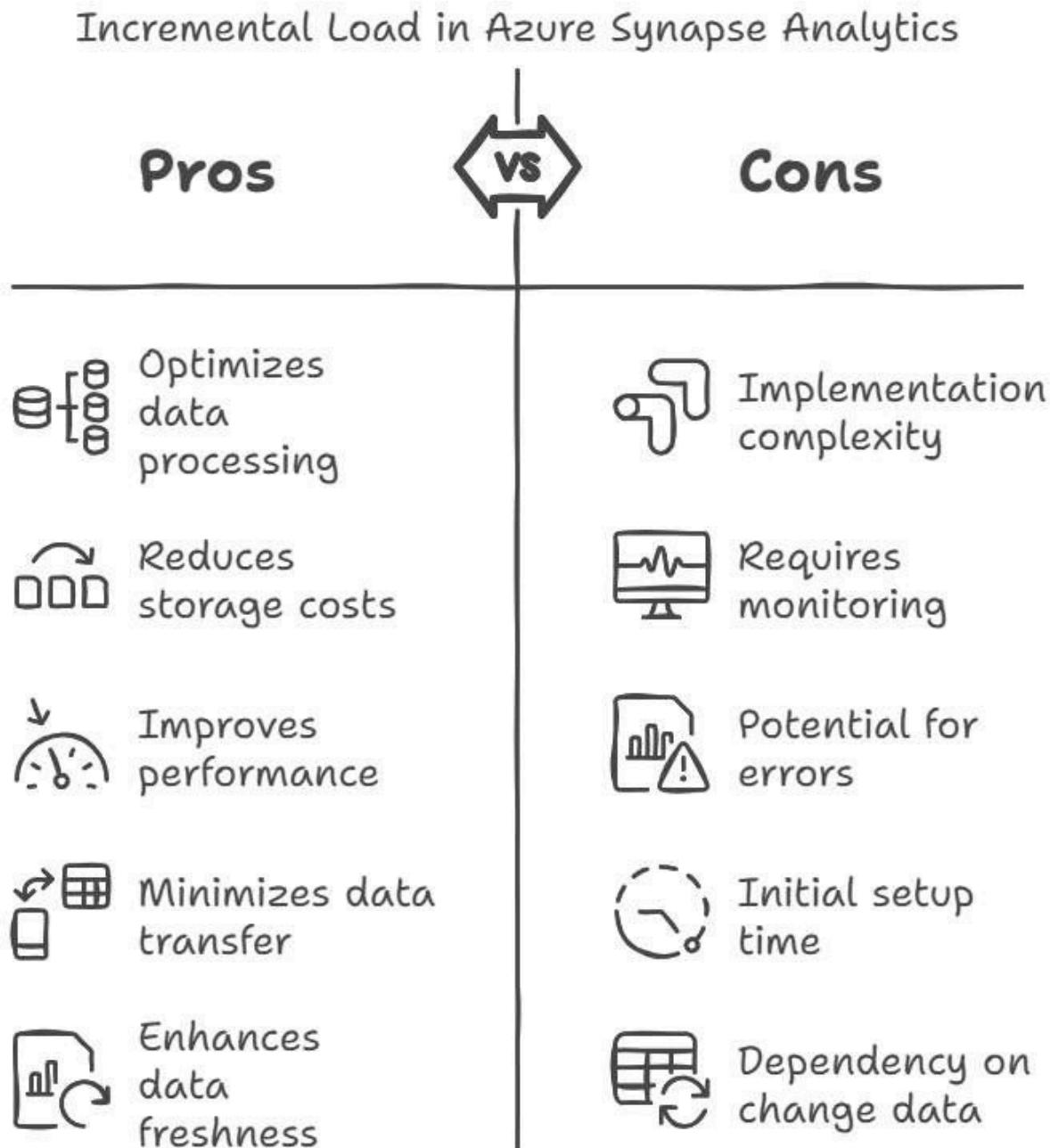


INCREMENTAL LOADING CONCEPT



Azure
Synapse
Analytics

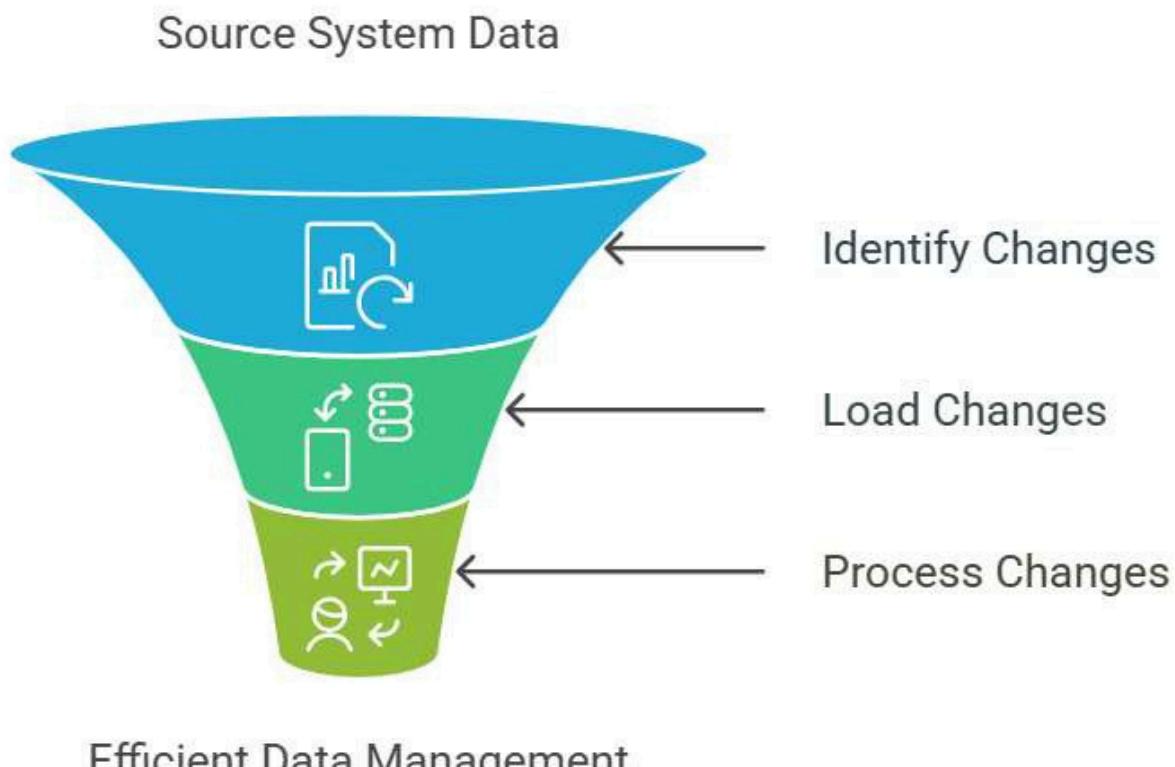
Incremental load is a crucial concept in data integration and ETL processes, particularly when working with large datasets in **Azure Synapse Analytics**.



What is the Incremental Load?

Incremental load refers to the process of loading only the new or changed data from a source system into a target system, rather than reloading the entire dataset. This approach is particularly beneficial in scenarios where data volumes are large, and the cost of transferring and processing data can be significant. By focusing on just the changes, organizations can save time, reduce resource consumption, and improve overall efficiency.

Efficient Data Transfer Process



Scenario

Assume you have an online retail store, and you have a lot of data being stored, which needs to be analyzed. There is an example case of understanding the changes in the data over period. Step 01 – Create 5 sample tables I have created multiple tables as follows:

Customer data table (To get the details of the customer) Create Customer id, Name , Phone number, Customer_datetime

SQL code-

```
--Customer data table (To get the detailsofthecustomer      )
CREATE TABLE Customer (
    CustomerID INT,
    Name VARCHAR(100) NOT NULL,
    Phone VARCHAR(20),
    Customerupdateddate DATETIME      ---DeltaColumn
);
```

Customer login table (To check the time spent online and on what products) Create Login id, Username, password and login_datetime

SQL code-

```
--Customer login table (To check the time spentonlineandonwhatproducts)
CREATE TABLE Login_id (
    LoginID INT,
    Username VARCHAR(50) UNIQUE NOT NULL,
    Password VARCHAR(255) NOT NULL,
    Updatedlogindata DATETIME      ---DeltaColumn
);
```

Payment table (To get the list of the transactions/payments completed) Create Transaction_id, Customer ID, Product ID and Transaction_datetime

SQL code-

```
--Payment table (To get the list of the transactions/payments completed)
CREATE TABLE Transactions (
    TransactionID INT PRIMARY KEY,      ---DeltaColumn
    CustomerID INT NOT NULL,
    ProductID INT NOT NULL,
    TransactionDate DATE NOT NULL,
);
```

Inventory table (To get the list of items in the inventory) Create Product_id, Product_name, Price, Quantity.

SQL code-

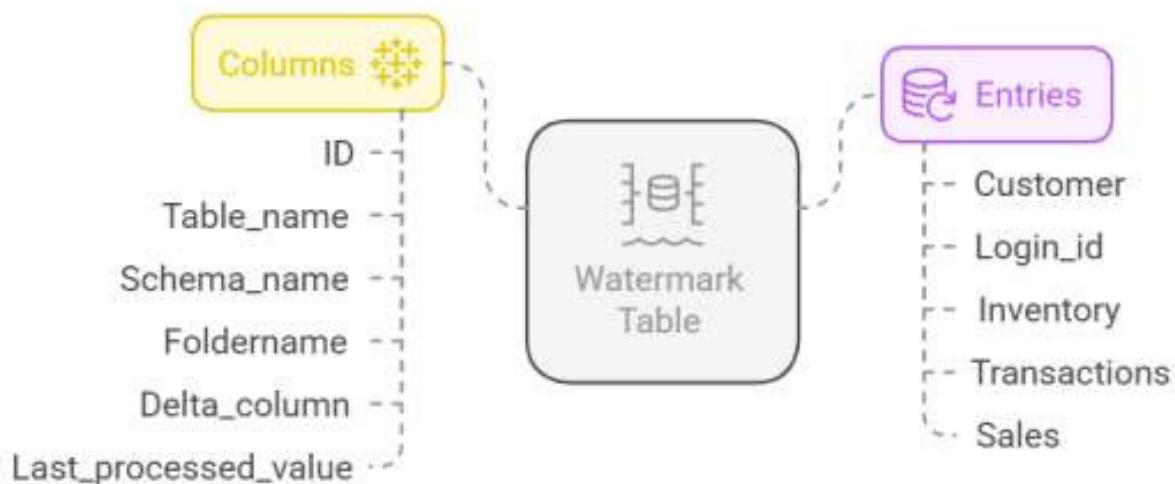
```
--Inventory table (To get the list of items in the inventory)
CREATE TABLE Inventory (
    ProductID INT PRIMARY KEY, ---DeltaColumn
    ProductName VARCHAR(200) NOT NULL,
    Quantity INT NOT NULL CHECK (Quantity >= 0),
    Price DECIMAL(10, 2) NOT NULL
);
```

Sales table (To check the sales of the items) Create Sales ID, Product ID, Sales_log, Revenue

SQL code-

```
--Sales table (To check the salesoftheitems)
CREATE TABLE Sales (
    SalesID INT PRIMARY KEY,
    ProductID INT NOT NULL,
    Sales_log DATETIME, ---DeltaColumn
    Revenue DECIMAL(15, 2) NOT NULL
);
```

Watermark Table Structure and Initialization



Insert values/data into the tables

```
INSERT INTO Customer (CustomerID, Name, Phone, Customerupdateddate)
VALUES
(1, 'John Doe', '123-456-7890', '2023-01-01 10:00:00'),
(2, 'Jane Smith', '987-654-3210', '2023-02-01 14:30:00'),
(3, 'Bob Johnson', NULL, '2023-03-01 09:15:00');
```

```
INSERT INTO Login_id (LoginID, Username, Password, Updatedlogindata)
VALUES
(1, 'johndoe', 'password123', '2023-01-01 10:00:00'),
(2, 'janeshsmith', 'securepass', '2023-02-01 14:30:00'),
(3, 'bobjohnson', 'secret123', '2023-03-01 09:15:00');
```

```
INSERT INTO Inventory (ProductID, ProductName, Quantity, Price)
VALUES
(1, 'Laptop', 50, 999.99),
(2, 'Smartphone', 100, 699.99),
(3, 'Headphones', 75, 149.99);
```

```
INSERT INTO Transactions (TransactionID, CustomerID, ProductID, TransactionDate)
VALUES
(1, 1, 1, '2023-01-05'),
(2, 2, 2, '2023-02-10'),
(3, 3, 3, '2023-03-15');
```

```
INSERT INTO Sales (SalesID, ProductID, Sales_log, Revenue)
VALUES (1, 1, '2023-01-05 10:00:00', 999.99),
(2, 2, '2023-02-10 14:30:00', 699.99),
(3, 3, '2023-03-15 09:15:00', 299.98);
```

Step 02 – Create a watermark Table

This step helps to monitor the changes in the data i.e. it may be data entries, data modifications, etc.

SQL Code-

```
--Create a Watermark table
CREATE TABLE Watermark (
    ID INT PRIMARY KEY,
    Unique_Value VARCHAR(100) --Can't accept similar id's or NULL -- Only
    Table_name VARCHAR(100),
    Schema_name VARCHAR(100),
    Foldername VARCHAR(50),
    Delta_column VARCHAR(100),
    Last_processed_value VARCHAR(255) NOT NULL
);
```

Inserted data into the watermark table

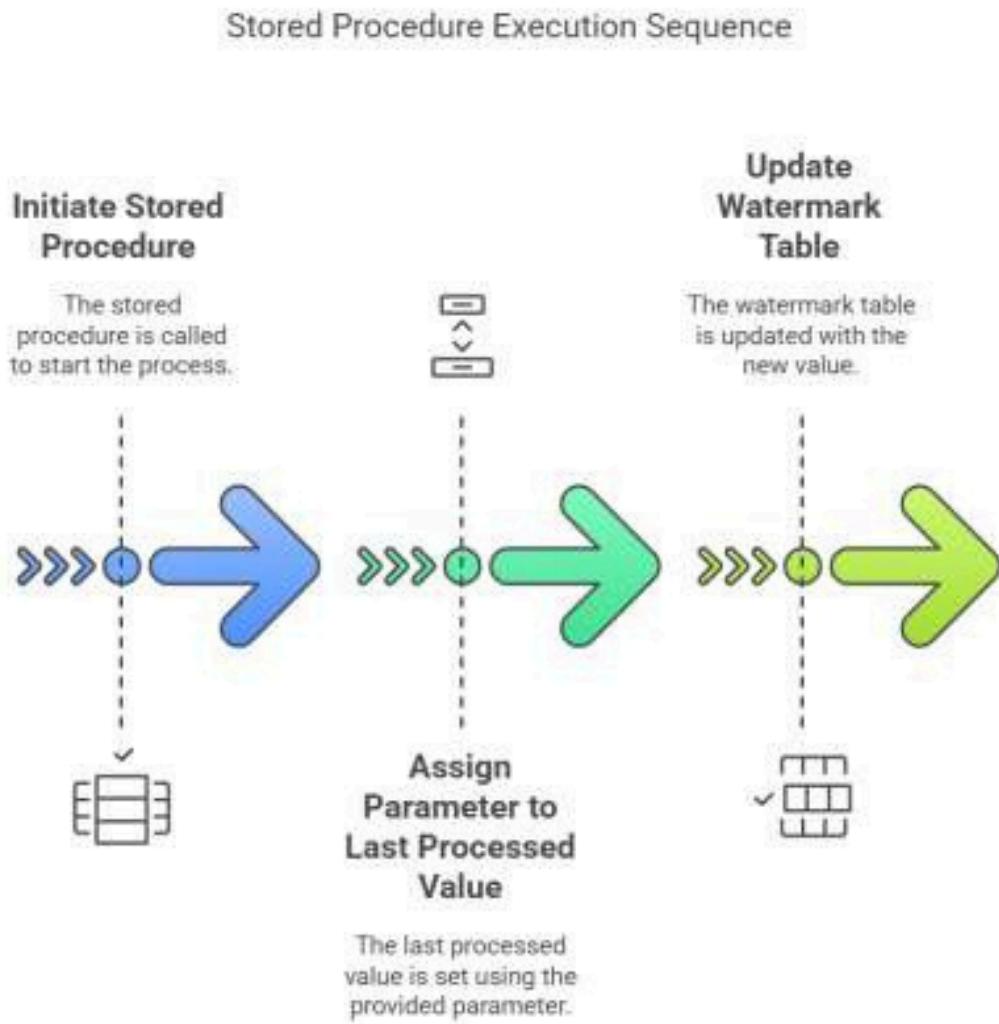
```
-- Initialize watermark entries
INSERT INTO Watermark
VALUES
(1,'Customer','dbo','RetailDB/Customer_data','Customerupdateddate','1900-01-01 00:00:00'),
(2,'Login_id','dbo','RetailDB/Login_id_data','Updatedlogindata','1900-01-01 00:00:00'),
(3,'Inventory','dbo','RetailDB/Inventory_data','ProductID','0'),
(4,'Transactions','dbo','RetailDB/Transactions_data','TransactionID','0'),
(5,'Sales','dbo','RetailDB/Sales_data','Sales_log','1900-01-01 00:00:00')
```

Now, the watermark table is ready to be used.

Step 03 – Create a stored procedure

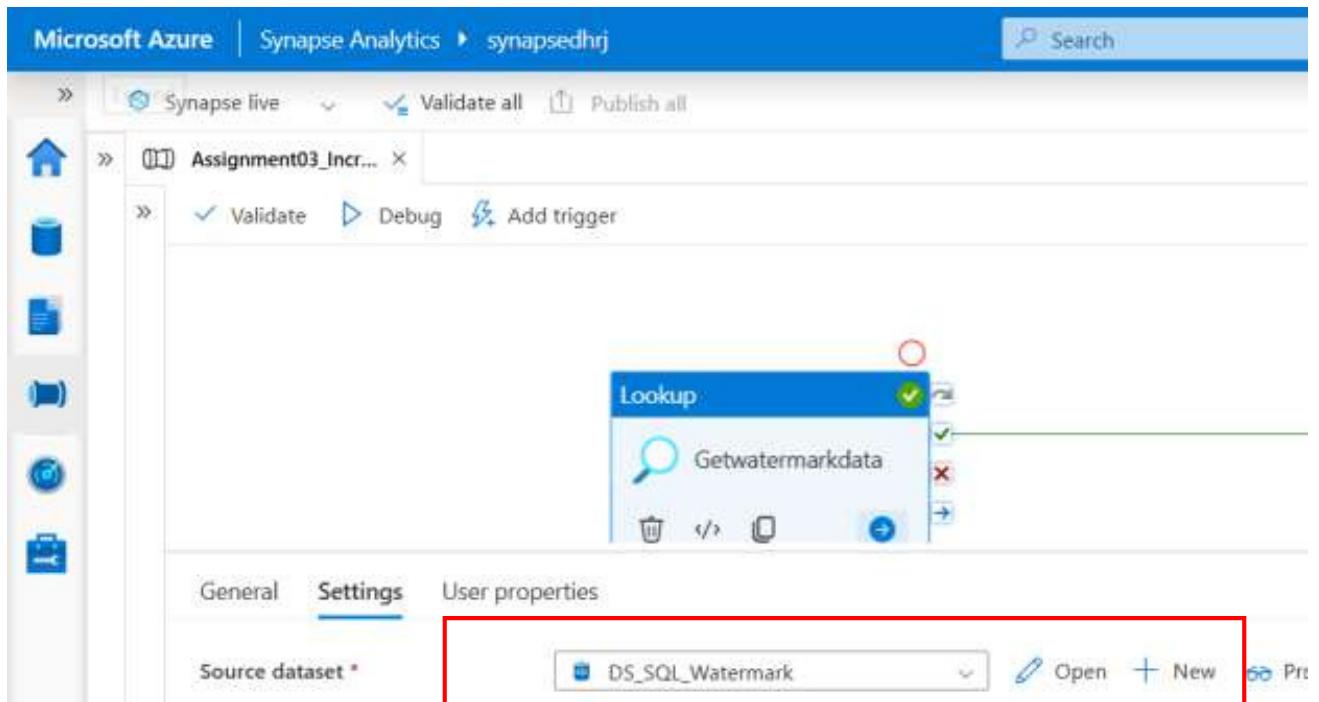
This helps to assign a parameter to the changes in the last processed values corresponding to the table names.

```
--CreateStored procedure
CREATE PROC USP_Watermark_RetailDB
@lpv VARCHAR(100),
@TBname VARCHAR(100)
AS
BEGIN
    UPDATE Watermark
    SET Last_processed_value=@lpv WHERE Table_name=@TBname
END
```



Step 04 – Creation of pipeline in Synapse

- Create a lookup activity and named it as Getwatermarkdata. Click on source dataset and create a new linked service with SQL server and name
- Its as DS_SQL_Watermark.



- Open the DS_SQL_Watermark dataset.
- Create 2 parameters i.e. Schemaname and Tablename for dynamically assigning the Schemaname and Tablename.

Microsoft Azure | Synapse Analytics > synapsedhrj

Search

Synapse live Validate all Publish all

Assignment03_Incr... DS_SQL_Watermark

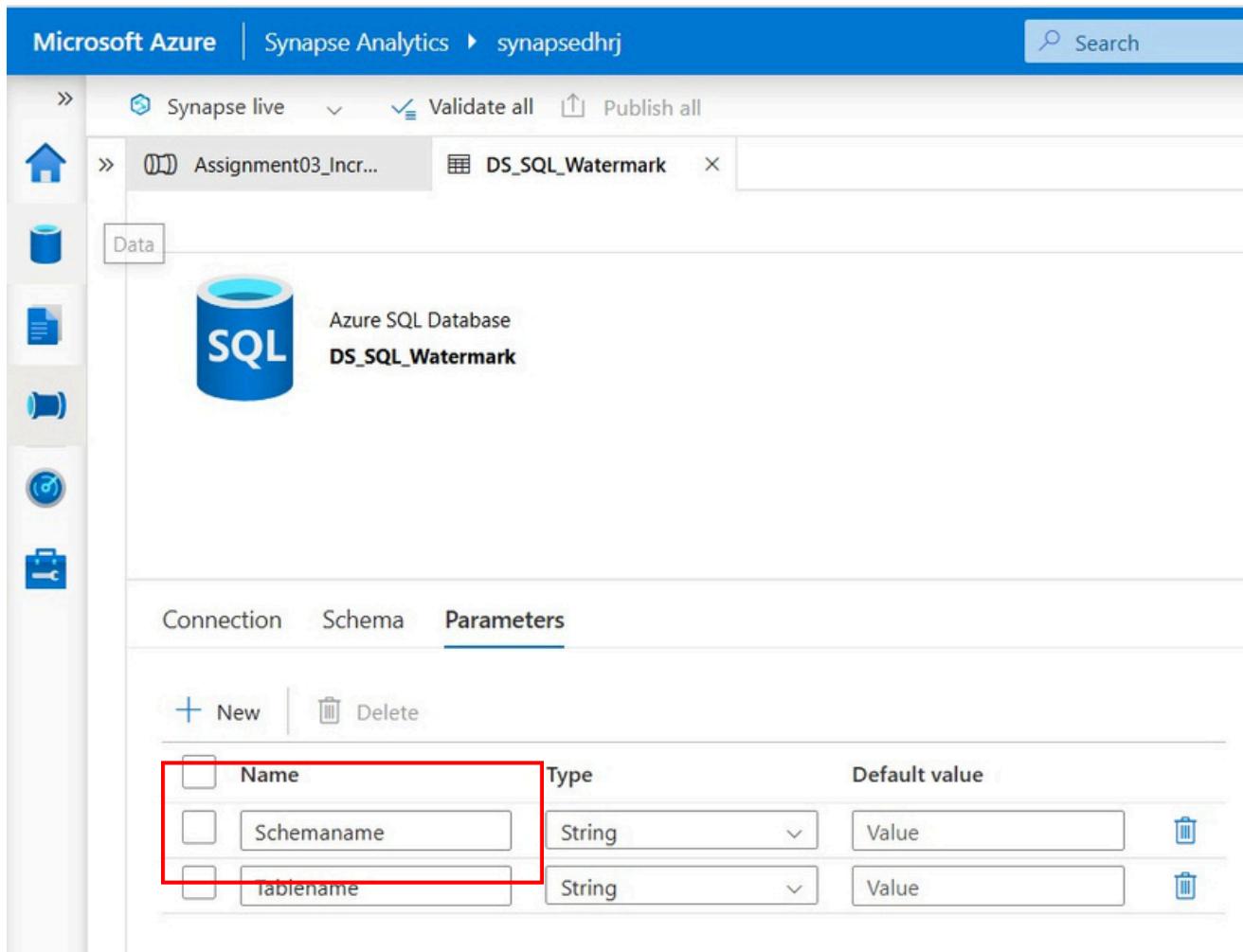
Data

Azure SQL Database
DS_SQL_Watermark

Connection Schema **Parameters**

+ New | Delete

Name	Type	Default value
Schemaname	String	Value
Tablename	String	Value



- Assigning dynamic variables to the Schema and Table name for the table

The screenshot shows the Microsoft Azure Synapse Analytics studio interface. At the top, the navigation bar displays "Microsoft Azure" and "Synapse Analytics > synapsedhrj". Below the navigation bar, there are tabs for "Synapse live", "Validate all", and "Publish all". A search bar is located at the top right.

The main workspace shows a dataset named "Assignment03_Incr..." which is connected to an "Azure SQL Database" named "DS_SQL_Watermark". On the left side, there is a vertical toolbar with icons for Home, Databases, Tables, Views, and Scripts.

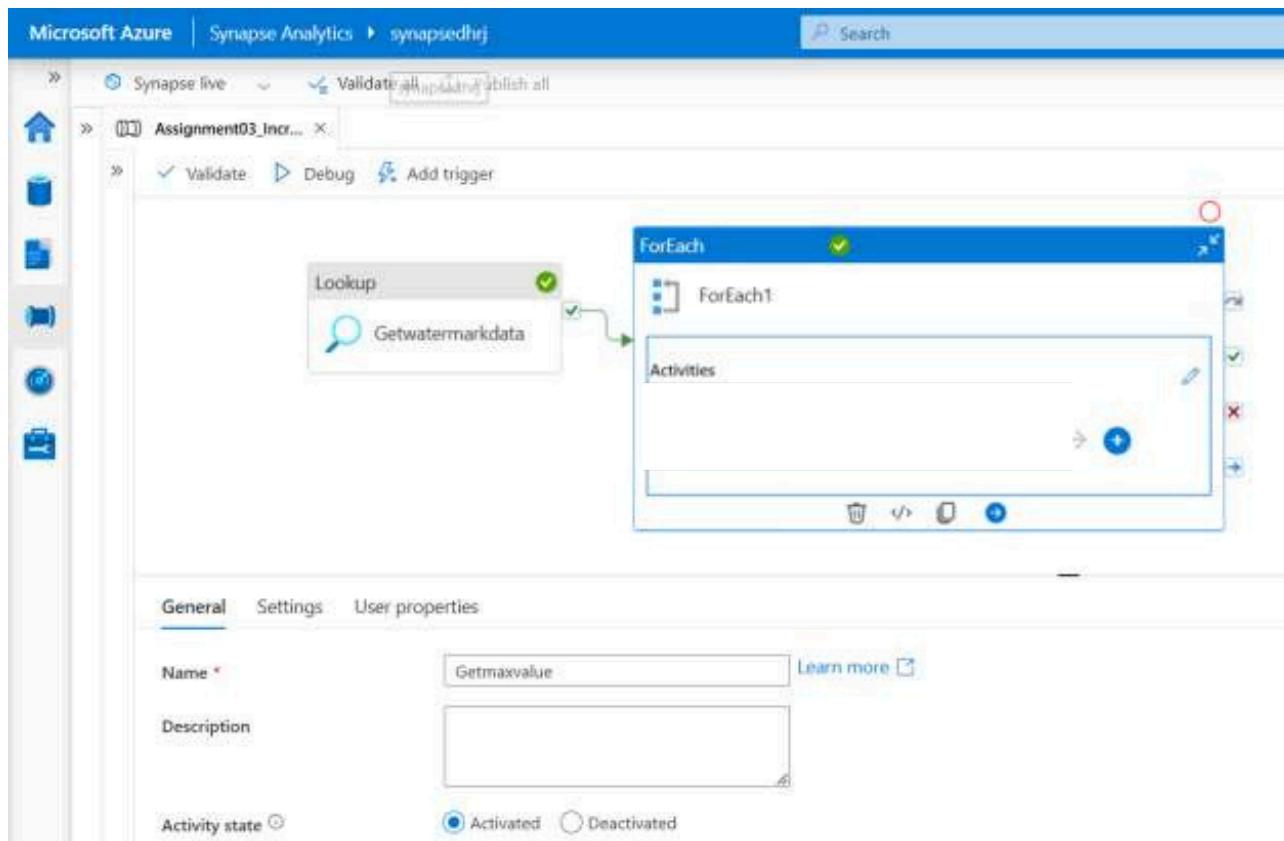
The dataset configuration pane has three tabs: "Connection", "Schema", and "Parameters". The "Connection" tab is selected. It shows the "Linked service" dropdown set to "AzureSqlDatabase1", the "Integration runtime" dropdown set to "AutoResolveIntegrationRuntime", and a "Table" input field containing the expression "@dataset().SchemaName" and "@dataset().TableName". A red box highlights this "Table" input field. There is also a checked checkbox labeled "Enter manually".

- Connect the source dataset to DS_SQL_Watermark (Watermark table created in SQL database)
- Enter the values of fields Schemaname and Tablename as dbo and Watermark respectively.

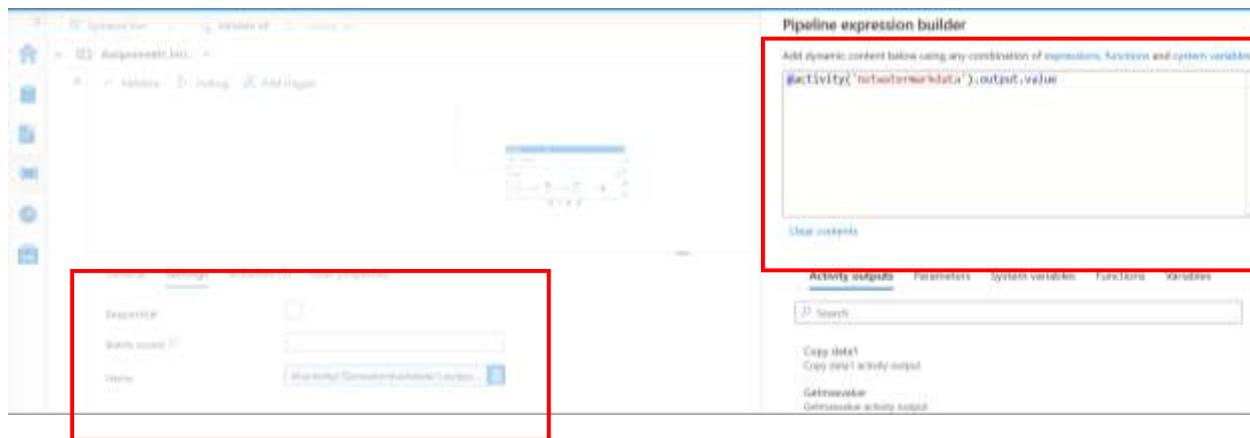
The screenshot shows the Microsoft Azure Synapse Analytics studio interface. At the top, the navigation bar includes 'Microsoft Azure', 'Synapse Analytics', and 'synapsesedhrij'. Below the navigation bar, there are tabs for 'Synapse live', 'Validate all', and 'Publish all'. The main workspace shows a pipeline named 'Assignment03_Incr...'. The pipeline has a single step, a 'Lookup' activity, which is currently selected. The 'Lookup' activity is connected to a 'Getwatermarkdata' dataset. The 'Settings' tab is selected for the 'Lookup' activity. In the 'Source dataset' section, the dropdown shows 'DS_SQL_Watermark'. A red box highlights the 'Dataset properties' section, which contains two rows: 'Schemaname' with value 'dbo' and 'Tablename' with value 'Watermark'. Other settings visible include 'First row only' (unchecked), 'Use query' (radio button selected for 'Table'), 'Query timeout (minutes)' set to 120, and 'Isolation level' set to 'Select...'. The left sidebar contains icons for Home, Datasets, Pipelines, Triggers, and Jobs.

Name	Value
Schemaname	dbo
Tablename	Watermark

- Create a Foreach activity
- Connect the lookup activity i.e. Getwatermarkdata with the Foreach activity



- Provide an expression in the items field in the Settings tab to connect with the lookup activity.
- Go to the pipeline expression builder i.e. in current scenario @activity('Getwatermarkdata').output.value



- Created a Lookup activity inside the Foreach activity for deriving the maximum value from the watermark table.
- Name it as lookup activity as Getmaxvalue

Microsoft Azure | Synapse Analytics > synapsedhrj

Search

Synapse live Validate all Publish all

Assignment03_Incr... X

Validate Debug Add trigger

Assignment03_Incremental data loading > ForEach1

Lookup

Getmaxvalue

General Settings User properties

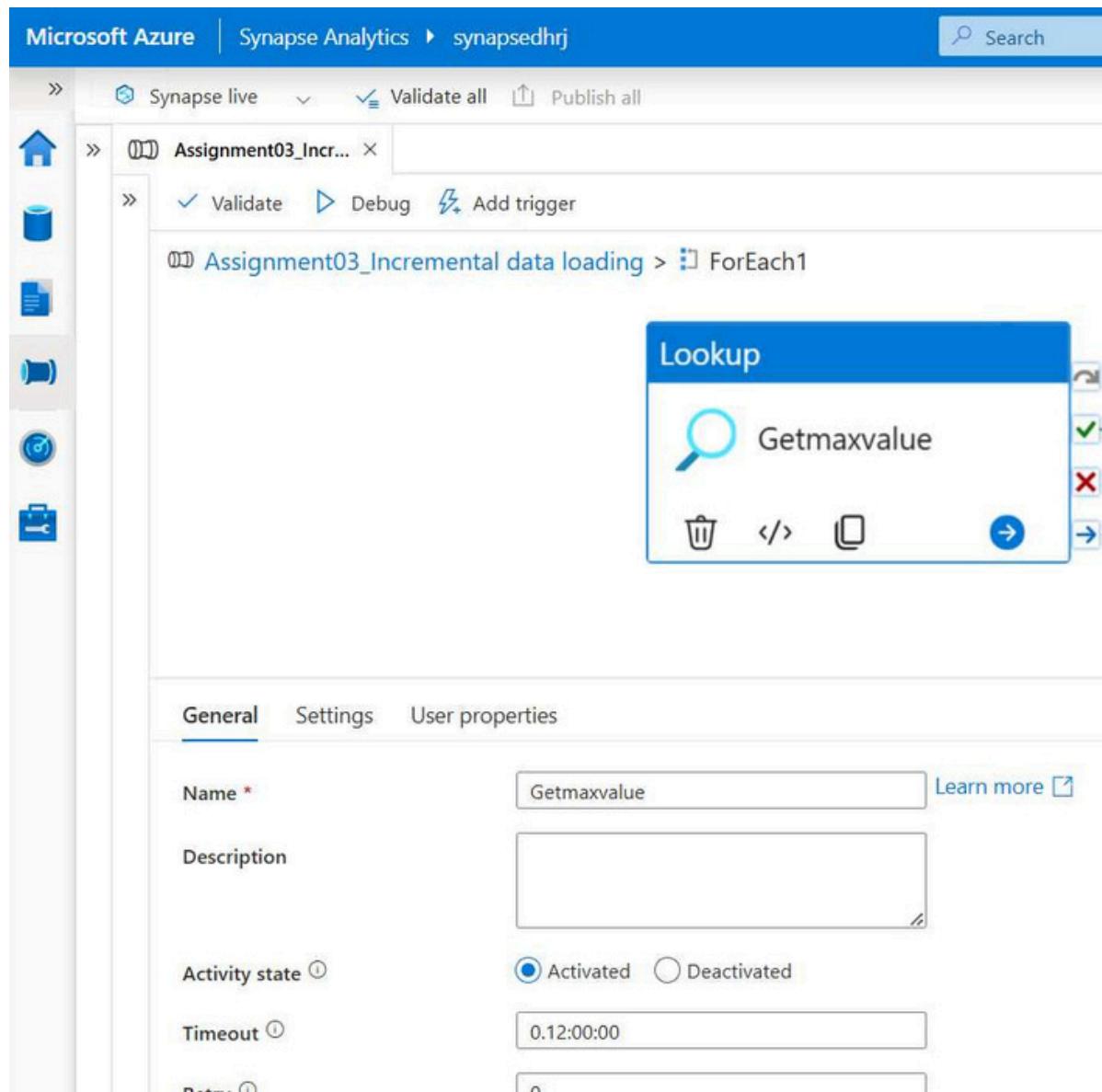
Name * Getmaxvalue Learn more ▾

Description

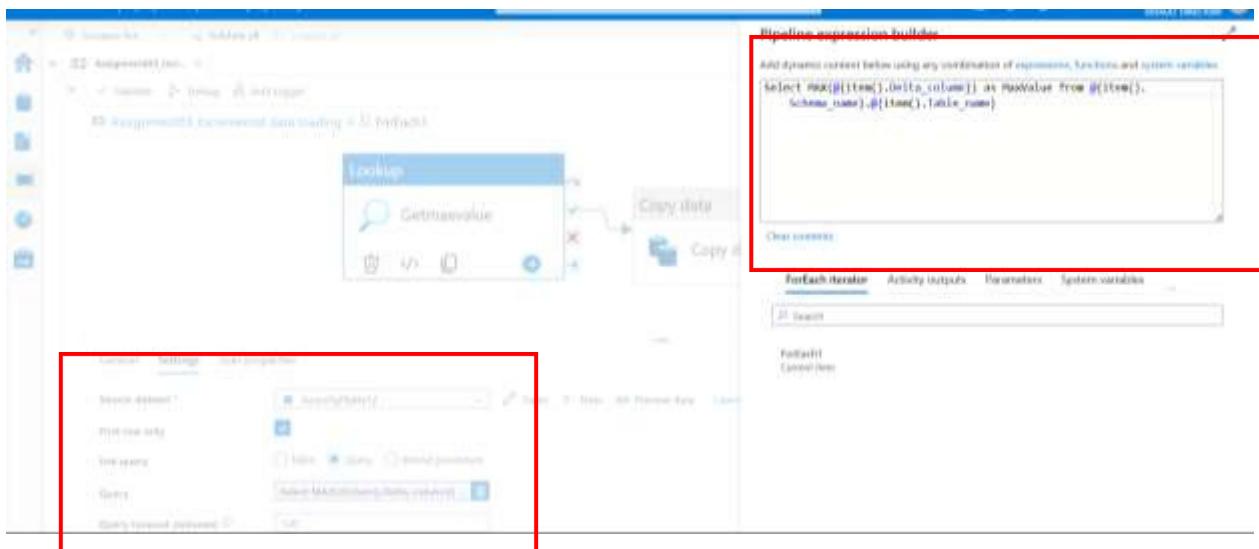
Activity state Activated Deactivated

Timeout 0.12:00:00

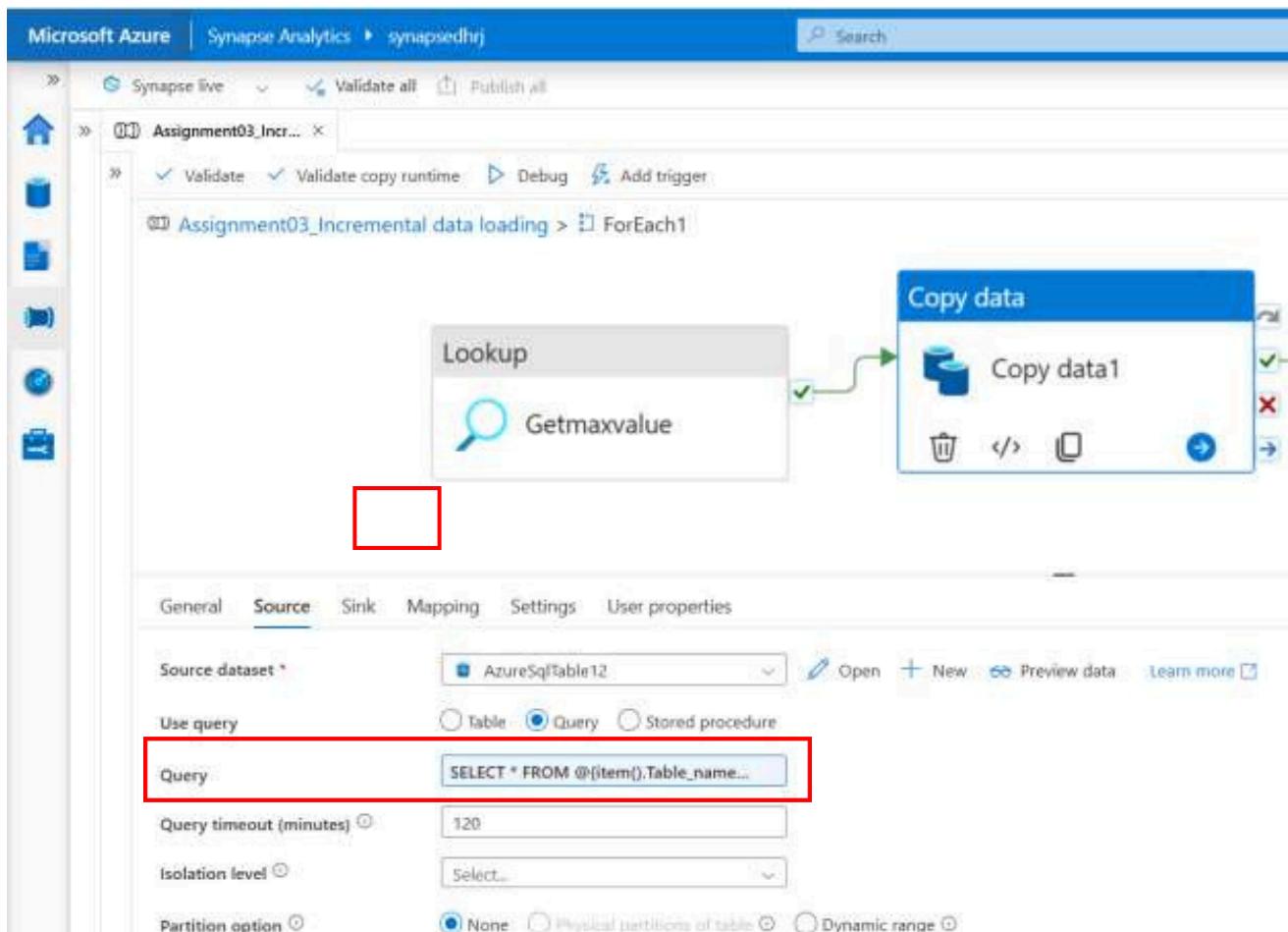
Duration

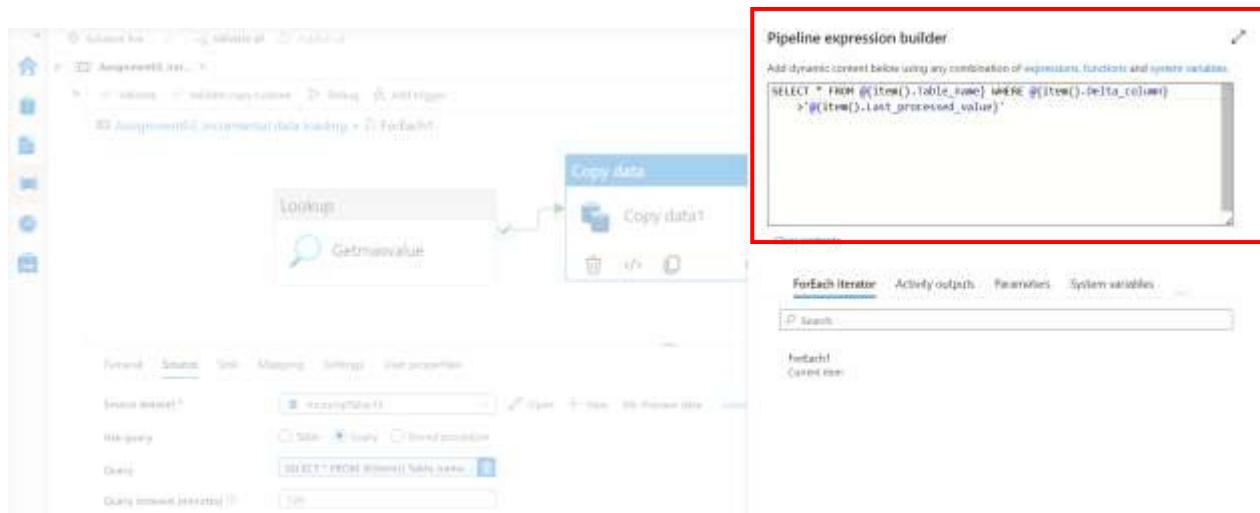


- Providing a query dynamically to get the maximum value from the watermark table
General expression (For getting the maximum value from the watermark table) SELECT MAX('Columnname') as MaxValue FROM dbo.Watermark
- Dynamic expression (Converting dynamically the above expression) SECLECT MAX(@{item().Delta_Column}) as MaxValue FROM @({item().Schema_name}).@({item().Table_name})

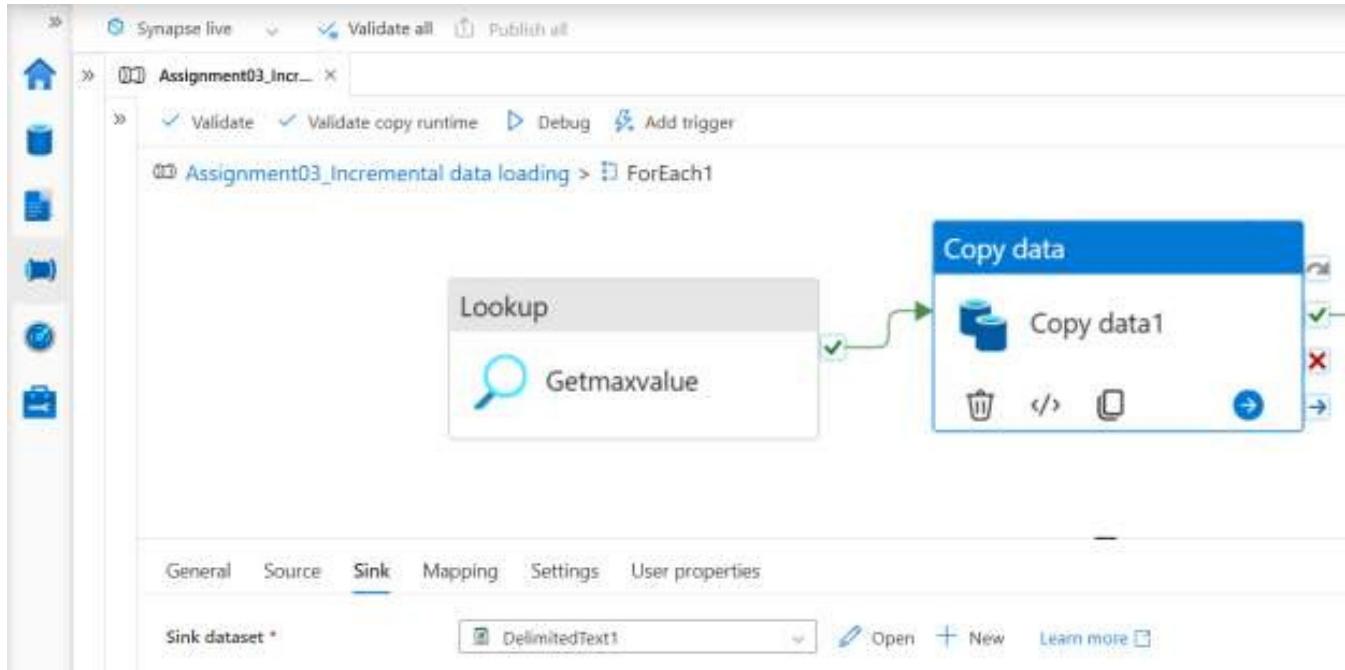


- Adding a Copy data activity
- Connect with the lookup activity i.e. Getmaxvalue activity.
- Select Source and write a dynamic query to copy only the modified values from the watermark table





- Select Sink and create a dataset name DelimitedText1 to store the output into a Azurelakegen2 or Blob storage .



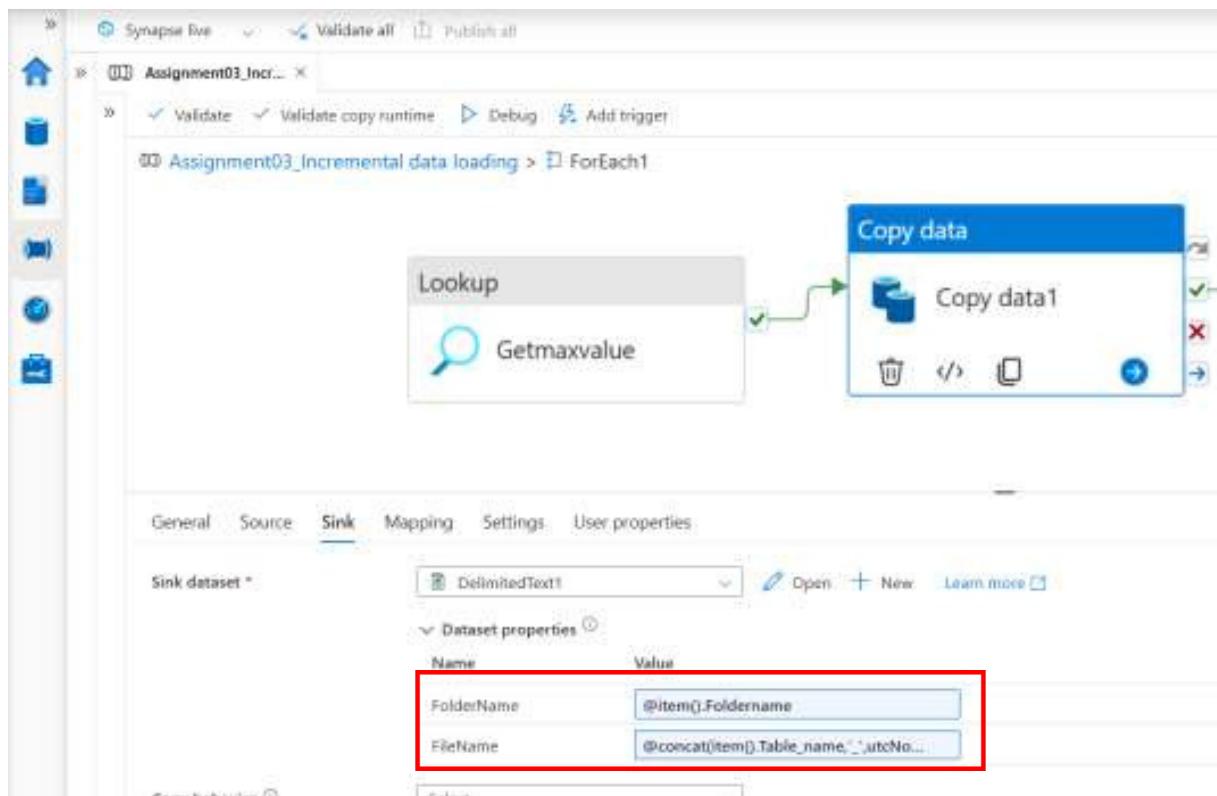
- Create 2 parameters named FolderName and FileName to store the folder name and file names respectively

The screenshot shows the Azure Data Factory studio interface. In the center, there's a preview pane with a CSV icon and the text "DelimitedText" and "DelimitedText1". Below this is a large red rectangular box highlighting the "Parameters" tab. The "Parameters" tab has three columns: "Name", "Type", and "Default value". It contains two entries: "FolderName" of type "String" with a "Value" placeholder, and "FileName" of type "String" with a "Value" placeholder. There are also "New" and "Delete" buttons at the top of this section.

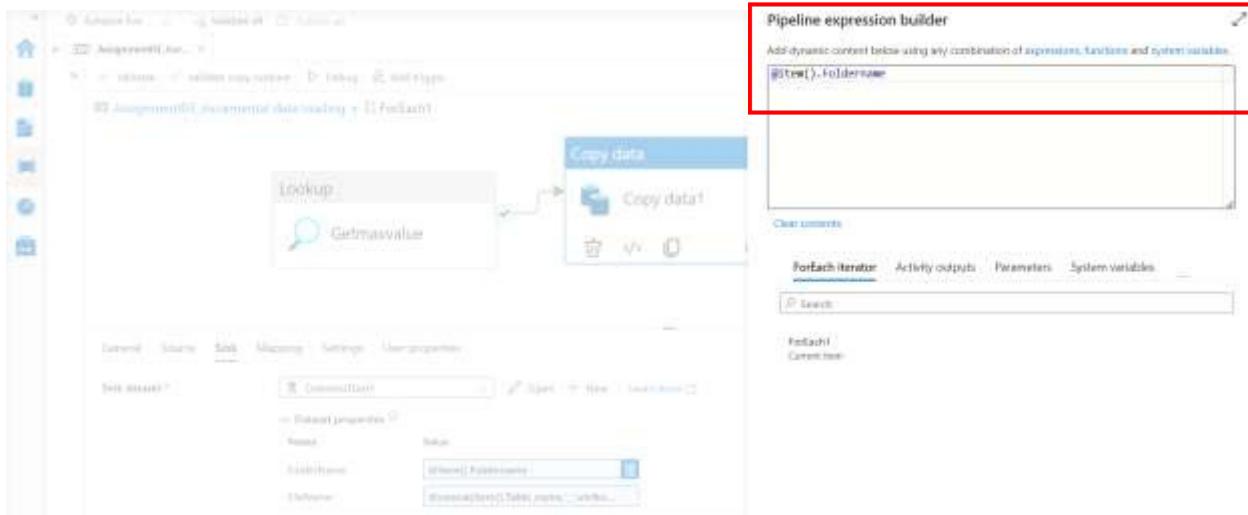
- Assign the parameters dynamically to the folder name and file name in the File path.

The screenshot shows the Azure Data Factory studio interface. In the center, there's a preview pane with a CSV icon and the text "DelimitedText" and "DelimitedText1". Below this is a large red rectangular box highlighting the "File path" field in the "Connection" tab. The "File path" field contains the expression "@dataset().FolderName + '/' + @dataset().FileName". Other settings visible include "Integration runtime", "Compression type" (No compressor), "Column delimiter" (Comma (,), "Row delimiter" (Default (CRLF) or \r\n), "Encoding" (Default(UTF-8)), and "Quote character" (Double quote ('')).

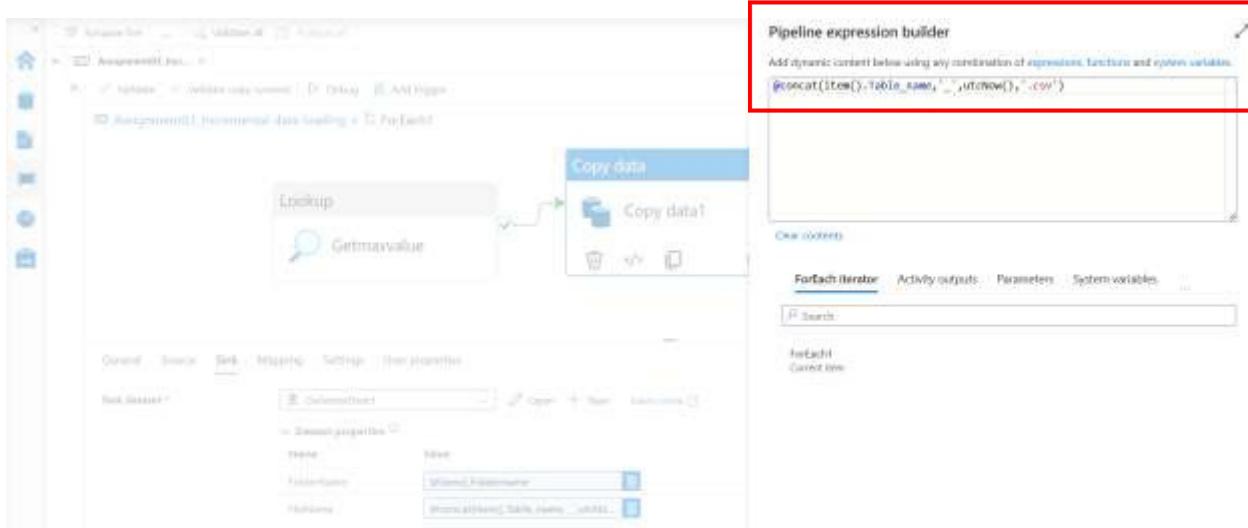
- Provide expressions as dynamic queries in the Foldername field and Filename fields.



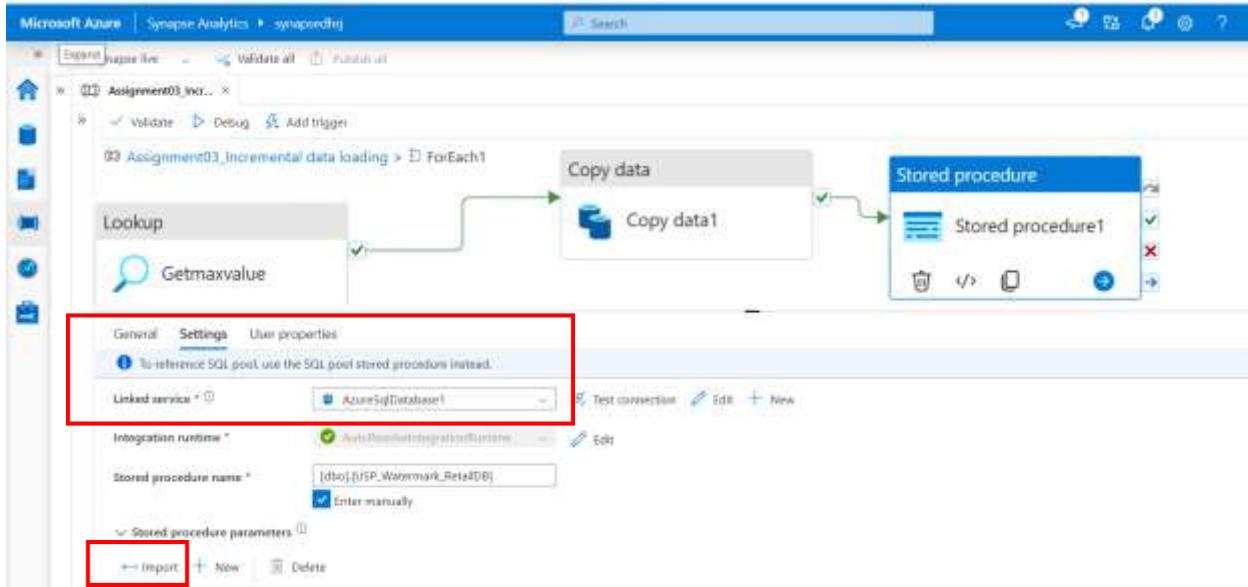
- Expression for the Foldername field would be `@(item().Foldername)`
- This expression will output the values with corresponding foldernames.



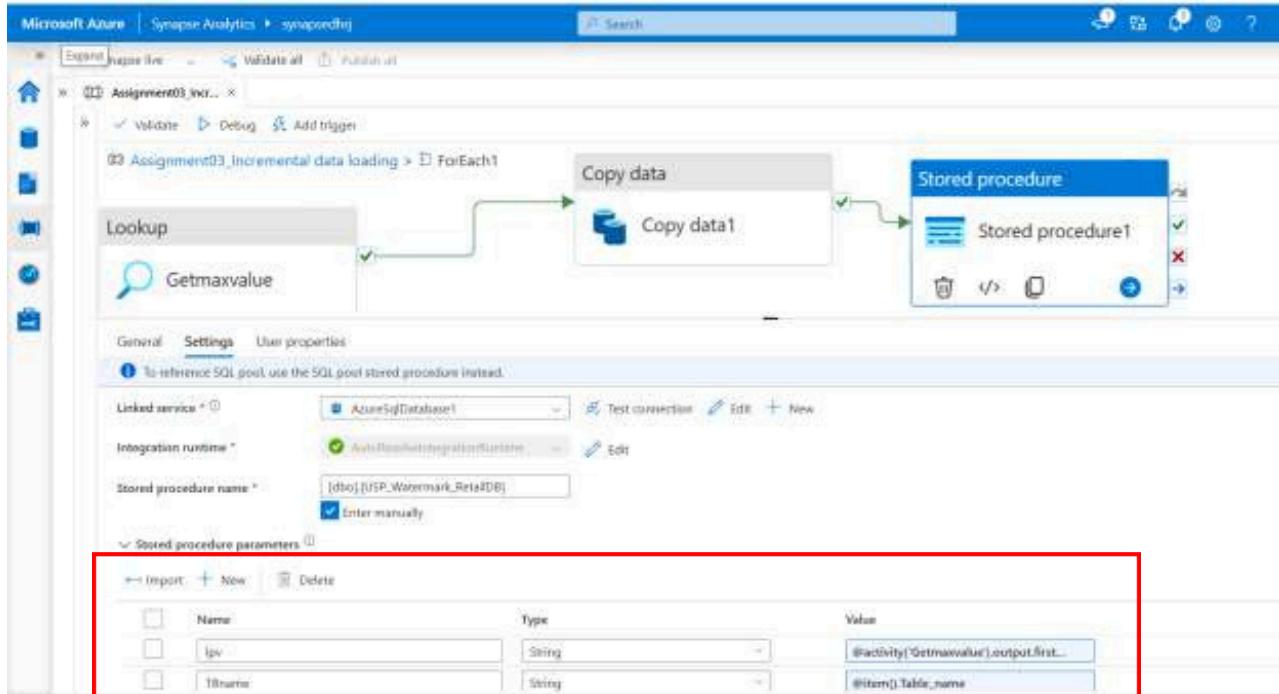
- Expression for the Foldername field would be `@(item().Foldername)` and `@concat(item().Tablename, '_', utcNow(), '.csv')`
- This expression will output the values as a .csv file format with current timestamp.



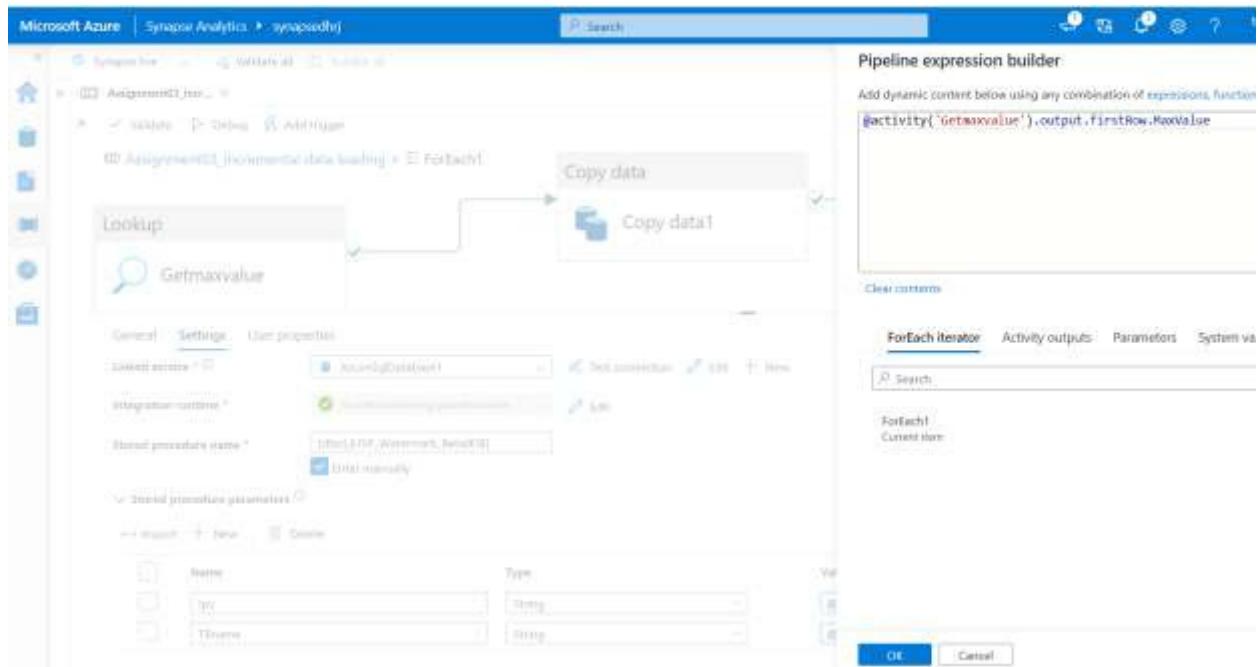
- Next Step - Adding a Stored procedure activity.
- Linking the source to the SQL database.



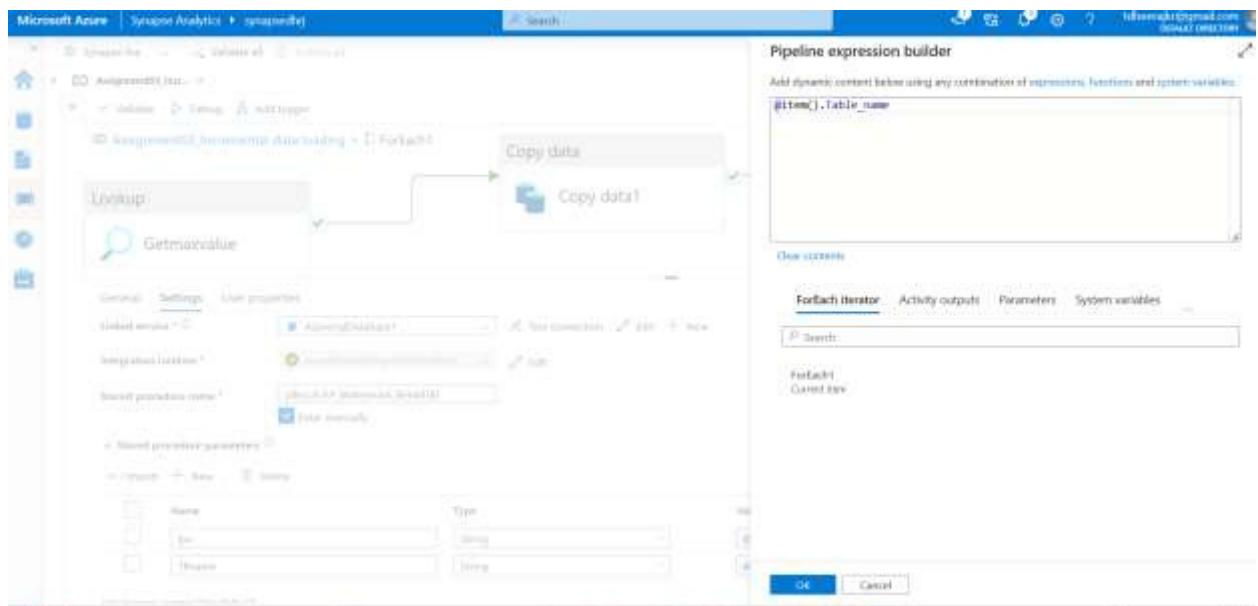
- Next click on Import parameters to import all the stored procedures created .



- Providing the dynamic expression for the lpv
@activity('Getmaxvalue').output.firstRow.Maxvalue
- This will only input the modified values from the dataset.



- Providing the dynamic expression for the TBname .@item().Table_name.
- This will only input the modified values from the respective Table names.



- Publishing and checking the pipeline if its working

Screenshot of the Microsoft Azure Synapse Analytics pipeline run history page.

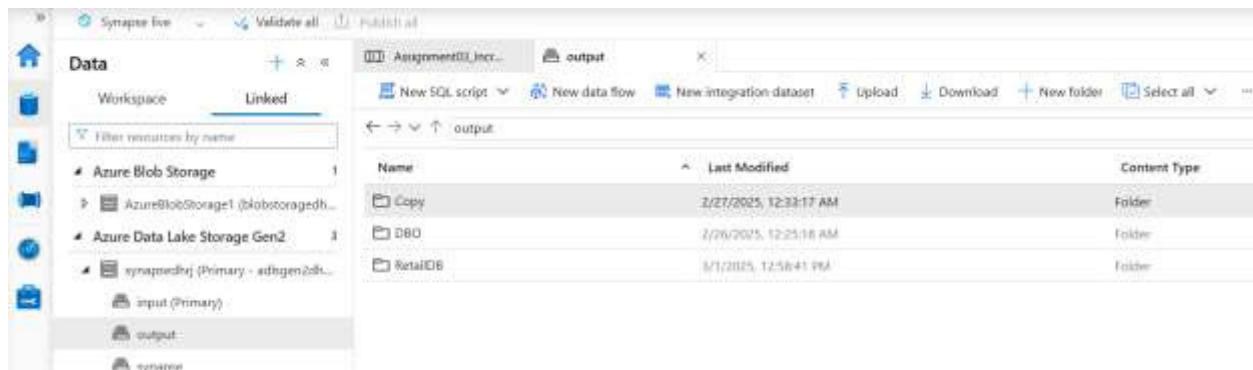
Pipeline run ID: 3b5e4966-44ac-4543-8530-3fb798ecb4b

Pipeline status: Succeeded

Activity name	Activity ID	Activity type	Run start time	Duration	Integration runtime	User properties	Activity run ID
Stored procedure1	1	Succeeded	3/1/2023, 1:09:20 PM	9s	AzSynapseIntegrationsRuntime (Australia East)	88ca019a-34b1-4ff1-82ff-c2397f33099	
Stored procedure1	2	Succeeded	3/1/2023, 1:09:19 PM	5s	AzSynapseIntegrationsRuntime (Australia East)	DB605284-5824-47ec-a43b-c4177717ea49	
Stored procedure1	3	Succeeded	3/1/2023, 1:09:18 PM	9s	AzSynapseIntegrationsRuntime (Australia East)	4c2a0fb6-a004-431d-8725-71e2a2a3476	
Stored procedure1	4	Succeeded	3/1/2023, 1:09:18 PM	9s	AzSynapseIntegrationsRuntime (Australia East)	97aa4695-7cb3-4f46-8d21-e4279124e48	
Stored procedure1	5	Succeeded	3/1/2023, 1:09:17 PM	5s	AzSynapseIntegrationsRuntime (Australia East)	ea7b162c-f114-4195-b26c-26cf41f793	
Copy data	6	Succeeded	3/1/2023, 1:09:01 PM	17s	AzSynapseIntegrationsRuntime (Australia East)	d220ce0a-7880-4522-b562-82x7r5a02a5	
Copy data	7	Succeeded	3/1/2023, 1:08:00 PM	19s	AzSynapseIntegrationsRuntime (Australia East)	8a0a729f-6a27-44b9-9427-4ab57094a3	
Copy data	8	Succeeded	3/1/2023, 1:08:00 PM	19s	AzSynapseIntegrationsRuntime (Australia East)	640512a4-6887-406b-95c5-73757d43bb0	
Copy data	9	Succeeded	3/1/2023, 1:08:00 PM	19s	AzSynapseIntegrationsRuntime (Australia East)	f993e60-4b5e-46c5-944f-93e55c7a2a3	
Copy data	10	Succeeded	3/1/2023, 1:08:00 PM	17s	AzSynapseIntegrationsRuntime (Australia East)	4704d9c1-0413-4e4e-b27a-e231fb7e1d0	
Generate output	11	Succeeded	3/1/2023, 1:08:55 PM	9s	AzSynapseIntegrationsRuntime (Australia East)	052d2943-2d53-4547-9420-1c5cde47b6d	
Generate output	12	Succeeded	3/1/2023, 1:08:55 PM	9s	AzSynapseIntegrationsRuntime (Australia East)	a4b4364a-159a-4630-6d3e-44b0fbcc07e	

Checking the output folders...

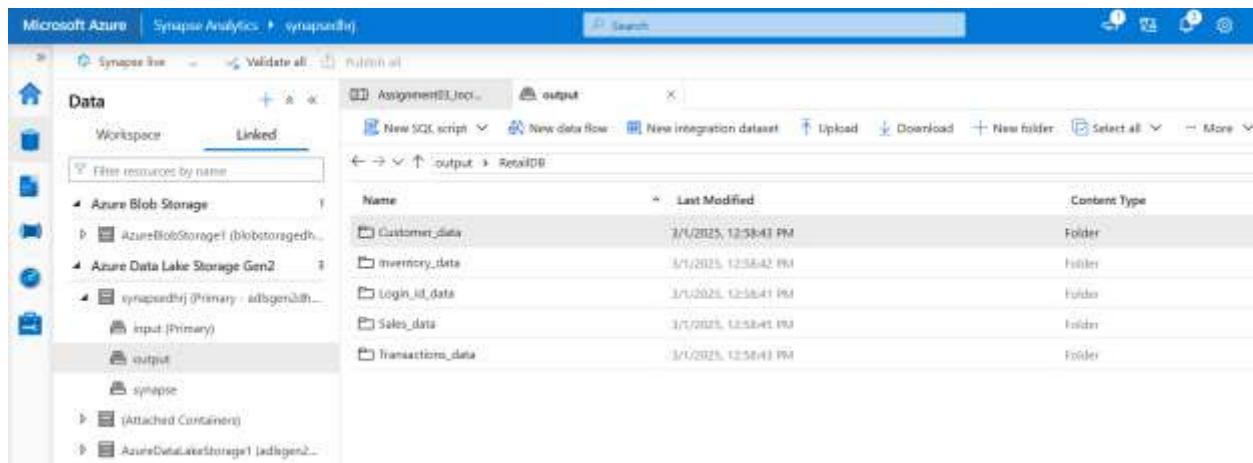
- RetailDB folder is created



The screenshot shows the Azure Synapse Analytics Data Explorer interface. On the left, the 'Data' sidebar is open, showing 'Workspace' and 'Linked' sections. Under 'Linked', there are entries for 'Azure Blob Storage' and 'Azure Data Lake Storage Gen2'. The 'output' folder under 'linked' is selected. The main pane displays a table with columns 'Name', 'Last Modified', and 'Content Type'. The table shows three items: 'Copy' (Last Modified: 2/27/2025, 12:33:17 AM), 'DBO' (Last Modified: 2/26/2025, 12:25:18 AM), and 'RetailDB' (Last Modified: 3/1/2025, 12:58:41 PM). The 'Content Type' column indicates all three are 'Folder'.

Name	Last Modified	Content Type
Copy	2/27/2025, 12:33:17 AM	Folder
DBO	2/26/2025, 12:25:18 AM	Folder
RetailDB	3/1/2025, 12:58:41 PM	Folder

- As mentioned in the Watermark table all the corresponding folders have been created i.e. Customer_data folder, Inventory_data folder, Login_id_data folder, Sales_data folder and Transactions_data folder.



This screenshot shows the Microsoft Azure Data Explorer interface, specifically the 'output' folder under 'linked'. The left sidebar shows 'Data' with 'Workspace' and 'Linked' selected. The 'output' folder under 'linked' is selected. The main pane displays a table with columns 'Name', 'Last Modified', and 'Content Type'. The table shows five items: 'Customer_data' (Last Modified: 3/1/2025, 12:58:41 PM), 'Inventory_data' (Last Modified: 3/1/2025, 12:58:42 PM), 'Login_id_data' (Last Modified: 3/1/2025, 12:58:41 PM), 'Sales_data' (Last Modified: 3/1/2025, 12:58:41 PM), and 'Transactions_data' (Last Modified: 3/1/2025, 12:58:41 PM). All items are categorized as 'Folder'.

Name	Last Modified	Content Type
Customer_data	3/1/2025, 12:58:41 PM	Folder
Inventory_data	3/1/2025, 12:58:42 PM	Folder
Login_id_data	3/1/2025, 12:58:41 PM	Folder
Sales_data	3/1/2025, 12:58:41 PM	Folder
Transactions_data	3/1/2025, 12:58:41 PM	Folder

- Cross checking the folder names in the SSMS.
- Checking in SSMS whether the name are matching with the Foldername

Assignment 03.sql - spserver-dheeraj\database\windows.net\sqlserver-dheeraj (Server (71)) - Microsoft SQL Server Management Studio

```

85 CREATE TABLE Watermark (
86     ID INT PRIMARY KEY,
87     Table_name VARCHAR(100),
88     Schema_name VARCHAR(100),
89     Foldername VARCHAR(50),
90     Delta_column VARCHAR(100),
91     Last_processed_value VARCHAR(255) NOT NULL,
92 );
93
94
95 SELECT * FROM Watermark
96
97 INSERT INTO Watermark VALUES
98 (1,'Customer','dbo','RetailDB/Customer_data','Customerupdateddate','1900-01-01 00:00:00'),
99 (2,'Login_id','dbo','RetailDB/Login_id_data','Updatedlogindata','1900-01-01 00:00:00'),
100 (3,'Inventory','dbo','RetailDB/Inventory_data','ProductID','0'),
101 (4,'Transactions','dbo','RetailDB/Transactions_data','TransactionID','0'),
102 (5,'Sales','dbo','RetailDB/Sales_data','Sales_log','1900-01-01 00:00:00')

```

100 %

Results Messages

ID	Table_name	Schema_name	Foldername	Delta_column	Last_processed_value
1	Customer	dbo	RetailDB/Customer_data	Customerupdateddate	2023-06-01T10:00:00
2	Login_id	dbo	RetailDB/Login_id_data	Updatedlogindata	2023-03-01T08:15:00
3	Inventory	dbo	RetailDB/Inventory_data	ProductID	0
4	Transactions	dbo	RetailDB/Transactions_data	TransactionID	3
5	Sales	dbo	RetailDB/Sales_data	Sales_log	2023-06-15T10:00:00

- Checking for the Incremental load.
- Adding more values into Customer, Inventory and Sales tables and publish in Synapse to check whether the incremental load is working or not ?

Assignment 03.sql -...heeraj (Server (71))* X

```

124 END
125
126
127 INSERT INTO Customer (CustomerID, Name, Phone, Customerupdateddate)
128 VALUES
129 (4, 'Alice Brown', '555-123-4567', '2023-04-01 12:00:00'),
130 (5, 'Charlie Davis', '555-789-0123', '2023-05-01 11:00:00'),
131 (6, 'Emily Wilson', NULL, '2023-06-01 10:00:00');
132
133 INSERT INTO Inventory (ProductID, ProductName, Quantity, Price)
134 VALUES
135 (4, 'Tablet', 30, 299.99),
136 (5, 'Smartwatch', 40, 199.99),
137 (6, 'Earbuds', 60, 89.99);
138
139
140 INSERT INTO Sales (SalesID, ProductID, Sales_log, Revenue)
141 VALUES
142 (4, 4, '2023-04-05 12:00:00', 299.99),
143 (5, 5, '2023-05-10 11:00:00', 199.99),
144 (6, 6, '2023-06-15 10:00:00', 89.99);
145

```

108 % ▶

	ID	Table_name	Schema_name	Foldername	Delta_column	Last_processed_value
1	1	Customer	dbo	RetailDB/Customer_data	Customerupdateddate	2023-06-01T10:00:00
2	2	Login_id	dbo	RetailDB/Login_id_data	Updatedlogindata	2023-03-01T09:15:00
3	3	Inventory	dbo	RetailDB/Inventory_data	ProductID	6
4	4	Transactions	dbo	RetailDB/Transactions_data	TransactionID	3
5	5	Sales	dbo	RetailDB/Sales_data	Sales_log	2023-06-15T10:00:00

- Verifying the output values in the corresponding file locations.
- **Inventory table**

The screenshot shows a Microsoft Excel spreadsheet with a sidebar containing navigation links such as 'File', 'Home', 'Insert', 'Page Layout', 'Formulas', 'Data', 'Review', 'View', and 'Help'. The main content area displays an inventory table with the following data:

Inventory_2025-03-01T18:09:00.457967Z.csv

Path: Home > Inventory > Inventory_2025-03-01T18:09:00.457967Z.csv

Modified: 03/01/2025, 18:09:00 UTC

With column header

Product ID	Product Name	Quantity	Price
1	Bacon	10	\$10.00
2	Ham	10	\$10.00
3	Sausage	10	\$10.00
4	Beef	10	\$10.00

- Customer table

The screenshot shows the Microsoft Power BI Data Flow interface. On the left, there's a navigation pane with various options like Home, My workspace, Recent, and others. In the center, a preview window displays a CSV file titled "Customers_2025-03-01T18:00:00Z-16174982.csv". The preview pane includes details such as the file path (<https://dflp.datalakefactory.azuredatalakestorage.net/Customers%202025-03-01T18:00:00Z-16174982.csv>) and the last modified date (2025-03-01T18:00:00Z). A "With custom header" button is shown as "On". The data preview table has columns: CUSTOMERID, NAME, PHONE, and CUSTOMERL. The data rows are:

CUSTOMERID	NAME	PHONE	CUSTOMERL
1	John Doe	00-123-456	John Doe
2	Anna Smith	00-789-012	Anna Smith
3	Mary White	00-000-000	Mary White
4	David Lee	00-999-999	David Lee

- Sales table

The screenshot shows a web-based application interface for viewing data. On the left, there is a sidebar with various icons and labels, including "Data", "File", "Import", "Export", "Search", "Help", and "Logout". Below these are sections for "Recent files", "Recent databases", "Recent queries", and "Integration libraries".

The main area displays a table titled "Sales_2025-09-01T16:28:39.8549193Z.csv". The table has columns: SALESID, PRODUCTID, SALESLOG, and REVENUE.

SALESID	PRODUCTID	SALESLOG	REVENUE
1	1	2025-09-01T16:28:39.8549193Z	100.00
2	2	2025-09-01T16:28:39.8549193Z	200.00
3	3	2025-09-01T16:28:39.8549193Z	300.00
4	4	2025-09-01T16:28:39.8549193Z	400.00