

AZURE DATA FACTORY CAPSTONE - COVID USE CASE

Name : Induja D (2319849)

Cohort Code : CSDAIA24AZ003

Introduction:

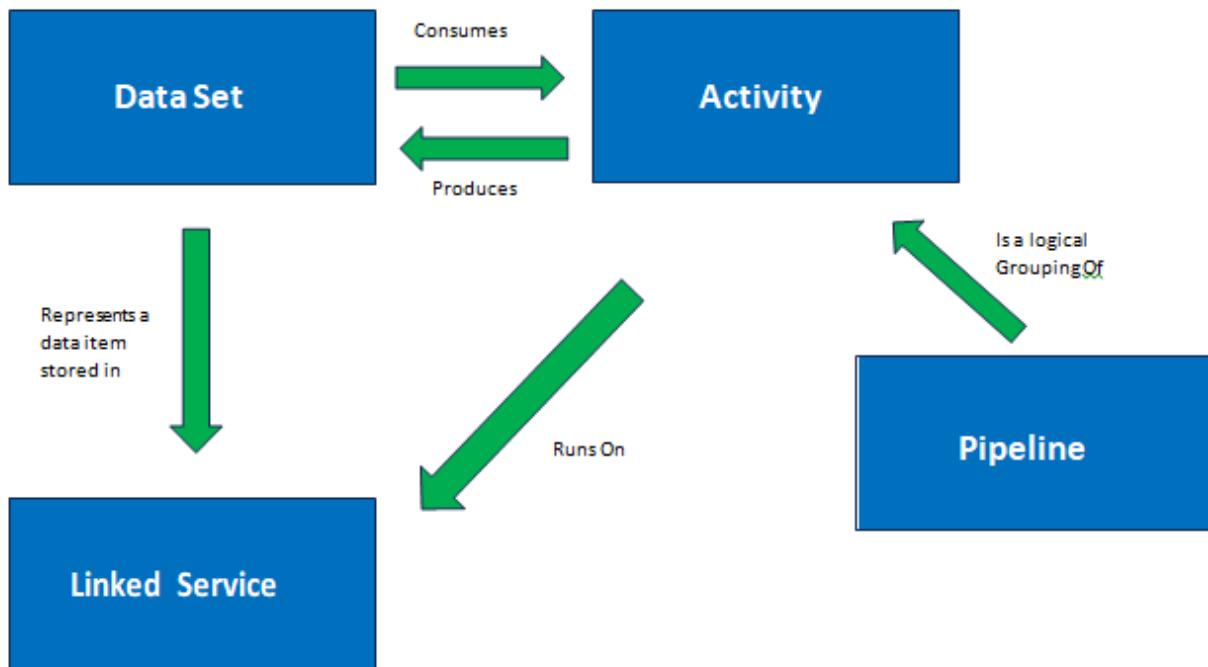
The "Azure Data Factory Capstone - Covid Use Case" project was completed in compliance with business requirements using a variety of tools, including Synapse, Azure Data Factory, and Azure Data Lake Storage. These Azure components facilitate the process of removing data from its source (Data Lake), transforming it to meet business requirements, and then loading it into a destination data warehouse. The "Azure Data Factory Capstone - Covid Use Case" system was developed to teach users how to use Azure cloud data services to build a practical data pipeline in Azure Data Factory (ADF) for the purpose of analyzing the covid trend across the regions. This case study will teach us anything.

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

Objective:

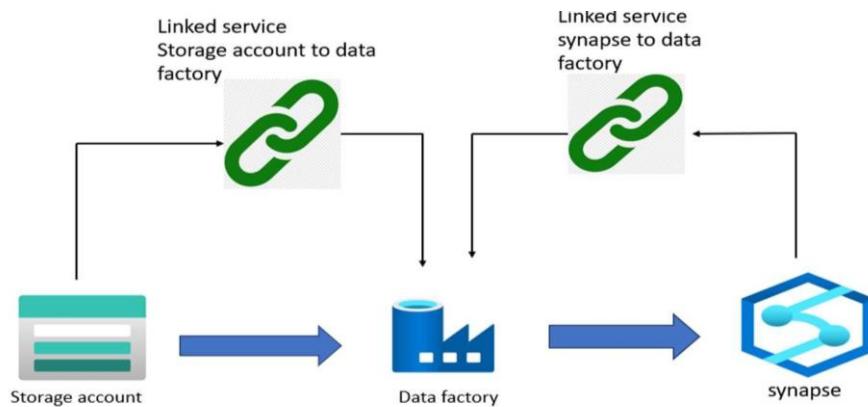
This project's main goal is to give us hands-on experience with storage, ADF pipelines, mapping data flows, and Azure Synapse. It also aims to teach us how to use Azure Data Factory (ADF) to ingest data from flat files into Azure Data Lake Gen2 and Azure Synapse, as well as how to transform data using Data Flows in Azure Data Factory (ADF) and load it into Azure data lakes. This report provides an overview of the entire project, enabling us to comprehend and analyze the Azure use case scenario and its applications.

Overview of Data Factory Flow:

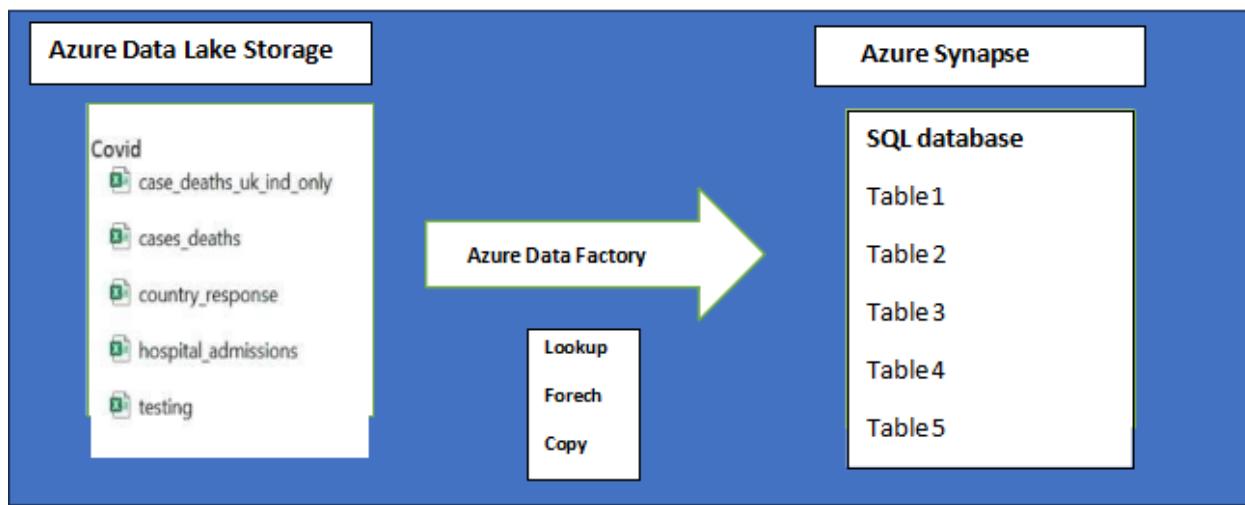


Project Requirements:

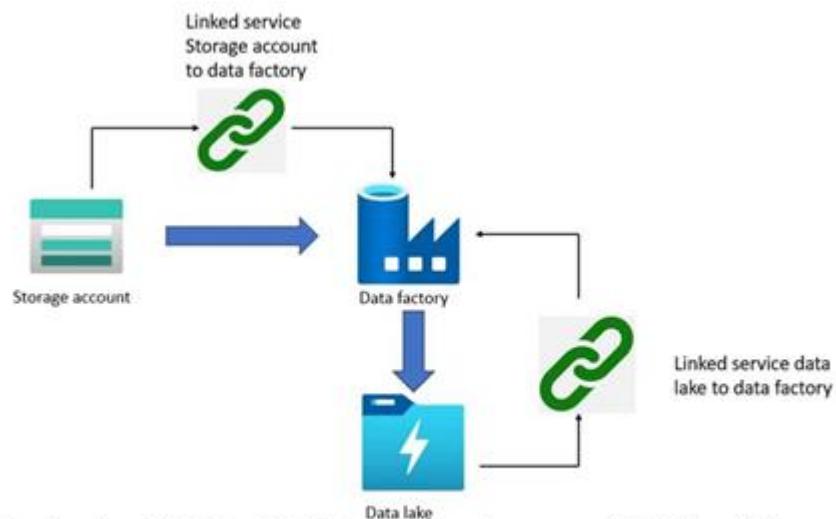
Requirement 1: Ingest raw flat files from Data Lake to synapse (data warehouse)



Ingesting process taken place in data factory by taking raw data from storage account(data lake) to relation database by creating tables in synapse(data warehouse)

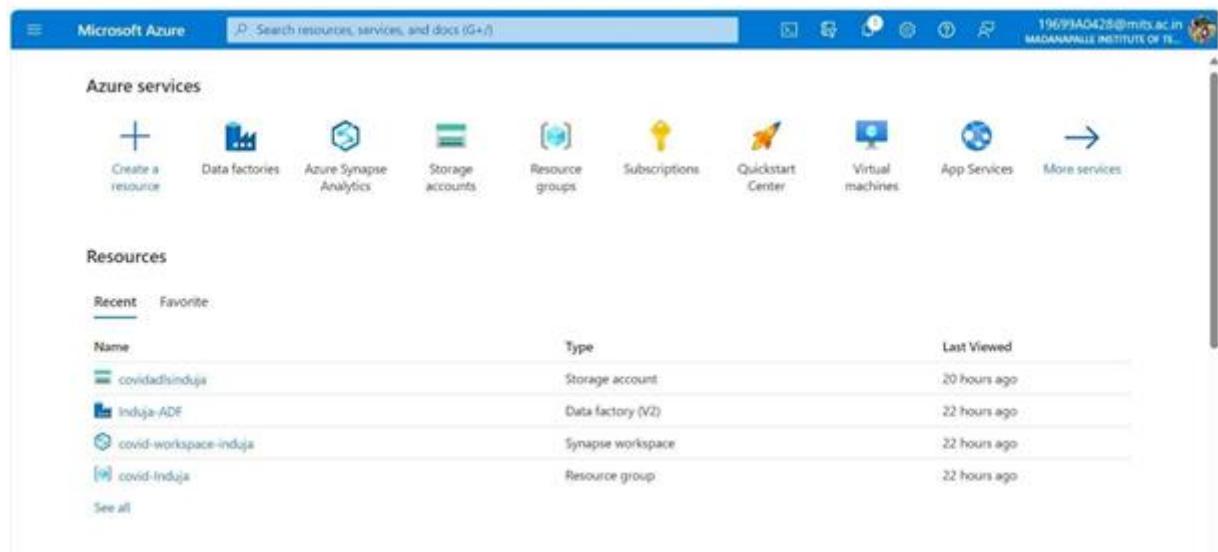


Requirement 2: Transform data using Data Flows in AzureData Factory (ADF)



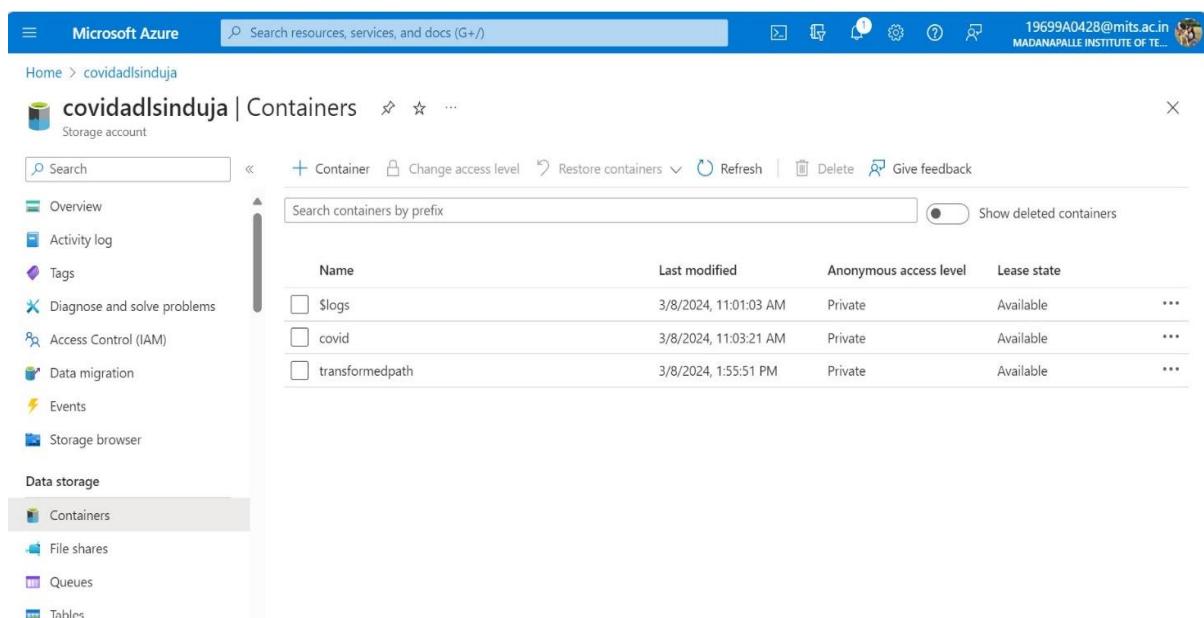
Data Transformation in data factory by taking Raw data from Storage account(data lake) modifying as per over requirements given in project and storing that file in storage account(data lake)





The screenshot shows the Microsoft Azure home page. At the top, there's a search bar and several navigation icons. Below the header, there's a section titled "Azure services" with links to "Create a resource", "Data factories", "Azure Synapse Analytics", "Storage accounts", "Resource groups", "Subscriptions", "Quickstart Center", "Virtual machines", "App Services", and "More services". Under "Resources", there's a table showing recent resources: "covidadlsinduja" (Storage account), "Induja-ADF" (Data factory (V2)), "covid-workspace-induja" (Synapse workspace), and "covid-Induja" (Resource group). A "See all" link is also present.

Step 1: Created one Resource group(**covid-Induja**) and required resources for project like Storage account(**covidadlsinduja**), Synapse workspace (data warehouse) (**covid-workspace-induja**), Azure Datafactory(**Induja-ADF**).



This screenshot shows the Azure Storage account interface for the container "covidadlsinduja". The left sidebar includes links for Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, and Storage browser. Under "Data storage", "Containers" is selected. The main area displays a list of containers with columns for Name, Last modified, Anonymous access level, and Lease state. The containers listed are \$logs, covid, and transformedpath, all of which are private and available.

Name	Last modified	Anonymous access level	Lease state
\$logs	3/8/2024, 11:01:03 AM	Private	Available
covid	3/8/2024, 11:03:21 AM	Private	Available
transformedpath	3/8/2024, 1:55:51 PM	Private	Available

Step 2: Created Container with name “**covid**” in Storage account (Data Lake) for holding folder which contains flat files.

The screenshot shows the Microsoft Azure Storage Container Overview page for a container named 'covid'. The top navigation bar includes the Microsoft Azure logo, a search bar, and various icons. The main content area displays the container's details: 'Authentication method: Access key (Switch to Microsoft Entra user account)' and 'Location: covid'. On the left, there is a sidebar with options like 'Overview', 'Diagnose and solve problems', 'Access Control (IAM)', 'Settings' (with sub-options for Shared access tokens, Manage ACL, Access policy, Properties, and Metadata), and a 'Search' bar. The main pane shows a table of blobs with columns: Name, Modified, Access tier, Archive status, and Blob type. A single blob named 'ingest' is listed under the 'Name' column.

The screenshot shows the Microsoft Azure Storage Container Overview page for a container named 'covid / ingest'. The top navigation bar and sidebar are identical to the previous screenshot. The main content area displays the container's details: 'Authentication method: Access key (Switch to Microsoft Entra user account)' and 'Location: covid / ingest'. The sidebar options remain the same. The main pane shows a table of blobs with columns: Name, Modified, Access tier, Archive status, and Blob type. Multiple CSV files are listed under the 'Name' column, including 'case_deaths_uk_ind_only.csv', 'cases_deaths.csv', 'country_response.csv', 'hospital_admissions.csv', and 'testing.csv'. All blobs are categorized as 'Block blob'.

Step 3: Create Folder with name “**ingest**” inside the Container “**covid**” and uploaded csv files (data sets) form personal computer which are there in zip file given in project document.

The screenshot shows the Microsoft Azure Synapse Analytics interface. The left sidebar navigation includes options like Synapse live, Validate all, Publish all, Analytics pools, SQL pools (selected), Apache Spark pools, Data Explorer pools, External connections, Linked services, Microsoft Purview, Integration, Triggers, Integration runtimes, Security, Access control, Credentials, and Managed private endpoints. The main content area is titled "SQL pools" and displays the following table:

Name	Type	Status	Size
Built-in	Serverless	Online	Auto
indujadedicatedpool	Dedicated	Online	DW100c

Step 4: Create Azure synapse resource and one **dedicated pool** inside the azure synapse for data warehouse creation and it should be turn on.

Step 5: Creating **two linked services** as per the project requirement in Azure data factory.

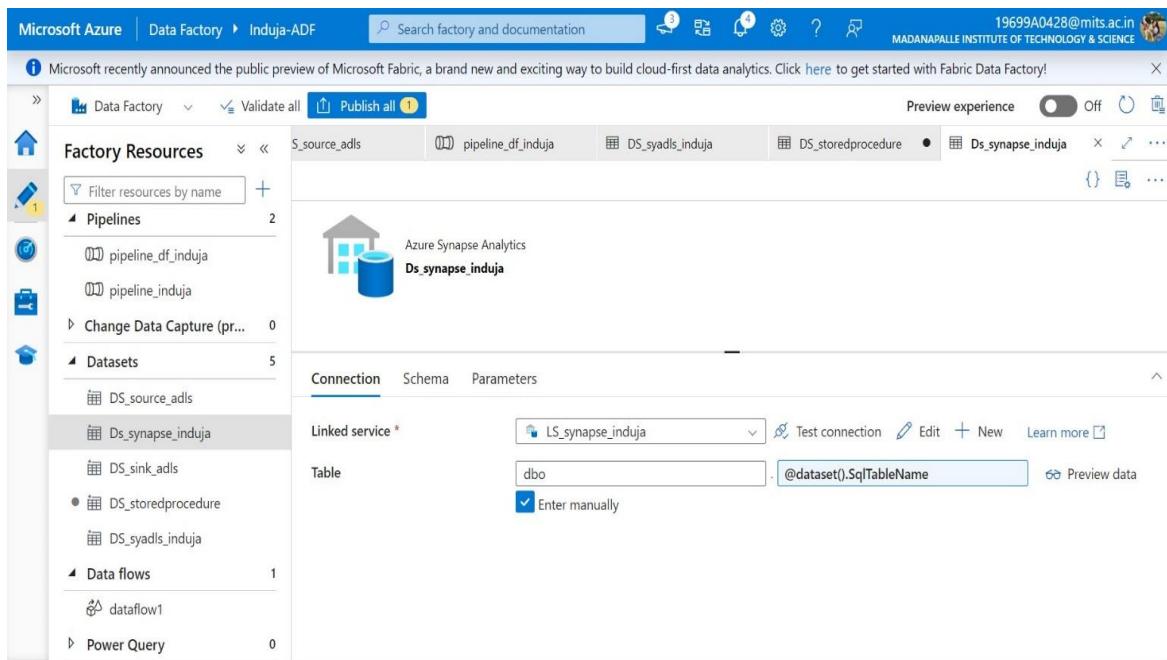
- **Storage account (Data Lake) to Azure data factory.**
- **Azure synapse workspace to Azure Data factory.**

The screenshot shows the Microsoft Azure Data Factory interface. The left sidebar navigation includes options like Data Factory, Validate all, Publish all, Factory Resources (selected), Pipelines, Datasets, Data flows, and Power Query. The main content area is titled "dataflow1" and shows the following configuration for a "DelimitedText" dataset:

Connection	Schema	Parameters
Linked service *	LS_syadls_induja	Test connection Edit + New Learn more
File path *	covid / @dataset().FolderName / @dataset().FileName	
Compression type	Select...	
Column delimiter	Comma (,)	
Row delimiter	Default (\r\n, or \n)	
Encoding	UTF-8	

Step 6: Created dataset (**DS_syadls_induja**) for fetching flat files from storage account (data lake) which are present in ingest folder inside covid container.

Step 7: Created required **SQL Tables** in synapse SQL database by writing create table queries in Synapse workspace (SQL Script).



The screenshot shows the Microsoft Azure Data Factory interface. On the left, there's a sidebar titled 'Factory Resources' with sections for Pipelines, Datasets, Data flows, and Power Query. Under 'Datasets', several datasets are listed: DS_source_adls, Ds_synapse_induja, DS_sink_adls, DS_storedprocedure, and DS_syadls_induja. The 'Ds_synapse_induja' dataset is currently selected. The main panel displays the 'Ds_synapse_induja' dataset details. It shows a preview icon for 'Azure Synapse Analytics' and the name 'Ds_synapse_induja'. Below this, there are tabs for 'Connection', 'Schema', and 'Parameters'. Under 'Connection', it shows a linked service named 'LS_synapse_induja' and a table named 'dbo'. A checkbox labeled 'Enter manually' is checked. At the top of the main panel, there are buttons for 'Publish all' and 'Validate all'.

Step 8: Created dataset (**Ds_synapse_induja**) for inserting into SQL Tables created in synapse (data warehouse).

```
97 Create PROCEDURE GetConfigSrcFileNames as  
98 begin  
99 select * from parameters  
100 end
```

Results Messages

00:00:06 Query executed successfully.

Step 9: Created stored procedure(**sp_GetConfigSrcFileNames**) for fetching records from parameters table.

Step 10: Created dataset(**DS_storedprocedure**) for fetching parameters table form synapse(data warehouse).

Screenshot of the Microsoft Azure Data Factory interface showing the 'DS_storedprocedure' dataset details.

The left sidebar shows 'Factory Resources' with the following items:

- Pipelines: pipeline_df_induja, pipeline_induja
- Change Data Capture (preview): 0
- Datasets: Ds_synapse_induja, DS_sink_adls, DS_source_adls, DS_storedprocedure (selected), DS_syadls_induja
- Data flows: 1
- Power Query: 0

The main pane displays the 'DS_storedprocedure' dataset, which is an Azure Synapse Analytics dataset. It includes tabs for Connection, Schema, and Parameters. Under Connection, it shows a linked service named 'LS_synapse_induja'. The Schema tab shows a single table named 'Select...'. There are buttons for Test connection, Edit, New, Refresh, Preview data, and Enter manually.

Screenshot of the Microsoft Azure Data Factory interface showing the 'DS_storedprocedure' dataset details.

The left sidebar shows 'Factory Resources' with the following items:

- Pipelines: pipeline_df_induja, pipeline_induja
- Change Data Capture (preview): 0
- Datasets: Ds_synapse_induja, DS_sink_adls, DS_source_adls, DS_storedprocedure (selected), DS_syadls_induja
- Data flows: 1
- Power Query: 0

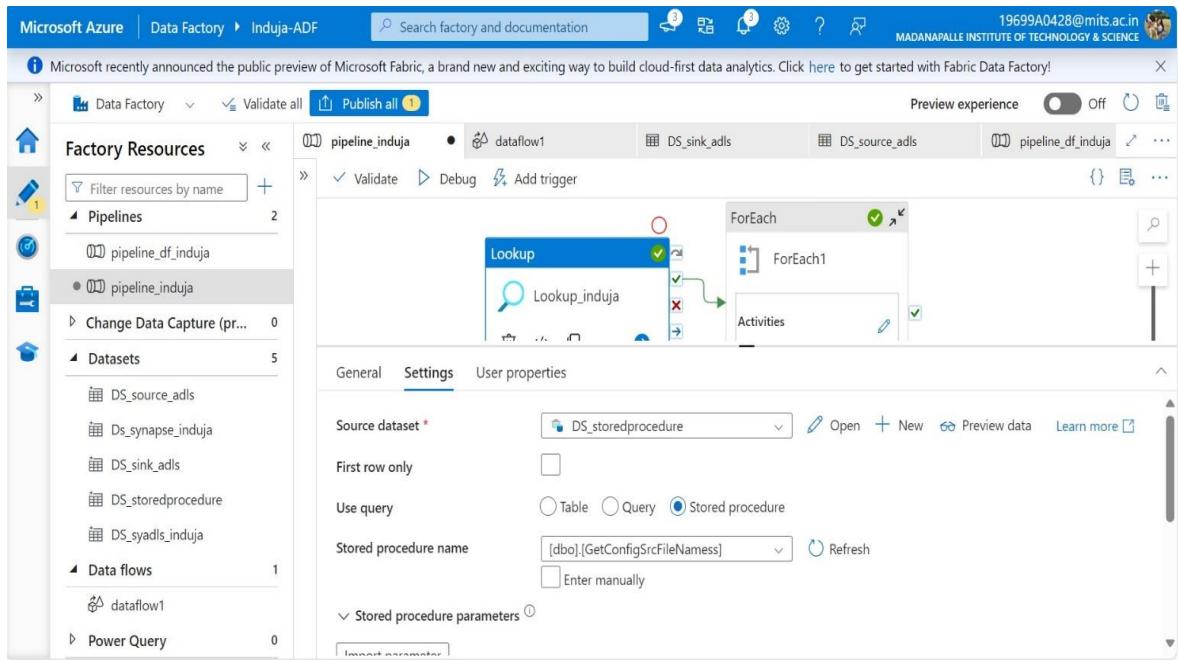
The main pane displays the 'DS_storedprocedure' dataset, which is an Azure Synapse Analytics dataset. It includes tabs for Connection, Schema, and Parameters. Under Connection, it shows a linked service named 'LS_synapse_induja'. The Schema tab shows a single table named 'Select...'. There are buttons for Test connection, Edit, New, Refresh, Preview data, and Enter manually.

Step 11: Creating required datasets in Azure data factory.

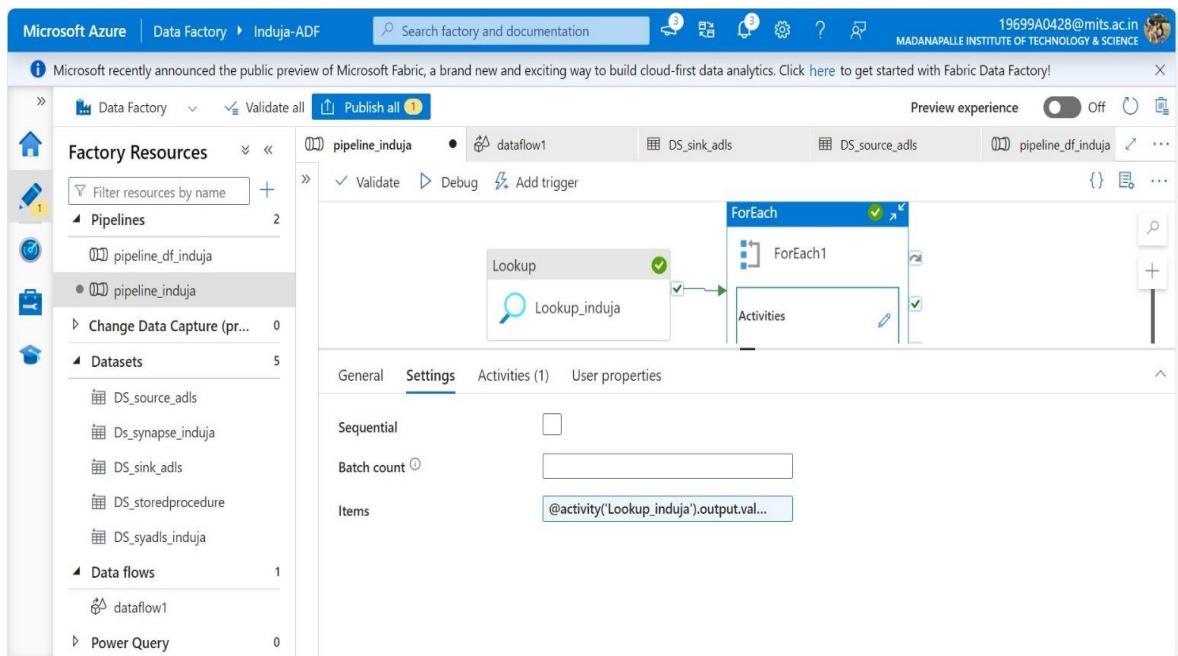
- To fetch covid flat files from Storage account (data lake) □(**DS_syadlsinduja**).
- To fetch parameters table from Synapse (data warehouse) □(**DS_storedprocedure**).
- To insert data into SQL tables in Synapse (data warehouse) □(**DS_synapse_induja**).
- To fetch cases_deaths.csv file from storage account (data lake) □(**DS_source_adls**).
- To insert data transformation done file by using dataflow into storage account (data lake) □(**DS_sink_adls**)

The screenshot shows the Microsoft Azure Synapse Analytics workspace interface. The left sidebar navigation menu includes options like Synapse live, Validate all, Publish all, Analytics pools, SQL pools (which is selected and highlighted in blue), Apache Spark pools, Data Explorer pools (prev...), External connections, Linked services, Microsoft Purview, Integration (Triggers, Integration runtimes), Security, Access control, Credentials, and Managed private endpoints. The main content area is titled "SQL pools" and contains the following text: "The serverless SQL pool, Built-in, is immediately available for your workspace. Dedicated SQL pools can be configured to adapt to team or organizational requirements and constraints." Below this is a button bar with "+ New", "Refresh", and a "Filter by name" input field. A table lists two items: "Built-in" (Serverless, Online, Auto) and "indujadicatedpool" (Dedicated, Online, DW100c). At the top of the page, there is a header bar with the workspace name "covid-workspace-induja", user information "19699A0428@mits.ac.in MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE", and standard browser controls (Accept, Reject, More options, X).

Step 12: Before creating pipeline in data factory, we need **to turn on the dedicated pool**. we need to check these two to three times while moving on to creation of pipeline.



Step 13: Create a pipeline (Pipeline_induja) and Drag and Drop the Look up Activity into pipeline workspace and set the source dataset (DS_storedprocedure) for Lookup and choose the option stored procedure and given the stored procedure name created in synapse (data warehouse).



Step 14: Drag and Drop **For each** activity in pipeline workspace and configure the for each activity settings like **Items** with output of look up activity (**@activity('Lookup1').Output.Value**).

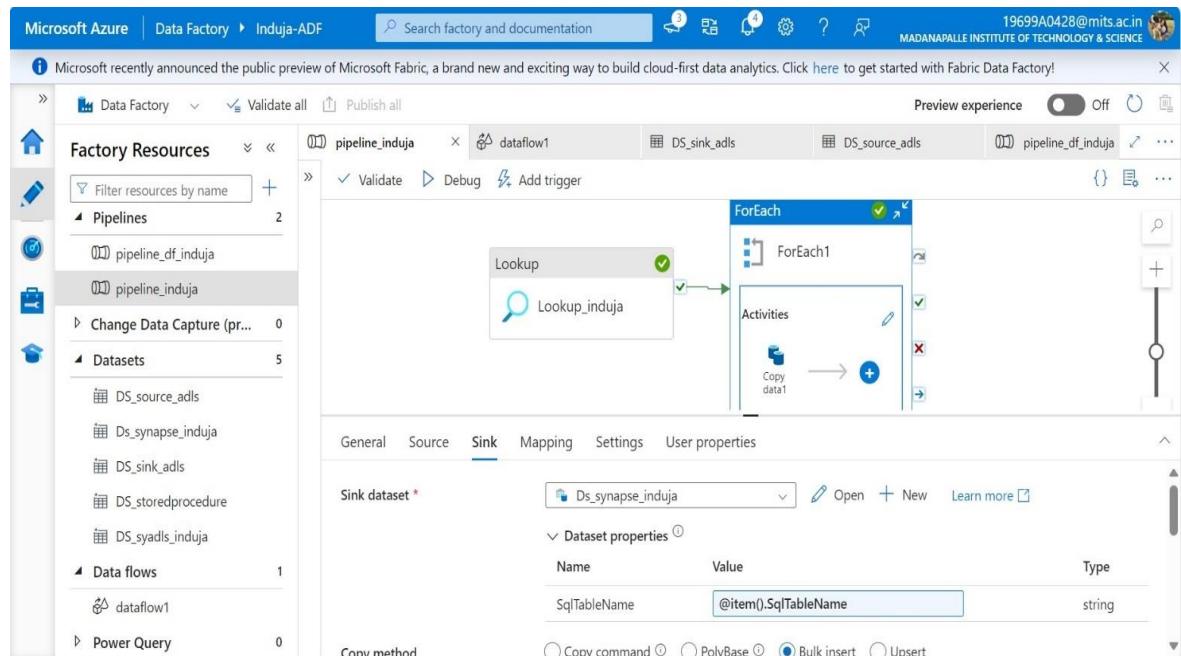
This screenshot shows the Microsoft Azure Data Factory pipeline workspace. A 'Copy data' activity is selected within a 'ForEach' loop. The 'Source' tab is active, showing the configuration for the 'Source dataset' (DS_adlsV2Dataset) and 'Dataset properties'. The 'Source dataset' dropdown is set to 'DS_adlsV2Dataset'. Under 'Dataset properties', there are two entries: 'FolderName' with the value '@{item().FolderName}' and 'FileName' with the value '@{item().FileName}'. The 'File path type' is set to 'File path in dataset'. The 'Source' tab also includes sections for 'Start time (UTC)' and 'End time (UTC)'.

This screenshot shows the Microsoft Azure Data Factory pipeline workspace. A 'ForEach' loop is selected, containing a 'Lookup' activity and a 'Copy data' activity. The 'Source' tab is active, showing the configuration for the 'Source dataset' (DS_syadls_induja) and 'Dataset properties'. The 'Source dataset' dropdown is set to 'DS_syadls_induja'. Under 'Dataset properties', there are two entries: 'FolderName' with the value '@{item().FolderName}' and 'FileName' with the value '@{item().FileName}'. The 'File path type' is set to 'File path in dataset'. The 'Source' tab also includes sections for 'Start time (UTC)' and 'End time (UTC)'. The pipeline resources sidebar on the left lists various pipelines, datasets, and data flows.

Step 15: Click on add activity symbol present on foreach activity and inside foreach activity add a **Copy activity** for copy data from CVS file into SQL table.

Configure settings at source side in copy activity by giving dataset (**DS_syadls_induja**) and giving folder name and file name dynamic by taking from foreach activity by **item**.

Folder Name (@ {item (). FolderName}), File Name (@ {item () . FileName})



Step 16: Configure setting in copy activity at sink side by giving dataset (**DS_synapse_induja**) and giving sqltableName dynamically by taking from foreach activity by item.SqlTableName (@ {item () . SqlTableName})

Microsoft Azure | Data Factory > covidadfc3t3

Search factory and documentation

Preview experience: Off

Factory Resources

- Pipelines: PL_ADFCovidUseCase (1)
- Datasets: DS_adlsV2Dataset, DS_configDataSet, DS_casesanddeath_source, DS_max_daily_deaths_target, DS_dwdataset (5)
- Data flows: dataflow1 (1)
- Power Query: 0

Pipeline status: Succeeded

Pipeline run ID: e67788ba-eb41-43d1-80d4-36c7fc432632

Activity name	Activity status	Activity type	Run start	Duration	Integration runtime
Copy data1	Succeeded	Copy data	3/6/2024, 3:59:19 PM	15s	AutoResolveIntegration
Copy data1	Succeeded	Copy data	3/6/2024, 3:59:19 PM	15s	AutoResolveIntegration
Copy data1	Succeeded	Copy data	3/6/2024, 3:59:19 PM	17s	AutoResolveIntegration
Copy data1	Succeeded	Copy data	3/6/2024, 3:59:19 PM	29s	AutoResolveIntegration
Copy data1	Succeeded	Copy data	3/6/2024, 3:59:19 PM	14s	AutoResolveIntegration
Data flow1	Succeeded	Data flow	3/6/2024, 3:59:51 PM	59s	debugpool-8Cores-Ger
ForEach1	Succeeded	ForEach	3/6/2024, 3:59:18 PM	32s	
Lookup1	Succeeded	Lookup	3/6/2024, 3:59:13 PM	4s	AutoResolveIntegration

Microsoft Azure | Data Factory > Induja-ADF

Search factory and documentation

Preview experience: Off

Factory Resources

- Pipelines: pipeline_df_induja (2), pipeline_induja (1)
- Datasets: DS_source_adls, Ds_synapse_induja, DS_sink_adls, DS_storedprocedure, DS_syadls_induja (5)
- Data flows: dataflow1 (1)
- Power Query: 0

Pipeline status: Succeeded

Pipeline run ID: 19699A0428@mits.ac.in

Activity name	Activity status	Activity type	Run start	Duration	Integration
Copy data1	Succeeded	Copy data	3/9/2024, 9:53:29 PM	17s	AutoResolv
Copy data1	Succeeded	Copy data	3/9/2024, 9:53:29 PM	29s	AutoResolv

Step 17: After setting whole pipeline by using Lookup and Foreach activity recheck all parameters given in each configuration setting, check the dedicated pool is turn on and then turn on the debug option in pipeline. Finally, all the activities are successfully completed.

```

1 select * from AlloverDeaths
2 WHERE country = 'India' AND
3 [indicator]='confirmed cases' AND
4 month([date])=3 AND
5 year([date]) = '2020';
6
7
8 SELECT country,[indicator],daily_count,[date] from AlloverDeaths
9 WHERE [indicator]='confirmed cases' and
10 year([date]) = '2020'
11 GROUP BY country,indicator,daily_count,[date]
12 ORDER by country asc,date asc;

```

Results Messages

View Table Chart Export results

Search

00:00:28 Query executed successfully.

Step 18: After successfully run of pipeline now we need check the data inserted into tables in data warehouse by preforming two SQL queries operation given in project documentation.

```

1 select * from AlloverDeaths

```

country	country_code	continent	population	indicator	daily_count	date	rate_14_day	source
India	IND	Asia	1380004385	confirmed cases	0	2020-03-01T00:00:00	0.000000	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	0	2020-03-02T00:00:00	0.000000	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	2	2020-03-03T00:00:00	0.000144	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	1	2020-03-04T00:00:00	0.000217	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	22	2020-03-05T00:00:00	0.001811	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	1	2020-03-06T00:00:00	0.001884	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	2	2020-03-07T00:00:00	0.002028	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	3	2020-03-08T00:00:00	0.002246	Epidemic intelli...

00:00:05 Query executed successfully.

Step 19: After running first and second SQL Query written in SQL script and it is successfully completed and given output as per the query.

ID	country	indicator	daily_count	date	rate_14	source
56023	India	confirmed cases	0	3/1/2020	0	Epidemic intelligence, national daily data
56024	India	confirmed cases	0	3/2/2020	0	Epidemic intelligence, national daily data
56025	India	confirmed cases	2	3/3/2020	0.000145	Epidemic intelligence, national daily data
56026	India	confirmed cases	1	3/4/2020	0.000217	Epidemic intelligence, national daily data
56027	India	confirmed cases	22	3/5/2020	0.001812	Epidemic intelligence, national daily data
56028	India	confirmed cases	1	3/6/2020	0.001884	Epidemic intelligence, national daily data
56029	India	confirmed cases	2	3/7/2020	0.002029	Epidemic intelligence, national daily data
56030	India	confirmed cases	3	3/8/2020	0.002246	Epidemic intelligence, national daily data
56031	India	confirmed cases	0	3/9/2020	0.002246	Epidemic intelligence, national daily data
56032	India	confirmed cases	10	#####	0.002971	Epidemic intelligence, national daily data
56033	India	confirmed cases	6	#####	0.003406	Epidemic intelligence, national daily data
56034	India	confirmed cases	23	#####	0.005072	Epidemic intelligence, national daily data
56035	India	confirmed cases	2	#####	0.005217	Epidemic intelligence, national daily data
56036	India	confirmed cases	8	#####	0.005797	Epidemic intelligence, national daily data
56037	India	confirmed cases	7	#####	0.006304	Epidemic intelligence, national daily data
56038	India	confirmed cases	3	#####	0.006522	Epidemic intelligence, national daily data
56039	India	confirmed cases	32	#####	0.008696	Epidemic intelligence, national daily data

Step 20: compare the output appear for previous SQL query written in SQL script with original CSV file in excelsheet.

country	indicator	daily_count	date
Afghanistan	confirmed cases	0	2020-01-02T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-03T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-04T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-05T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-06T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-07T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-08T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-09T00:00:00.000000
Afghanistan	confirmed cases	0	2020-01-10T00:00:00.000000

Step 21: Successfully query is done and given output as per the query.

The screenshot shows an Excel spreadsheet titled "cases_deaths.csv". The table contains 18 rows of data, each representing a record from March 8, 2024. The columns are labeled: country, indicator, daily_ci, date, rate_14, and source. The data shows 38928341 confirmed cases for Afghanistan (Asia) across various dates in March 2020, with the source being "Epidemic intelligence, national daily data".

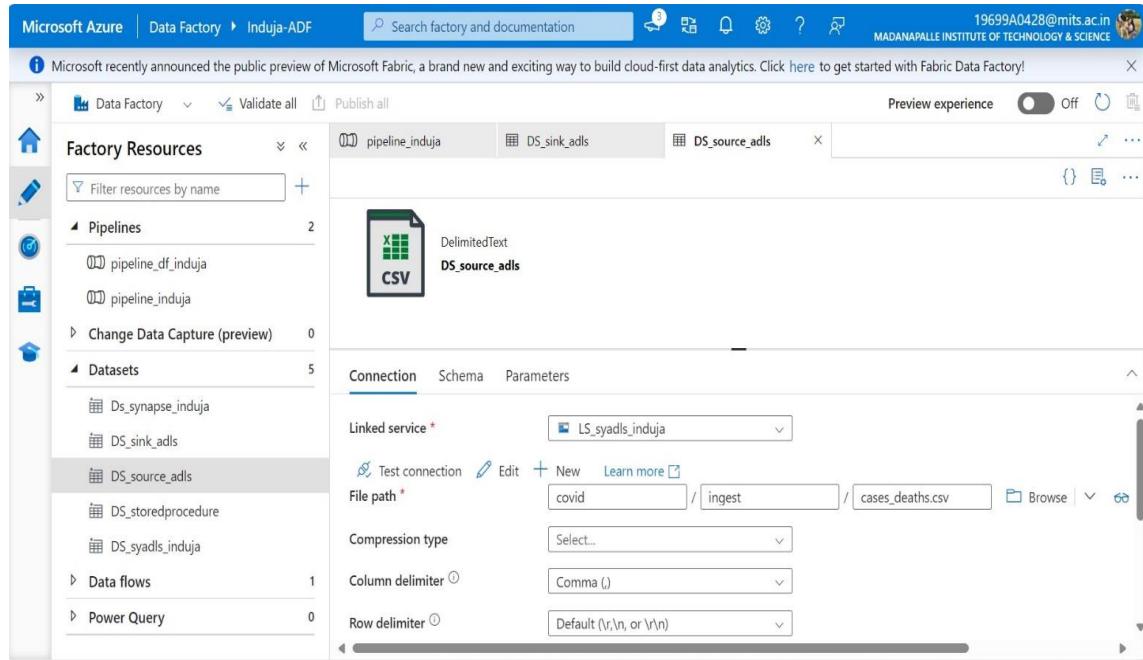
	country	country	contine	population	indicator	daily_ci	date	rate_14	source
1	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/2/2020		Epidemic intelligence, national daily data
2	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/3/2020		Epidemic intelligence, national daily data
3	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/4/2020		Epidemic intelligence, national daily data
4	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/5/2020		Epidemic intelligence, national daily data
5	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/6/2020		Epidemic intelligence, national daily data
6	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/7/2020		Epidemic intelligence, national daily data
7	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/8/2020		Epidemic intelligence, national daily data
8	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/9/2020		Epidemic intelligence, national daily data
9	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/10/2020		Epidemic intelligence, national daily data
10	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/11/2020		Epidemic intelligence, national daily data
11	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/12/2020		Epidemic intelligence, national daily data
12	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/13/2020		Epidemic intelligence, national daily data
13	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/14/2020		Epidemic intelligence, national daily data
14	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/15/2020		Epidemic intelligence, national daily data
15	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/16/2020		0 Epidemic intelligence, national daily data
16	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/17/2020		0 Epidemic intelligence, national daily data
17	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/18/2020		0 Epidemic intelligence, national daily data
18	Afghanist	AFG	Asia	38928341	confirmed cases	0	1/19/2020		0 Epidemic intelligence, national daily data

Step 22: Compare the above SQL query output with data which is appear from CSV file which is open in excel for reference check.

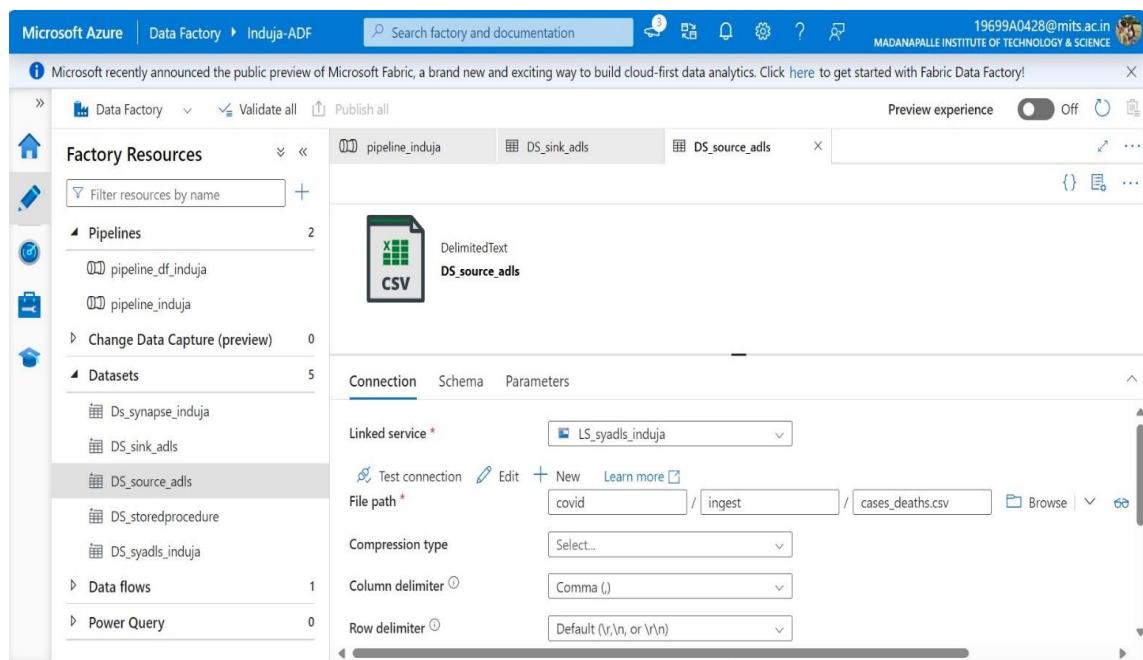
The screenshot shows the Microsoft Azure Storage Explorer interface. The left sidebar shows navigation options like Home, covidadlsinduja, Overview, Activity log, Tags, Diagnose and solve problems, Access Control (IAM), Data migration, Events, and Storage browser. The main area is titled "covidadlsinduja | Containers" and shows a list of containers. There is a search bar at the top and a table below it. The table has columns: Name, Last modified, Anonymous access level, and Lease state. The containers listed are \$logs, covid, and transformedpath, all of which were created on March 8, 2024, at different times, and are set to Private anonymous access.

Name	Last modified	Anonymous access level	Lease state
\$logs	3/8/2024, 11:01:03 AM	Private	Available
covid	3/8/2024, 11:03:21 AM	Private	Available
transformedpath	3/8/2024, 1:55:51 PM	Private	Available

Step 23: Create another container with name “**transformedpath**” for second requirement given in project for storing transformed data file by using data flow.



The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists Pipelines, Datasets, and Power Query. Under Datasets, 'DS_source_adls' is selected. The main workspace displays a pipeline named 'pipeline_induja'. Within this pipeline, there is a dataset named 'DS_source_adls' which is a DelimitedText CSV file. The 'Connection' tab is active, showing the linked service 'LS_syadls_induja' and the file path 'covid / ingest / cases_deaths.csv'. Other tabs include 'Schema' and 'Parameters'.



This screenshot is identical to the one above, showing the Microsoft Azure Data Factory interface. The 'Factory Resources' sidebar and the pipeline configuration for 'pipeline_induja' are the same. The dataset 'DS_source_adls' is selected in the workspace, and its properties are displayed in the 'Connection' tab, including the linked service 'LS_syadls_induja' and the file path 'covid / ingest / cases_deaths.csv'.

Step 24: Create source dataset (**DS_source_adls**) for dataflow by giving a file specific filename on which data transformation need to be taken place as per project requirement.

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists various datasets, pipelines, and data flows. In the center, a dataset named 'DS_source_adls' is being edited. The 'Connection' tab is active, showing a 'Linked service' dropdown set to 'LS_syadls_induja'. Below it, 'File path' is set to 'transformedpath / Directory / File name'. Other settings like 'Compression type', 'Column delimiter', and 'Row delimiter' are also visible.

Step 25: Create target dataset (**DS_sink_adls**) for dataflow to keep that data transformpath file in specific place for further use.

The screenshot shows the Microsoft Azure Data Factory interface with a data flow named 'dataflow1' selected. The data flow diagram shows a sequence of operations: 'source1' (with 9 total columns), followed by 'pivot1' (described as pivoting row values into columns, grouping columns and aggregating data), then 'aggregate1' (described as aggregating data by 'continent' producing columns 'maximum'), and finally 'rank1' (described as ranking rows on columns 'maximum'). Below the diagram, the 'Source settings' tab is active, showing 'Source type' as 'Dataset' and 'Dataset' selected in the dropdown. Other tabs include 'Source options', 'Projection', 'Optimize', 'Inspect', and 'Data preview'. The 'Dataset' dropdown in the 'Source settings' section also lists 'DS_source_adls'.

Step 26: Develop dataflow by using some transformations like source, filter, pivot, aggregate, rank and sink as per the question given in project documentation.

The screenshot shows the Microsoft Azure Data Factory interface. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (pipeline_induja, pipeline_df_induja), 'Datasets' (DS_source_adls, DS_sink_adls, DS_storedprocedure, DS_syadls_induja), and 'Data flows' (dataflow1). The main workspace is titled 'pipeline_induja' and contains a data flow named 'dataflow1'. The data flow consists of several stages: 'source1' (Import data from DS_source_adls), 'pivot1' (Columns: 9 total), 'aggregate1' (Aggregating data by 'continent' producing columns 'maximum'), 'rank1' (Ranking rows on columns 'maximum'), and a final stage (labeled 'Ex'). Below the data flow, the 'Pivot settings' tab is selected, showing steps 1. Group by, 2. Pivot key, and 3. Pivoted columns. Under 'Columns', there are two entries: 'abc_country' mapped to 'country' and 'abc_country_code' mapped to 'country_code'.

Step 27: After successfully creating dataflow, I click on dataflow debug output appear as per the question.

Step 28: Drag and drop the data flow into the pipeline workspace and connect with before activity to run after successfully complication of before activity.

The screenshot shows the Microsoft Azure Data Factory interface. The 'Activities' section is open, showing a search bar with 'dat' and a list of activities: 'Move and transform' (Copy data, Data flow), 'Azure Data Explorer', 'Databricks' (Notebook, Jar, Python), and 'Data Lake Analytics'. A 'Data flow' item is selected. The main workspace shows a pipeline named 'pipeline_induja' with a single activity named 'Data flow1'. The 'Output' tab is selected, displaying a message: 'Data flow activity for this debug run will start as soon as the data flow debug session is ready.' Below this, a table shows one item: 'Activity name' (Data flow1), 'Activity status' (Succeeded), 'Activity type' (Data flow), 'Run start' (3/8/2024, 2:28:42 PM), and a 'D' column.

Step 29: Dataflow is completed successfully after click on debug as how in above picture

The screenshot shows the Microsoft Azure Storage Blob view for the file 'covid_analysis.csv'. The file was uploaded to the 'transformedpath' container under the 'covidadlsinduja' account. The file details are displayed, including its name, type (CSV), size (1.1 KB), and last modified date (2023-09-12). The file content is shown as a table:

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

Below the table is a blue 'Edit' button.

Step 30: After that we need to check file is appear in the transformation container in storage account (data lake). I successfully got that file in my container as per the question given in the project documentation.

Conclusion:

Requirement 1 Output:

The screenshot shows the Microsoft Azure Data Factory pipeline editor. On the left, the 'Factory Resources' sidebar lists 'Pipelines' (pipeline_induja), 'Datasets' (DS_source_adls, DS_synapse_induja, DS_sink_adls, DS_storedprocedure, DS_syadls_induja), and 'Data flows' (dataflow1). The main workspace displays a pipeline named 'pipeline_induja'. It contains a 'Lookup' activity followed by a 'ForEach' activity. The 'ForEach' activity has a child 'ForEach1' activity. Below the activities, the 'Output' tab is selected, showing a table of pipeline status and activity details. The table shows two 'Copy data1' activities, both succeeded, with run start times of 3/9/2024, 9:53:29 PM and durations of 17s and 29s respectively. The pipeline status is 'Succeeded'.

Activity name	Activity status	Activity type	Run start	Duration	Integration
Copy data1	Succeeded	Copy data	3/9/2024, 9:53:29 PM	17s	AutoResolve
Copy data1	Succeeded	Copy data	3/9/2024, 9:53:29 PM	29s	AutoResolve

The screenshot shows the Microsoft Azure Synapse Analytics results page. The top navigation bar includes 'Synapse live', 'Validate all', 'Publish all', 'Run', 'Undo', 'Publish', 'Query plan', 'Connect to', 'indujadedicatedpool', 'Use database', 'indujadedicatedpool', and a refresh button. The main area is titled 'SQL script 1_validat...' and shows a table with the following data:

country	indicator	daily_count	date
Afghanistan	confirmed cases	0	2020-01-02T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-03T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-04T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-05T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-06T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-07T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-08T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-09T00:00:00.0000000
Afghanistan	confirmed cases	0	2020-01-10T00:00:00.0000000

At the bottom, a message indicates '00:00:28 Query executed successfully.'

Synapse live Validate all Publish all

SQL script 1_validati... ● SQL script 1_induja

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 indujadicatedpool Use database indujadicatedpool

```
1 select * from AlloverDeaths
```

Results Messages

View Table Chart Export results

Search

country	country_code	continent	population	indicator	daily_count	date	rate_14_day	source
India	IND	Asia	1380004385	confirmed cases	0	2020-03-01T00...	0.000000	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	0	2020-03-02T00...	0.000000	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	2	2020-03-03T00...	0.000144	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	1	2020-03-04T00...	0.000217	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	22	2020-03-05T00...	0.001811	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	1	2020-03-06T00...	0.001884	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	2	2020-03-07T00...	0.002028	Epidemic intelli...
India	IND	Asia	1380004385	confirmed cases	3	2020-03-08T00...	0.002246	Epidemic intelli...

00:00:05 Query executed successfully.

Requirement 2 Output:

Microsoft Azure Search resources, services, and docs (G+)

Home > covidadlsinduja | Containers > transformedpath >

covid_analysis.csv ...

Blob

Save Discard Download Refresh Delete

Overview Versions Edit Generate SAS

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

Edit