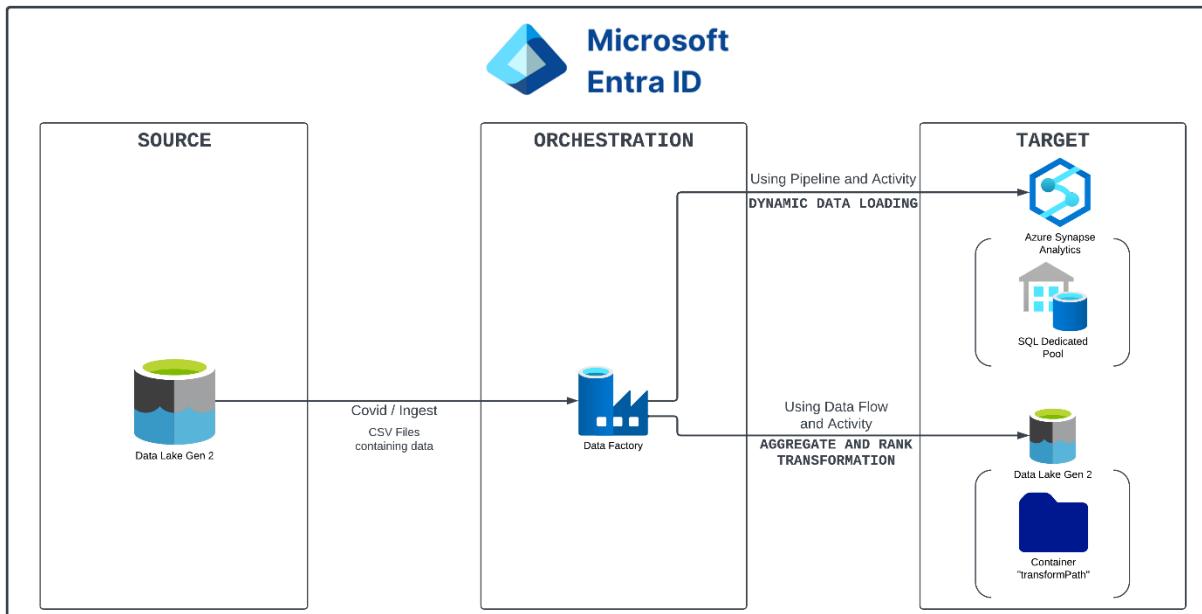
Utilizing ADF's robust capabilities, we will create a pipeline that sources up-to-date COVID-19 data from various repositories, processes it through custom transformations, and loads it into a centralized data store for analysis. This end-to-end solution will enable stakeholders to make informed decisions based on the latest trends and statistics of the pandemic.

This introduction sets the stage for the project, outlining its purpose and the technology used. It can be further expanded to detail specific objectives, data sources, and expected outcomes.

The purpose of the Covid use case exercise is to learn how to build a real-world data pipeline in Azure Data Factory (ADF) to analyze the covid trend across the regions using Azure cloud data services. By performing this case study, you will learn.

- How to ingest data from flat files into Azure Data Lake Gen2 and Azure Synapse using Azure Data Factory (ADF)
- How to transform data using Data Flows in Azure Data Factory (ADF) and load into Azure Synapse.

PROJECT ARCHITECTURE FLOW



RESOURCE REQUIREMENTS

Azure Service	Azure Service Name
Resource Group	covid-rg
Azure Data Lake Storage Gen2 Account	Covid-adls
Azure Data Lake Storage Gen2 Container	Covid
Azure Synapse Workspace	Covid-Synapse-Workplace
Azure Data Factory	Covid-ADF

PROJECT WORKFLOW

- This project consists of two requirements.

Procedure for Requirement 1:

Step1: Create a Resource group(covid-rg2319933).

The screenshot shows the Microsoft Azure portal interface. The user is creating a new resource group named 'covid-rg2319933'. The 'Review + create' tab is selected. A green validation message at the top states 'Validation passed.' The 'Basics' section shows the subscription is 'Azure for Students', the resource group name is 'covid-rg2319933', and the region is 'East US'. The 'Tags' section shows the owner is 'yukthareddy'. At the bottom, there are buttons for 'Create', '< Previous', 'Next >', and 'Download a template for automation'. The taskbar at the bottom of the screen shows various pinned icons and the date and time as 3/8/2024 12:13 PM.

Step2: Create an Azure Datalake storage (covidadls2319933).

The screenshot shows the Microsoft Azure portal interface. The user is creating a new storage account named 'covidadls2319933'. The 'Properties' tab is selected. The 'Overview' section shows the resource group is 'covid-rg2319933', location is 'eastus', and the account kind is 'StorageV2 (general purpose v2)'. Other details include 'Standard' performance level, 'Locally-redundant storage (LRS)' replication, and 'Succeeded' provisioning state. The 'Disk state' is 'Available'. The 'Created' date is 3/8/2024, 12:23:20 PM. The taskbar at the bottom of the screen shows various pinned icons and the date and time as 3/13/2024 9:12 AM.

Step3: Create a container (covid) within adls.

The screenshot shows the Microsoft Azure portal interface. The top navigation bar has tabs for 'Be.Cognizant - Home', 'covid - Microsoft Azure', and 'Microsoft Azure Sponsorships'. The main content area is titled 'covid' under 'Containers'. On the left, there's a sidebar with options like 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Properties', and 'Metadata'. The main panel shows a table with columns: Name, Modified, Access tier, Archive status, Blob type, and Size. A search bar at the top says 'Search blobs by prefix (case-sensitive)' and a toggle switch is next to it. The table below shows 'No results'.

Step4: Create a directory(ingest) in covid container and add files into the container.

This screenshot continues from the previous one, showing the 'covid' container page. The sidebar now includes a 'Properties' section with a 'Create directory' button. The main table now lists several CSV files: 'case_deaths_uk_ind_only.csv', 'cases_deaths.csv', 'country_response.csv', 'hospital_admissions.csv', and 'testing.csv'. Each file entry includes a checkbox, a preview icon, the name, the modified date (e.g., 3/8/2024, 12:37:04 PM), the access tier ('Hot (Inferred)'), the blob type ('Block blob'), and the size (e.g., 131, 13.7, 46.2, 1.01, 83.8).

Step5: Create an Azure Data Factory (covidadf2319933).

The screenshot shows the Microsoft Azure portal with the URL https://portal.azure.com/#Microsoft_Azure_Education_correlationId=88849426-1b11-40c5-b865-b1edc979822e&Microsoft_Azure_Education_newA4.... The page title is "covidadf2319933 - Microsoft Azure". The main content area is titled "Azure Data Factory Studio" and features a "Launch studio" button. On the left, there is a navigation sidebar with sections like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Networking, Managed identities, Properties, Locks), Getting started, and Quick start. The "Overview" section is selected. The center pane displays the "Subscription ID" as 49012363-76da-490e-836d-b25b3f8b274a and includes a blue factory icon.

Step6: Create a Synapse workspace(covid-synapse-workspace2319933).

The screenshot shows the Microsoft Azure portal with the URL <https://portal.azure.com/#@mits.ac.in/resource/subscriptions/49012363-76da-490e-836d-b25b3f8b274a/resourceGroups/covid-rg2319933/providers/Microsoft.DataFactory/factories/covidadf2319933#loginHint=...>. The page title is "covid-synapse-workspace2319933 - Microsoft Azure". The main content area shows the "Synapse workspace" interface. The left sidebar includes sections for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Microsoft Entra ID, Properties, Locks), Analytics pools (SQL pools, Apache Spark pools), and Tags. The "Overview" section is selected. The right pane displays workspace details such as Resource group ([move](#) covid-rg2319933), Status (Succeeded), Location (East US), Subscription ([move](#) Azure for Students), Subscription ID (49012363-76da-490e-836d-b25b3f8b274a), Managed virtual network (No), Managed Identity object ID (d7d93cf4-734d-4997-93fa-4290c8a30f7e), Workspace web URL (<https://web.azuresynapse.net?workspace=%2bsubscriptions%2f49012363-%2fcovid-rg2319933>), and various endpoint URLs (Dedicated SQL endpoint, Serverless SQL endpoint, Development endpoint). A "JSON View" link is also present.

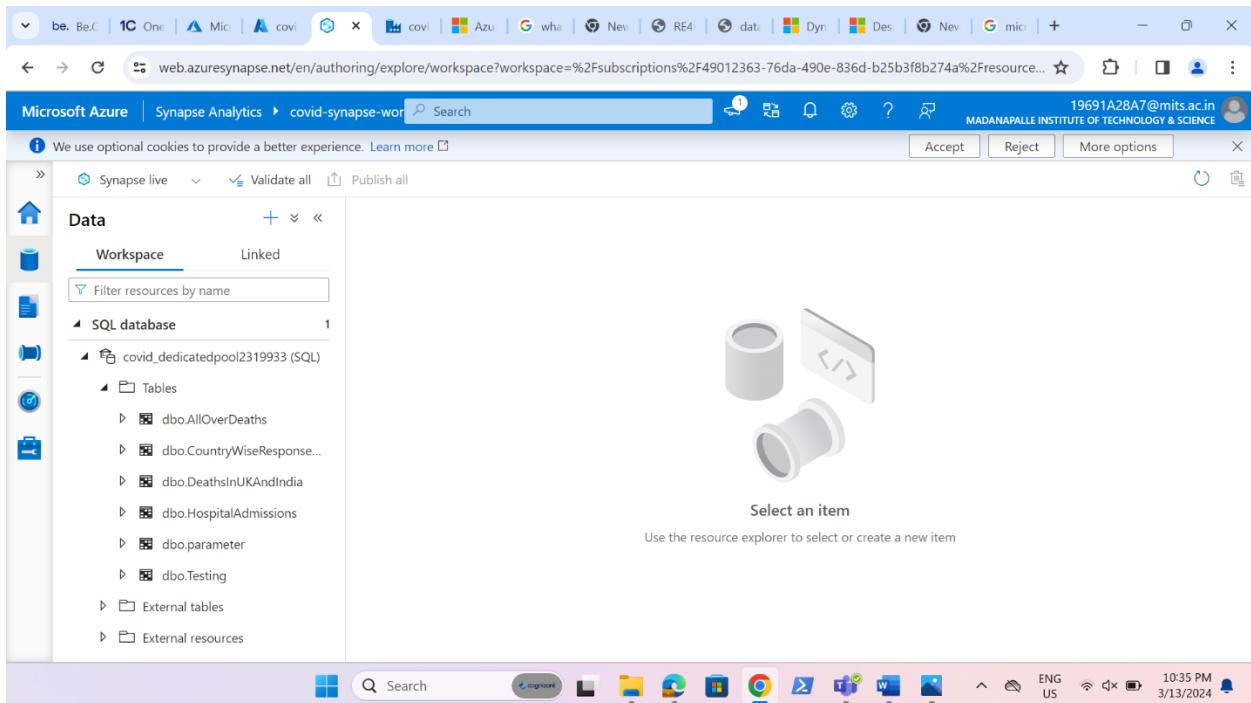
Step7: Create a dedicated pool(covid_dedicatedpool2319933).

The screenshot shows the Microsoft Azure portal interface. The user is navigating through the Azure Synapse workspace. On the left sidebar, under 'Analytics pools', 'SQL pools' is selected. In the main content area, a table lists existing pools: 'Built-in' (Serverless, N/A, Auto) and 'covid_dedicatedpool2319933' (Dedicated, Paused, DW100c). A search bar at the top is set to 'Search resources, services, and docs (G+)'. The top navigation bar includes links for Be.C, One, Mic, covi, Azur, wha, New, RE4v, data, Dyn, Desc, Nev, mic, and a plus sign for creating new resources.

Step8: Open the dedicated pool. Open the data in the left panel and select new SQL script to create table within the dedicated pool.

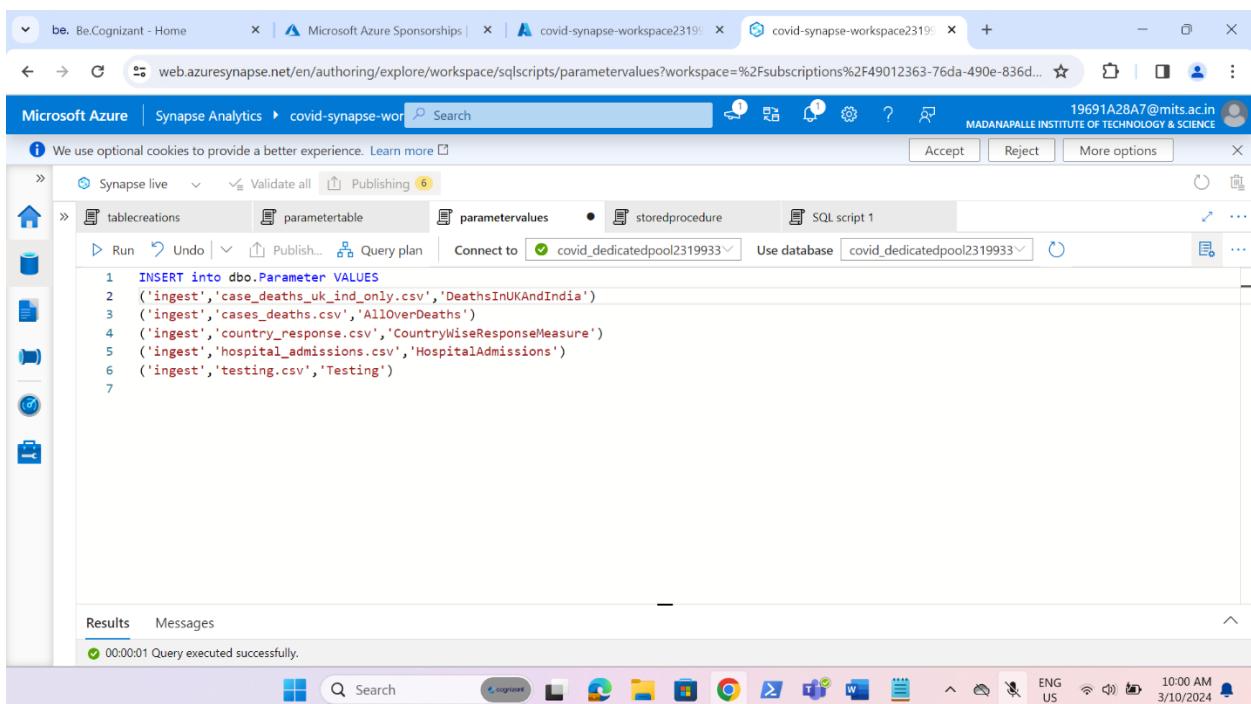
The screenshot shows the Azure Synapse Analytics workspace. The left sidebar under 'Data' shows a 'Workspace' section with one item, 'covid_dedicatedpool2319933 (SQL)'. A context menu is open over this item, with 'New SQL script' selected, showing a submenu with 'Empty script' and 'Bulk load'. The main workspace area displays two cylinders and a code editor icon. A message at the bottom says 'Select an item' and 'Use the resource explorer to select or create a new item'. The top navigation bar is identical to the previous screenshot, showing various Azure services and a user profile.

Step9: Create five tables in the sql scripts and run each of them.



The screenshot shows the Microsoft Azure Synapse Analytics Data Explorer. On the left, there's a navigation pane with icons for Home, Workspace, and Linked. Under 'Workspace', there's a section for 'SQL database' which is expanded to show 'covid_dedicatedpool2319933 (SQL)'. This database contains several tables: 'dbo.AllOverDeaths', 'dbo.CountryWiseResponse...', 'dbo.DeathsInUKAndIndia', 'dbo.HospitalAdmissions', 'dbo.parameter', 'dbo.Testing', 'External tables', and 'External resources'. To the right of the navigation pane, there's a large icon of two cylinders and a code editor window with the placeholder text 'Select an item'. Below the icon, it says 'Use the resource explorer to select or create a new item'. At the bottom of the screen, there's a taskbar with various application icons and system status indicators.

Step10: Create a table(parameter) to store all these 5 table schemas which is used for dynamic loading of data.

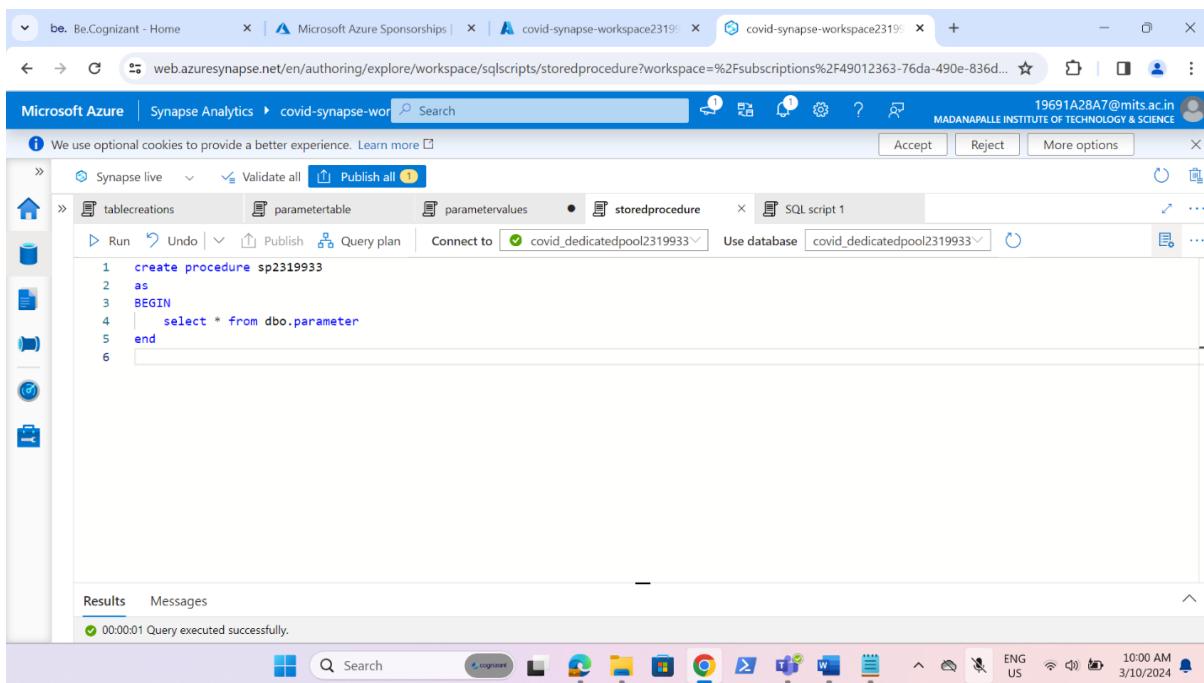


The screenshot shows the Microsoft Azure Synapse Analytics SQL Editor. The top navigation bar includes tabs for 'tablecreations', 'parametertable', 'parametervalues', 'storedprocedure', and 'SQL script 1'. The 'SQL script 1' tab is active. The main area contains a SQL script with the following content:

```
1 INSERT into dbo.Parameter VALUES
2 ('ingest','case_deaths_uk_ind_only.csv','DeathsInUKAndIndia')
3 ('ingest','cases_deaths.csv','AllOverDeaths')
4 ('ingest','country_response.csv','CountryWiseResponseMeasure')
5 ('ingest','hospital_admissions.csv','HospitalAdmissions')
6 ('ingest','testing.csv','Testing')
```

Below the script, there's a 'Results' tab with a message indicating '0:00:01 Query executed successfully.' The bottom of the screen features a taskbar with various application icons and system status indicators.

Step11: Create a stored procedure in Programmability which is present in left panel.

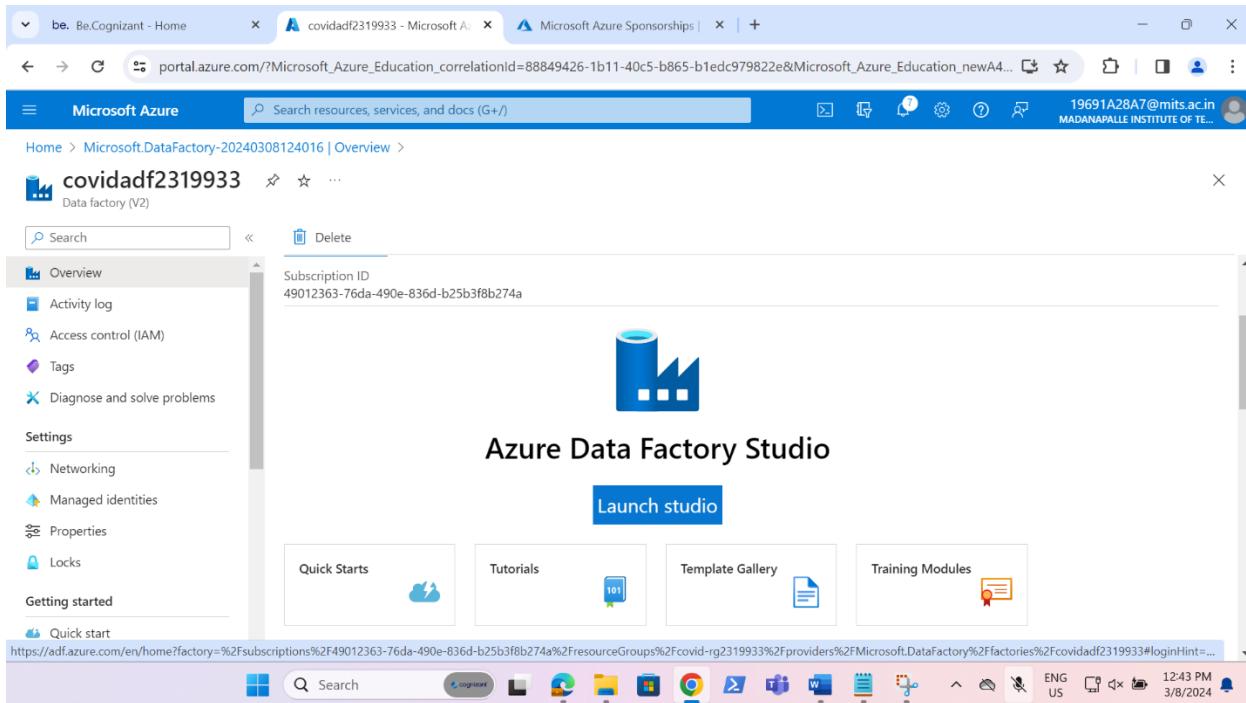


The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. In the center, there is a code editor window titled "storedprocedure". The code is as follows:

```
1 create procedure sp2319933
2 as
3 BEGIN
4     select * from dbo.parameter
5 end
6
```

Below the code editor, the "Results" tab is selected, showing the message: "00:00:01 Query executed successfully." The top navigation bar shows the URL: <https://web.azuresynthesize.net/en/authoring/explore/workspace/sqlscripts/storedprocedure?workspace=%2Fsubscriptions%2F49012363-76da-490e-836d-b25b3f8b274a%2FresourceGroups%2Fcovid-rg2319933%2Fproviders%2FMicrosoft.DataFactory%2Ffactories%2Fcovidadfd2319933#loginHint=...>

Step12: Launch the Azure Data Factory Studio (covidadfd2319933).



Step13: Create one Linked service (ls_dw2319933) for synapse workspace in manage.

The screenshot shows the Microsoft Azure Data Factory interface. The left sidebar has 'General' selected under 'Connections'. The main area is titled 'Linked services' and lists two items: 'ls_adls2319933' (Azure Data Lake Storage Gen2) and 'ls_dw2319933' (Azure Synapse Analytics). The table has columns for Name, Type, Related, and Annotations.

Name	Type	Related	Annotations
ls_adls2319933	Azure Data Lake Storage Gen2	3	
ls_dw2319933	Azure Synapse Analytics	1	

Step14: Create two datasets one for adls(ds_adls_2_2319933) and for synapse Workspace (ds_dwreq2_2319933).

The screenshot shows the Microsoft Azure Data Factory interface. The left sidebar has 'General' selected under 'Connections'. The main area is titled 'ds_adls_2_2319933' and shows a 'DelimitedText' dataset type. The configuration pane includes fields for 'Linked service' (ls_adls2319933), 'File path' (@covid / @dataset().FolderName / @dataset().FileName), 'Compression type' (Select...), 'Column delimiter' (Comma ()), 'Row delimiter' (Default (\r\n, or \n)), and 'Encoding' (Default(UTF-8)).

The screenshot shows the Microsoft Azure Data Factory interface. The left sidebar has 'Data Factory' selected. In the main area, a dataset named 'ds_dwreq2_2319933' is being edited. The 'Connection' tab is active, showing a linked service 'ls_dw2319933' and a table 'dbo'. The 'Schema' and 'Parameters' tabs are also visible. The top right shows the user's email '19691A28A7@mits.ac.in' and the institution 'MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE'. The bottom status bar shows the date '3/13/2024' and time '11:28 PM'.

Step15: Create a pipeline(copycoviddata_adls_synapse) that will able to copy the datasets in ADLSGen2 dynamically in a loop.

The screenshot shows the Microsoft Azure Data Factory interface displaying a pipeline named 'copycoviddata_adls_synapse'. The pipeline consists of a 'Lookup' activity followed by a 'ForEach' activity. Inside the 'ForEach' activity, there is another 'ForEach' activity and an 'Activities' section containing a 'Copy data' activity. The top right shows the user's email '19691A28A7@mits.ac.in' and the institution 'MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE'. The bottom status bar shows the date '3/13/2024' and time '11:52 PM'.

Step16: Now debug the pipeline and make sure the dedicated pool which was created before should not pause it should be on online mode.

The screenshot shows the Microsoft Azure Data Factory pipeline status page. The pipeline run ID is 7fdcb644-0dcb-4681-a94a-bdacc731204d, and the Pipeline status is Succeeded. The table below lists the activities and their statuses from the run:

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime	User properties	Activity run ID
Copy data	Succeeded	Copy data	3/10/2024, 10:31:58 AM	37s	AutoResolveIntegration	9ba639f5-f5c	
Copy data	Succeeded	Copy data	3/10/2024, 10:31:58 AM	22s	AutoResolveIntegration	f0dbc4b7-c3:	
Copy data	Succeeded	Copy data	3/10/2024, 10:31:58 AM	21s	AutoResolveIntegration	240d7055-a3	
Copy data	Succeeded	Copy data	3/10/2024, 10:31:58 AM	19s	AutoResolveIntegration	b6e6c016-70	
Copy data	Succeeded	Copy data	3/10/2024, 10:31:58 AM	22s	AutoResolveIntegration	27aa9c15-0d	
ForEach	Succeeded	ForEach	3/10/2024, 10:31:57 AM	54s		554126b7-5f	
Lookup	Succeeded	Lookup	3/10/2024, 10:31:10 AM	45s	AutoResolveIntegration	0b22635d-61	

Step17: On successful completion of debugging check whether all the files are loaded in the sink or not.

To check this write some queries in synapse workspace.

The screenshot shows the Microsoft Azure Synapse Analytics workspace. A query has been run against the covid_dedicatedpool2319933 database:

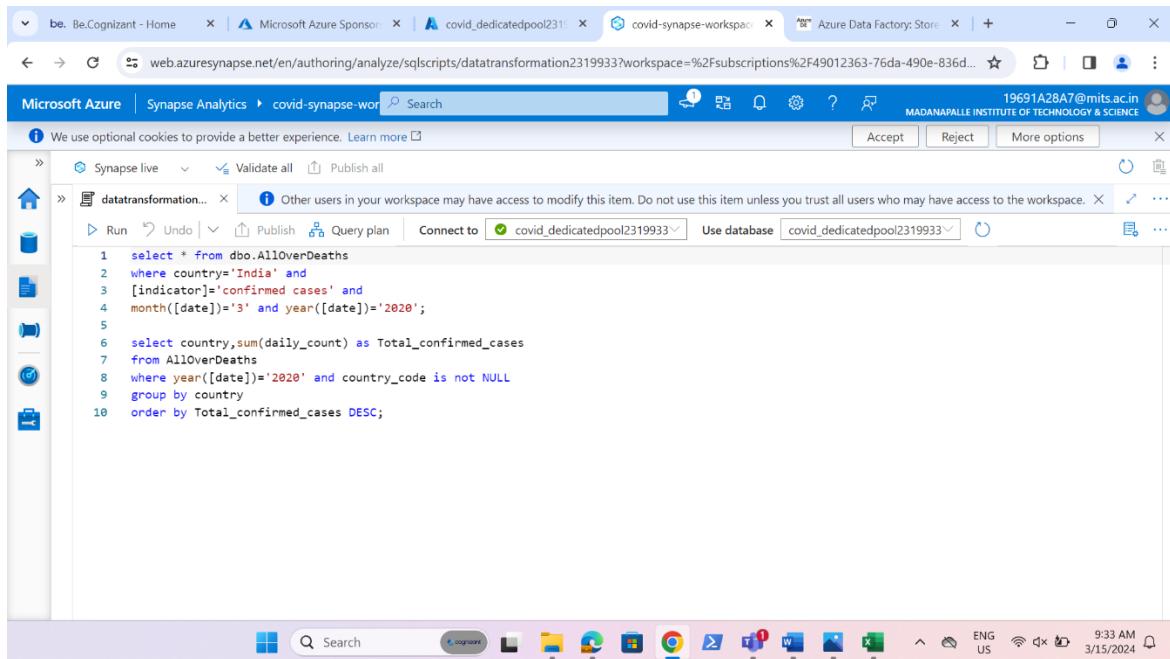
```
1 select * from dbo.AllOverDeaths;
```

The results table shows data for countries like Austria and Azerbaijan:

country	country_code	continent	population	indicator	daily_count	date	rate_14_day	source
Austria	AUT	Europe	8858775	confirmed cases	0	2020-02-10T00:00:00	0	Epidemic intelli...
Austria	AUT	Europe	8858775	deaths	0	2020-01-08T00:00:00	(NULL)	Epidemic intelli...
Austria	AUT	Europe	8858775	deaths	3	2020-10-19T00:00:00	10	Epidemic intelli...
Azerbaijan	AZE	Europe	10139175	confirmed cases	142	2020-09-11T00:00:00	20	Epidemic intelli...

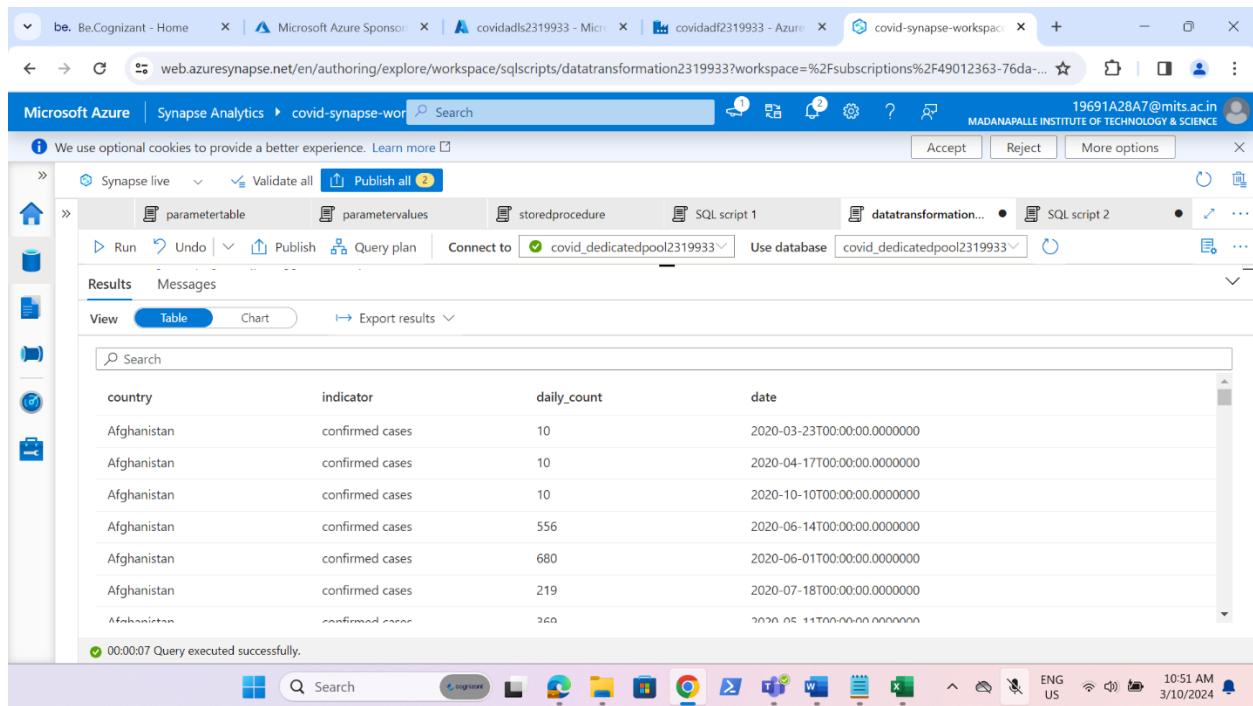
DATA VERIFICATION:

- Write a Query to check the confirmed cases in India in March'2020 and verify with source data in excel.
- Write a query to show country wise confirmed case in 2020.



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. A SQL script titled 'datatransformation...' is open in the center pane. The script retrieves data from the 'AllOverDeaths' table where the country is 'India', the indicator is 'confirmed cases', and the month of the date is '3' (March). It then groups the results by country and orders them by the total number of confirmed cases in descending order. The results are displayed in a table below.

```
1 select * from dbo.AllOverDeaths
2 where country='India' and
3 [indicator]='confirmed cases' and
4 month([date])='3' and year([date])='2020';
5
6 select country,sum(daily_count) as Total_confirmed_cases
7 from AllOverDeaths
8 where year([date])='2020' and country_code is not NULL
9 group by country
10 order by Total_confirmed_cases DESC;
```



The screenshot shows the Microsoft Azure Synapse Analytics workspace interface after the SQL script has been executed. The results tab is selected, displaying the output of the query. The table shows data for Afghanistan, specifically for the 'confirmed cases' indicator, across different dates in 2020. The data includes the country, indicator, daily count, and date.

country	indicator	daily_count	date
Afghanistan	confirmed cases	10	2020-03-23T00:00:00.0000000
Afghanistan	confirmed cases	10	2020-04-17T00:00:00.0000000
Afghanistan	confirmed cases	10	2020-10-10T00:00:00.0000000
Afghanistan	confirmed cases	556	2020-06-14T00:00:00.0000000
Afghanistan	confirmed cases	680	2020-06-01T00:00:00.0000000
Afghanistan	confirmed cases	219	2020-07-18T00:00:00.0000000
Afghanistan	confirmed cases	260	2020-05-11T00:00:00.0000000

00:00:07 Query executed successfully.

Microsoft Azure | Synapse Analytics > covid-synapse-workspace | Search

We use optional cookies to provide a better experience. Learn more

Synapse live | Validate all | Publish all

SQL script 2 | Other users in your workspace may have access to modify this item. Do not use this item unless you trust all users who may have access to the workspace.

Run | Undo | Publish | Query plan | Connect to: covid_dedicatedpool2319933 | Use database: covid_dedicatedpool2319933

Results | Messages | View: Table | Export results

Search

country	TotalConfirmedCases
India	17603248
Brazil	15966690
Russia	11075076
France	3045976
Argentina	2243816
Spain	2219872
United States	2167770

00:00:03 Query executed successfully.

The screenshot shows a Microsoft Azure Synapse Analytics workspace interface. The main area displays a table titled 'TotalConfirmedCases' with data for several countries. The table has two columns: 'country' and 'TotalConfirmedCases'. The data includes India (17,603,248), Brazil (15,966,690), Russia (11,075,076), France (3,045,976), Argentina (2,243,816), Spain (2,219,872), and the United States (2,167,770). The interface also shows navigation tabs for 'Synapse live', 'Validate all', and 'Publish all'. A message at the top right indicates that other users in the workspace may have access to modify the item. The bottom status bar shows the query execution time as 00:00:03 and a success message.

Procedure of Requirement 2:

Step1: Create a Resource group(covid-rg2319933).

Validation passed.

Basics

Subscription: Azure for Students
Resource group: covid-rg2319933
Region: East US

Tags

Owner: yukthareddy

Create < Previous Next > Download a template for automation

Step2: Create an Azure Datalake storage (covidadls2319933).

covidadls2319933

Storage account

Search

Upload Open in Explorer Delete Move Refresh Open in mobile CLI / PS Feedback

Overview

Essentials

Resource group (move)
covid-rg2319933

Location
eastus

Subscription (move)
Azure for Students

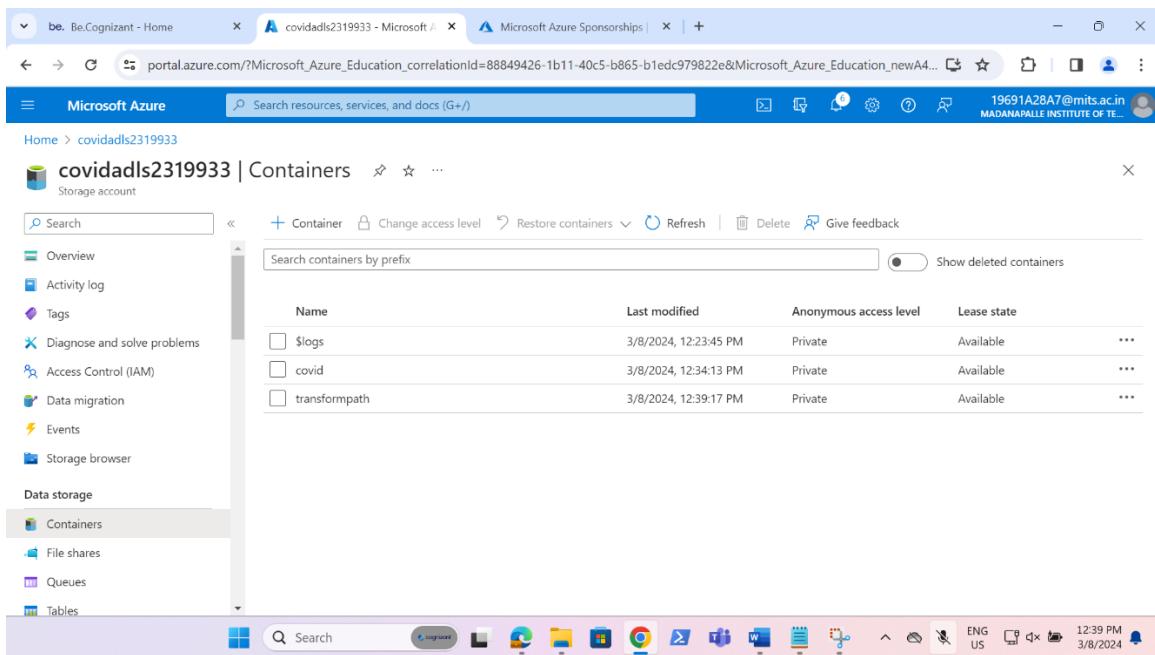
Subscription ID
49012363-76da-490e-836d-b25b3f8b274a

Disk state
Available

Tags (edit)
Add tags

Properties Monitoring Capabilities (5) Recommendations (0) Tutorials Tools + SDKs

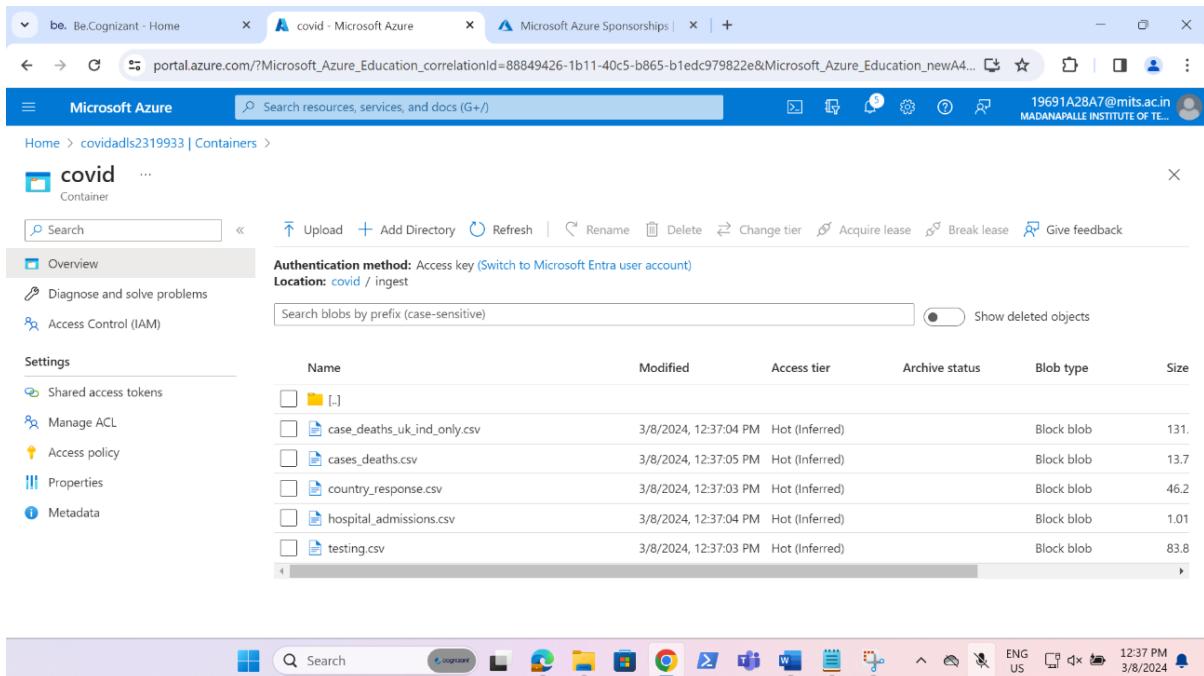
Step3: Create two containers(covid,transformpath) within adls storage.



The screenshot shows the Microsoft Azure Storage account interface for 'covidadls2319933'. The left sidebar has 'Containers' selected under 'Data storage'. The main area displays a table of containers:

Name	Last modified	Anonymous access level	Lease state
\$logs	3/8/2024, 12:23:45 PM	Private	Available
covid	3/8/2024, 12:34:13 PM	Private	Available
transformpath	3/8/2024, 12:39:17 PM	Private	Available

Step4: Create a directory(ingest) in covid container and add files into the container.



The screenshot shows the Microsoft Azure Storage account interface for the 'covid' container. The left sidebar has 'Overview' selected under 'Container'. The main area displays a table of blobs:

Name	Modified	Access tier	Archive status	Blob type	Size
[...]					
case_deaths_uk_ind_only.csv	3/8/2024, 12:37:04 PM	Hot (Inferred)		Block blob	131.
cases_deaths.csv	3/8/2024, 12:37:05 PM	Hot (Inferred)		Block blob	13.7
country_response.csv	3/8/2024, 12:37:03 PM	Hot (Inferred)		Block blob	46.2
hospital_admissions.csv	3/8/2024, 12:37:04 PM	Hot (Inferred)		Block blob	1.01
testing.csv	3/8/2024, 12:37:03 PM	Hot (Inferred)		Block blob	83.8

Step5: Create an Azure Data Factory (covidadf2319933).

The screenshot shows the Microsoft Azure portal interface for an Azure Data Factory named 'covidadf2319933'. The left sidebar contains navigation links for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (Networking, Managed identities, Properties, Locks), and Getting started (Quick start). The main area displays the 'Subscription ID' (49012363-76da-490e-836d-b25b3f8b274a) and features a large blue factory icon. Below the icon, the text 'Azure Data Factory Studio' is centered, with a 'Launch studio' button. To the right of the button are four cards: 'Quick Starts' (cloud icon), 'Tutorials' (book icon with '101'), 'Template Gallery' (document icon), and 'Training Modules' (certificate icon). The bottom status bar shows the URL 'https://adf.azure.com/en/home?factory=%2Fsubscriptions%2F49012363-76da-490e-836d-b25b3f8b274a%2FresourceGroups%2Fcovid-rg2319933%2Fproviders%2FMicrosoft.DataFactory%2Ffactories%2Fcovidadf2319933#loginHint=' and the system time '12:43 PM 3/8/2024'.

Step6: Open Azure Data Factory Studio create one Linked service (ls_adls2319933) for Adls in manage.

The screenshot shows the Microsoft Azure portal interface for the 'Data Factory' section of the 'covidadf2319933' factory. The left sidebar lists options: General, Factory settings, Connections (selected), Integration runtimes, Microsoft Purview, Source control, Git configuration, ARM template, Author, Triggers, Global parameters, Data flow libraries, and Security. The main area is titled 'Linked services' and contains a table of existing linked services:

Name	Type	Related	Annotations
ls_adls2319933	Azure Data Lake Storage Gen2	3	
ls_dw2319933	Azure Synapse Analytics	1	

The bottom status bar shows the URL 'https://adf.azure.com/en/management/datalinkedservices?factory=%2Fsubscriptions%2F49012363-76da-490e-836d-b25b3f8b274a%2FresourceGroups%2Fcovid-rg2319933%2FlinkedServices' and the system time '12:43 PM 3/8/2024'.

Step7: Create two datasets, one for source path and one for sink path(ds_adls_ingest2319933) and (ds_adls_transform2319933) in author.

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (1), and 'Data flows' (0). The main workspace displays a pipeline named 'pipeline1'. A dataset named 'ds_adls_ingest2319933' is selected, shown as a 'DelimitedText CSV' file icon. Below the dataset, the 'Connection' tab is active, showing a linked service 'ls_adls2319933' with a successful connection. The 'File path' is set to 'covid / ingest / cases_deaths.csv'. Other settings include 'Compression type' (Select...), 'Column delimiter' (Comma (,), selected), and 'Row delimiter' (Default (\r\n, or \n)).

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (1), 'Datasets' (1), and 'Data flows' (0). The main workspace displays a dataset named 'ds_adls.transform2319933' (note the misspelling). The 'Connection' tab is active, showing a linked service 'ls_adls2319933' with a successful connection. The 'File path' is set to 'transformpath / Directory / File name'. Other settings include 'Compression type' (Select...), 'Column delimiter' (Comma (,), selected), 'Row delimiter' (Default (\r\n, or \n)), and 'Encoding' (Default(UTF-8)).

Step8: Create a Dataflow(covidtransform2319933) and after creating turn on the dataflow debug.

The screenshot shows the Microsoft Azure Data Factory Data Flow blade. A complex pipeline named 'covidtransform2319933' is displayed. The pipeline consists of several stages: 'source' (Import data from ds_adls_ingest2319933), 'cast1' (Cast columns to different types), 'pivot' (Pivots raw values into columns, groups columns and aggregates), 'aggregate' (Aggregating data by 'continent' producing columns 'countdeaths'), 'rank' (Ranking rows on columns 'countdeaths'), and 'sink' (Export data to ds_adls_transform2319933). A tooltip for 'Data flow debug' indicates the cluster is ready with Session ID: 407cb907-eb48-4349-b48d-dde1651b1f86. The pipeline is currently in 'Data flow debug' mode.

Step9: Create a Pipeline(covidtotransformpath2319933), drag and drop the dataflow activity and select the above dataflow and debug it.

The screenshot shows the Microsoft Azure Data Factory Pipeline blade. A pipeline named 'covidtotransformpath2319933' is displayed. The pipeline run status is shown as 'Succeeded'. The pipeline run ID is 60280909-5a9c-490c-a197-6e4183265eac. The pipeline status is 'Succeeded'. The pipeline run duration was 51s. The pipeline run user properties include 'debugpool-8Cores-Gen' and '475a3e15-'. The pipeline run activity status is 'Succeeded'.

Step10: After successful completion of debugging check whether the file uploaded in transformpath or not.

The screenshot shows the Microsoft Azure Storage Explorer interface. The left sidebar displays the 'transformpath' container with its settings and access controls. The main pane shows a single blob named 'finaldata2319933.csv' with details: Modified on 3/8/2024, 5:10:20 PM, Access tier Hot (Inferred), Archive status Not yet archived, Blob type Block blob, and Size 98 B. The browser address bar shows the URL for the Azure portal, and the system tray at the bottom right indicates the date and time as 5:12 PM on 3/8/2024.

DATA VERIFICATION:

The screenshot shows the details of the 'finaldata2319933.csv' blob. The blob's properties are displayed, including Save, Discard, Download, Refresh, and Delete options. The 'Edit' tab is selected, showing a table with the following data:

continent	countdeaths	Ranking
America	7740	1
Europe	5363	2
Asia	2500	3
Africa	698	4
Oceania	60	5

Queries used in Project:

5 SQL QUERIES:

```
create table Testing(  
country Varchar(1000),  
country_code Varchar(1000),  
year_week Varchar(1000),  
new_cases BigInt,  
tests_done BigInt,  
population BigInt,  
testing_rate Decimal,  
positivity_rate Decimal,  
testing_data_source Varchar(2000)  
);
```

```
create table HospitalAdmissions  
(country Varchar(1000),  
indicator Varchar(1000),  
date Date,  
year_week Varchar(1000),  
value Decimal,  
source Varchar(1000),  
url Varchar(2000)  
);
```

```
create table CountryWiseResponseMeasure(  
Country varchar(100),  
Response_measure varchar(100),  
change int,  
date_start date,  
date_end varchar(100)  
);
```

```
create table AllOverDeaths(country Varchar(100),  
country_code Varchar(100),  
continent Varchar(100),  
population BigInt,  
indicator Varchar(100),  
daily_count BigInt,  
date date,
```

```
rate_14_day Decimal,  
source Varchar(100)  
);  
  
create table DeathsInUKAndIndia (country Varchar(100),  
country_code Varchar(100),  
continent Varchar(100),  
population BigInt,  
indicator Varchar(100),  
daily_count BigInt,  
date date,  
rate_14_day Decimal,  
source Varchar(100)  
);
```

Parameter Table schema:

```
create table parameter  
(  
    FolderName VARCHAR(200),  
    [FileName] VARCHAR(200),  
    SQLTable VARCHAR(200)  
);  
  
INSERT into dbo.Parameter VALUES  
('ingest','case_deaths_uk_ind_only.csv','DeathsInUKAndIndia')  
('ingest','cases_deaths.csv','AllOverDeaths')  
('ingest','country_response.csv','CountryWiseResponseMeasure')  
('ingest','hospital_admissions.csv','HospitalAdmissions')  
('ingest','testing.csv','Testing')
```

Stored Procedure Schema:

```
create procedure sp3final  
as  
BEGIN  
    select * from dbo.parameter  
end
```

Data Verification queries:

```
select * from dbo.AllOverDeaths  
where country='India' and  
[indicator]='confirmed cases' and  
month([date])='3' and year([date])='2020';
```

```
select country,sum(daily_count) as Total_confirmed_cases  
from AllOverDeaths  
where year([date])='2020' and country_code is not NULL  
group by country  
order by Total_confirmed_cases DESC;
```