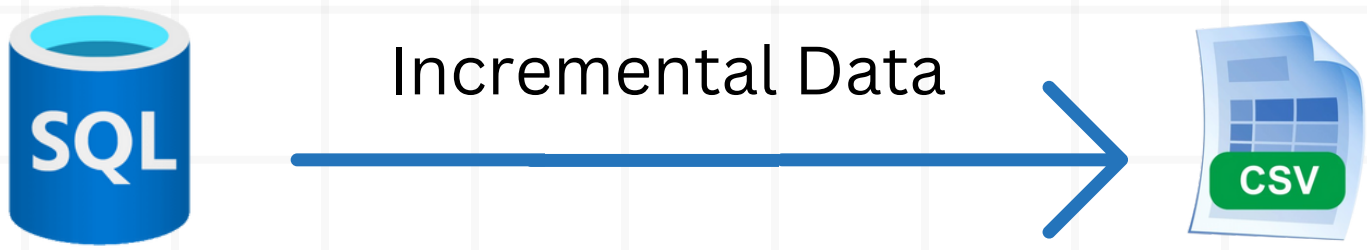


Incremental Loading in Azure Synapse Workspace

Small Project

Incremental Loading in Azure

Incremental loading refers to the process of loading only new or changed data from a data source into a destination storage or processing system.

Why do we need this concept?

It efficiently updates databases with only new or changed data, reducing load times and resource usage, thereby optimizing system performance and minimizing costs.

Want to see small example?

Yes? Let’s go!

Harpalsinh Vaghela

Incremental Loading in Azure Synapse Workspace

Create 5 tables in SSMS, two tables with int column and three tables with datetime column as delta column.

- Suppliers
- Products
- Employees
- Sales
- Inventory Logs

Run this **Create Table queries**, in SSMS**:**

```
CREATE TABLE Suppliers (
   supplier_id INT PRIMARY KEY, ----------Delta Column
   name VARCHAR(255),
   contact_name VARCHAR(255),
   phone VARCHAR(50),
   address Varchar(100)
);

CREATE TABLE Products (
   product_id INT PRIMARY KEY,  ----------Delta Column
   name VARCHAR(255),
   category VARCHAR(100),
   price DECIMAL(10, 2),
   stock INT,
   supplier_id INT,
   FOREIGN KEY (supplier_id) REFERENCES Suppliers(supplier_id)
);

CREATE TABLE Employees (
   employee_id INT PRIMARY KEY,
   first_name VARCHAR(255),
   last_name VARCHAR(255),
   hire_date DATETIME,
   last_review_date DATETIME,  ----------Delta Column
   role VARCHAR(100)
);

CREATE TABLE Sales (
   sale_id INT PRIMARY KEY,
   sale_date DATETIME,     ----------Delta Column
   product_id INT,
   quantity INT,
   total_amount DECIMAL(10, 2),
   cashier_id INT,
   FOREIGN KEY (product_id) REFERENCES Products(product_id),
   FOREIGN KEY (cashier_id) REFERENCES Employees(employee_id)
);

CREATE TABLE Inventory_Logs (
   log_id INT PRIMARY KEY,
   product_id INT,
```

```
  log_date DATETIME,   ----------------------Delta Column
  change_quantity INT,
  remaining_stock INT,
  FOREIGN KEY (product_id) REFERENCES Products(product_id)
);
```

```sql
CREATE TABLE Suppliers (
    supplier_id INT PRIMARY KEY, -------------Delta Column
    name VARCHAR(255),
    contact_name VARCHAR(255),
    phone VARCHAR(50),
    address Varchar(100)
);

CREATE TABLE Products (
    product_id INT PRIMARY KEY,  ------------Delta Column
    name VARCHAR(255),
    category VARCHAR(100),
    price DECIMAL(10, 2),
    stock INT,
    supplier_id INT,
    FOREIGN KEY (supplier_id) REFERENCES Suppliers(supplier_id)
);

CREATE TABLE Employees (
    employee_id INT PRIMARY KEY,
    first_name VARCHAR(255),
    last_name VARCHAR(255),
    hire_date DATETIME,
    last_review_date DATETIME,  ----------------Delta Column
    role VARCHAR(100)
);

CREATE TABLE Sales (
    sale_id INT PRIMARY KEY,
    sale_date DATETIME,      -------------------Delta Column
    product_id INT,
    quantity INT,
    total_amount DECIMAL(10, 2),
    cashier_id INT,
    FOREIGN KEY (product_id) REFERENCES Products(product_id),
    FOREIGN KEY (cashier_id) REFERENCES Employees(employee_id)
);

CREATE TABLE Inventory_Logs (
    log_id INT PRIMARY KEY,
    product_id INT,
    log_date DATETIME,    ---------------------Delta Column
    change_quantity INT,
    remaining_stock INT,
    FOREIGN KEY (product_id) REFERENCES Products(product_id)
```

Insert Data into those 5 tables as below:

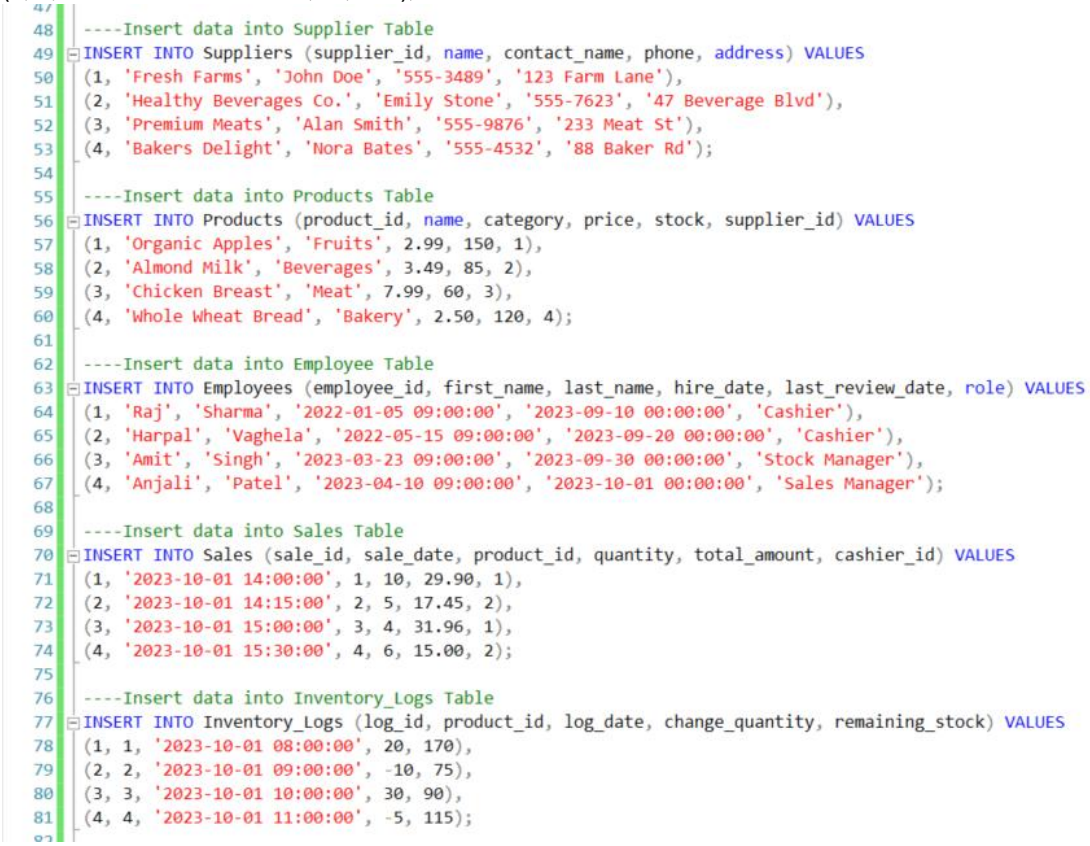
```
----Insert data into the Supplier Table
INSERT INTO Suppliers (supplier_id, name, contact_name, phone, address) VALUES
(1, 'Fresh Farms', 'John Doe', '555-3489', '123 Farm Lane'),
(2, 'Healthy Beverages Co.', 'Emily Stone', '555-7623', '47 Beverage Blvd'),
(3, 'Premium Meats', 'Alan Smith', '555-9876', '233 Meat St'),
(4, 'Bakers Delight', 'Nora Bates', '555-4532', '88 Baker Rd');
```

----Insert data into the Products Table
INSERT INTO Products (product_id, name, category, price, stock, supplier_id) VALUES
(1, 'Organic Apples', 'Fruits', 2.99, 150, 1),
(2, 'Almond Milk', 'Beverages', 3.49, 85, 2),
(3, 'Chicken Breast', 'Meat', 7.99, 60, 3),
(4, 'Whole Wheat Bread', 'Bakery', 2.50, 120, 4);

----Insert data into Employee Table
INSERT INTO Employees (employee_id, first_name, last_name, hire_date, last_review_date, role) VALUES
(1, 'Raj', 'Sharma', '2022-01-05 09:00:00', '2023-09-10 00:00:00', 'Cashier'),
(2, 'Harpal', 'Vaghela', '2022-05-15 09:00:00', '2023-09-20 00:00:00', 'Cashier'),
(3, 'Amit', 'Singh', '2023-03-23 09:00:00', '2023-09-30 00:00:00', 'Stock Manager'),
(4, 'Anjali', 'Patel', '2023-04-10 09:00:00', '2023-10-01 00:00:00', 'Sales Manager');

----Insert data into the Sales Table
INSERT INTO Sales (sale_id, sale_date, product_id, quantity, total_amount, cashier_id) VALUES
(1, '2023-10-01 14:00:00', 1, 10, 29.90, 1),
(2, '2023-10-01 14:15:00', 2, 5, 17.45, 2),
(3, '2023-10-01 15:00:00', 3, 4, 31.96, 1),
(4, '2023-10-01 15:30:00', 4, 6, 15.00, 2);

----Insert data into Inventory_Logs Table
INSERT INTO Inventory_Logs (log_id, product_id, log_date, change_quantity, remaining_stock) VALUES
(1, 1, '2023-10-01 08:00:00', 20, 170),
(2, 2, '2023-10-01 09:00:00', -10, 75),
(3, 3, '2023-10-01 10:00:00', 30, 90),
(4, 4, '2023-10-01 11:00:00', -5, 115);

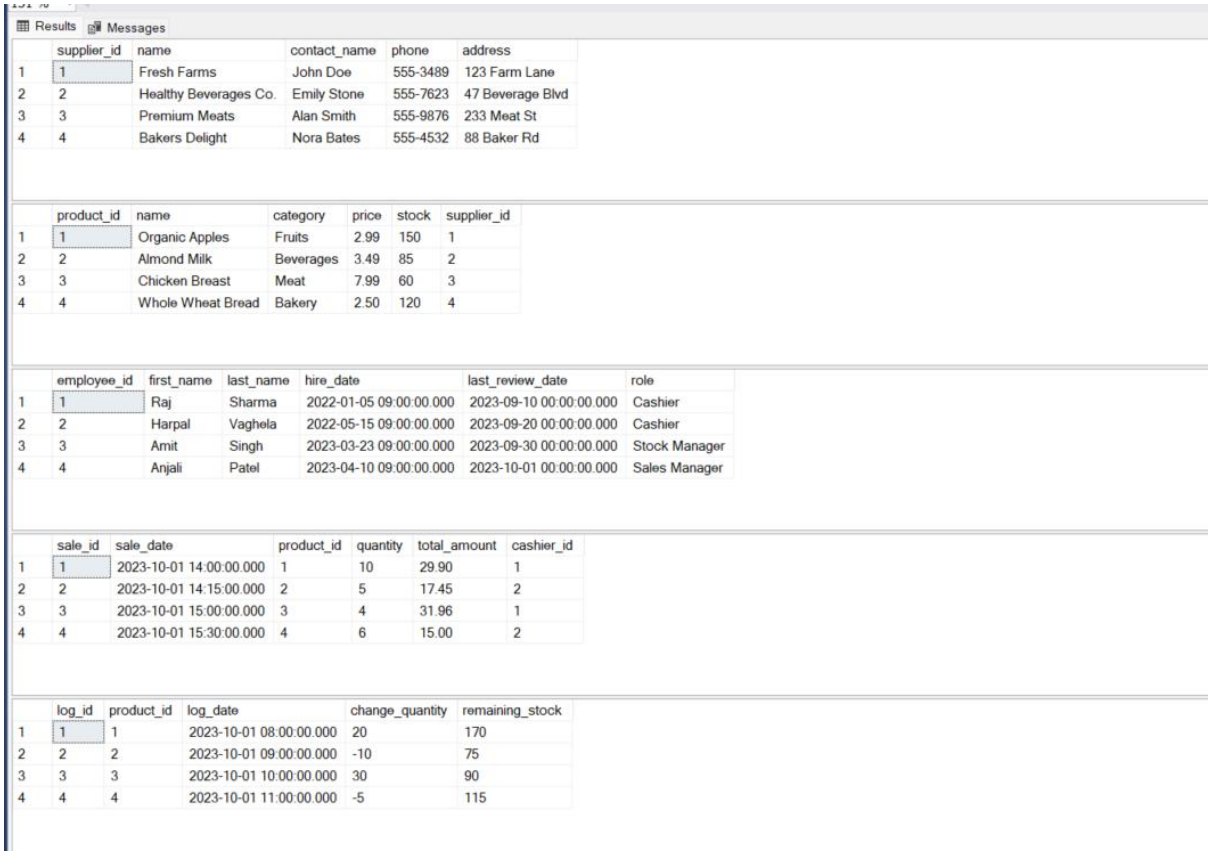


```sql
----Insert data into Supplier Table
INSERT INTO Suppliers (supplier_id, name, contact_name, phone, address) VALUES
(1, 'Fresh Farms', 'John Doe', '555-3489', '123 Farm Lane'),
(2, 'Healthy Beverages Co.', 'Emily Stone', '555-7623', '47 Beverage Blvd'),
(3, 'Premium Meats', 'Alan Smith', '555-9876', '233 Meat St'),
(4, 'Bakers Delight', 'Nora Bates', '555-4532', '88 Baker Rd');

----Insert data into Products Table
INSERT INTO Products (product_id, name, category, price, stock, supplier_id) VALUES
(1, 'Organic Apples', 'Fruits', 2.99, 150, 1),
(2, 'Almond Milk', 'Beverages', 3.49, 85, 2),
(3, 'Chicken Breast', 'Meat', 7.99, 60, 3),
(4, 'Whole Wheat Bread', 'Bakery', 2.50, 120, 4);

----Insert data into Employee Table
INSERT INTO Employees (employee_id, first_name, last_name, hire_date, last_review_date, role) VALUES
(1, 'Raj', 'Sharma', '2022-01-05 09:00:00', '2023-09-10 00:00:00', 'Cashier'),
(2, 'Harpal', 'Vaghela', '2022-05-15 09:00:00', '2023-09-20 00:00:00', 'Cashier'),
(3, 'Amit', 'Singh', '2023-03-23 09:00:00', '2023-09-30 00:00:00', 'Stock Manager'),
(4, 'Anjali', 'Patel', '2023-04-10 09:00:00', '2023-10-01 00:00:00', 'Sales Manager');

----Insert data into Sales Table
INSERT INTO Sales (sale_id, sale_date, product_id, quantity, total_amount, cashier_id) VALUES
(1, '2023-10-01 14:00:00', 1, 10, 29.90, 1),
(2, '2023-10-01 14:15:00', 2, 5, 17.45, 2),
(3, '2023-10-01 15:00:00', 3, 4, 31.96, 1),
(4, '2023-10-01 15:30:00', 4, 6, 15.00, 2);

----Insert data into Inventory_Logs Table
INSERT INTO Inventory_Logs (log_id, product_id, log_date, change_quantity, remaining_stock) VALUES
(1, 1, '2023-10-01 08:00:00', 20, 170),
(2, 2, '2023-10-01 09:00:00', -10, 75),
(3, 3, '2023-10-01 10:00:00', 30, 90),
(4, 4, '2023-10-01 11:00:00', -5, 115);
```

Check the output of tables:

Select * From Suppliers
Select * From Products

Select * From Employees
Select * From Sales
Select * From Inventory_Logs



Results Messages

	supplier_id	name	contact_name	phone	address
1	1	Fresh Farms	John Doe	555-3489	123 Farm Lane
2	2	Healthy Beverages Co.	Emily Stone	555-7623	47 Beverage Blvd
3	3	Premium Meats	Alan Smith	555-9876	233 Meat St
4	4	Bakers Delight	Nora Bates	555-4532	88 Baker Rd

	product_id	name	category	price	stock	supplier_id
1	1	Organic Apples	Fruits	2.99	150	1
2	2	Almond Milk	Beverages	3.49	85	2
3	3	Chicken Breast	Meat	7.99	60	3
4	4	Whole Wheat Bread	Bakery	2.50	120	4

	employee_id	first_name	last_name	hire_date	last_review_date	role
1	1	Raj	Sharma	2022-01-05 09:00:00.000	2023-09-10 00:00:00.000	Cashier
2	2	Harpal	Vaghela	2022-05-15 09:00:00.000	2023-09-20 00:00:00.000	Cashier
3	3	Amit	Singh	2023-03-23 09:00:00.000	2023-09-30 00:00:00.000	Stock Manager
4	4	Anjali	Patel	2023-04-10 09:00:00.000	2023-10-01 00:00:00.000	Sales Manager

	sale_id	sale_date	product_id	quantity	total_amount	cashier_id
1	1	2023-10-01 14:00:00.000	1	10	29.90	1
2	2	2023-10-01 14:15:00.000	2	5	17.45	2
3	3	2023-10-01 15:00:00.000	3	4	31.96	1
4	4	2023-10-01 15:30:00.000	4	6	15.00	2

	log_id	product_id	log_date	change_quantity	remaining_stock
1	1	1	2023-10-01 08:00:00.000	20	170
2	2	2	2023-10-01 09:00:00.000	-10	75
3	3	3	2023-10-01 10:00:00.000	30	90
4	4	4	2023-10-01 11:00:00.000	-5	115

So we have data in those 5 tables now.

Let's create a **Watermark2 table** in SSMS. (I took watermark 2 as I already have the watermark table in the database)

```
CREATE TABLE dbo.WATERMARK2
(
ID INT IDENTITY(1,1),
TABLE_NAME VARCHAR(100),
SCHEMA_NAME  VARCHAR(100),
FOLDER_NAME VARCHAR(100),
LPV VARCHAR(100),
DELTA_COLUMN VARCHAR(100)
)
```

```
CREATE TABLE dbo.WATERMARK2
(
ID INT IDENTITY(1,1),
TABLE_NAME VARCHAR(100),
SCHEMA_NAME  VARCHAR(100),
FOLDER_NAME VARCHAR(100),
LPV VARCHAR(100),
DELTA_COLUMN VARCHAR(100)
)

Select * From WATERMARK2
```

Let's **insert data into the watermark2 table:**

INSERT INTO dbo.WATERMARK2 VALUES ('Products', 'dbo', 'RetailDB/Products', 0, 'product_id');
INSERT INTO dbo.WATERMARK2 VALUES ('Sales', 'dbo', 'RetailDB/Sales', '1900-01-01 00:00:00', 'sale_date');
INSERT INTO dbo.WATERMARK2 VALUES ('Employees', 'dbo', 'RetailDB/Employees', '1900-01-01 00:00:00', 'last_review_date');
INSERT INTO dbo.WATERMARK2 VALUES ('Suppliers', 'dbo', 'RetailDB/Suppliers', 0, 'supplier_id');
INSERT INTO dbo.WATERMARK2 VALUES ('Inventory_Logs', 'dbo', 'RetailDB/InventoryLogs', '1900-01-01 00:00:00', 'log_date');

Watermark2 table output:

108 %

Results | Messages

	ID	TABLE_NAME	SCHEMA_NAME	FOLDER_NAME	LPV	DELTA_COLUMN
1	1	Products	dbo	RetailDB/Products	0	product_id
2	2	Sales	dbo	RetailDB/Sales	1900-01-01 00:00:00	sale_date
3	3	Employees	dbo	RetailDB/Employees	1900-01-01 00:00:00	last_review_date
4	4	Suppliers	dbo	RetailDB/Suppliers	0	supplier_id
5	5	Inventory_Logs	dbo	RetailDB/InventoryLogs	1900-01-01 00:00:00	log_date

Create a Stored Procedure to update the value of LPV as shown below:

```
----------Stored Procedure----------


CREATE PROC USP_WATERMARK2_UPDATE
@Table_Name VARCHAR(100),
@LPV_Value VARCHAR(50)
AS
UPDATE WATERMARK2
SET LPV = @LPV_Value
WHERE TABLE_NAME = @Table_Name
```

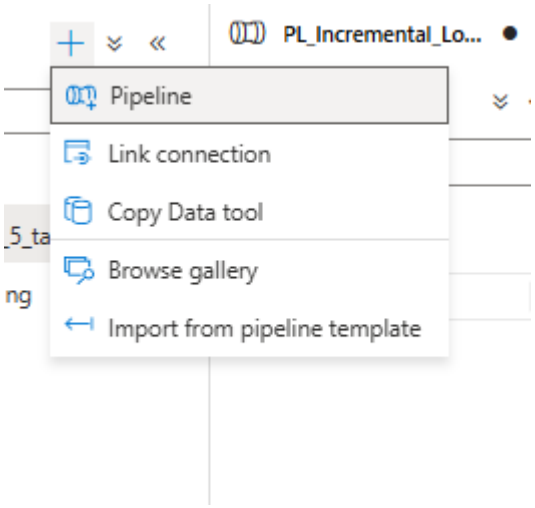
CREATE PROC USP_WATERMARK2_UPDATE

```
@Table_Name VARCHAR(100),
@LPV_Value VARCHAR(50)
AS
UPDATE WATERMARK2
SET LPV = @LPV_Value
WHERE TABLE_NAME = @Table_Name
```

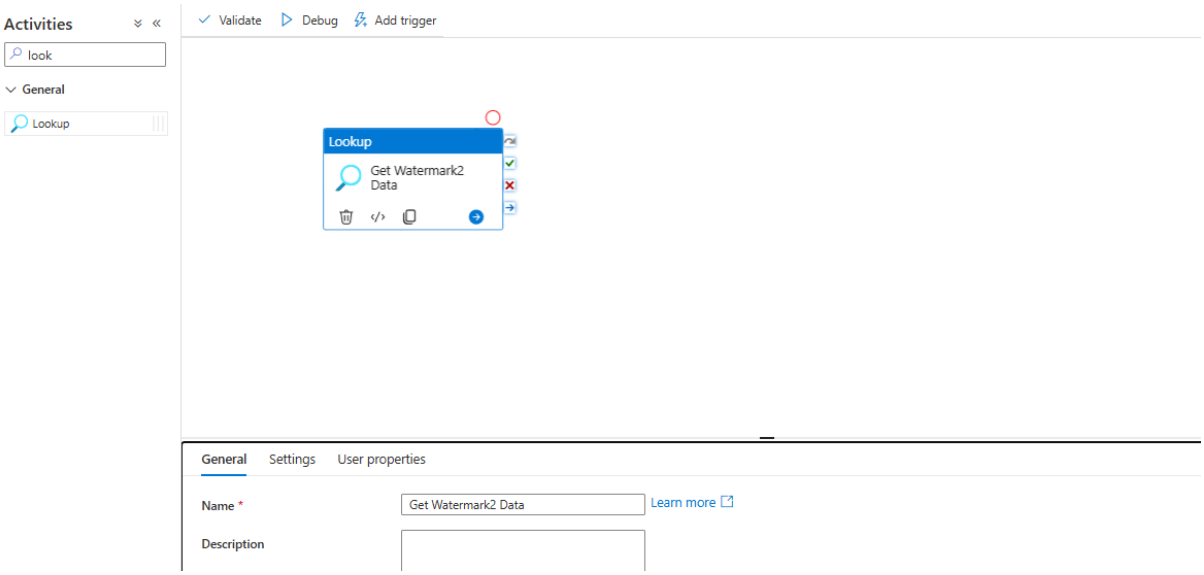
Let's Design the Pipeline

Open Azure Portal and go to **Azure Synapse Workspace.**

Integrate Tab -> Create New Pipeline

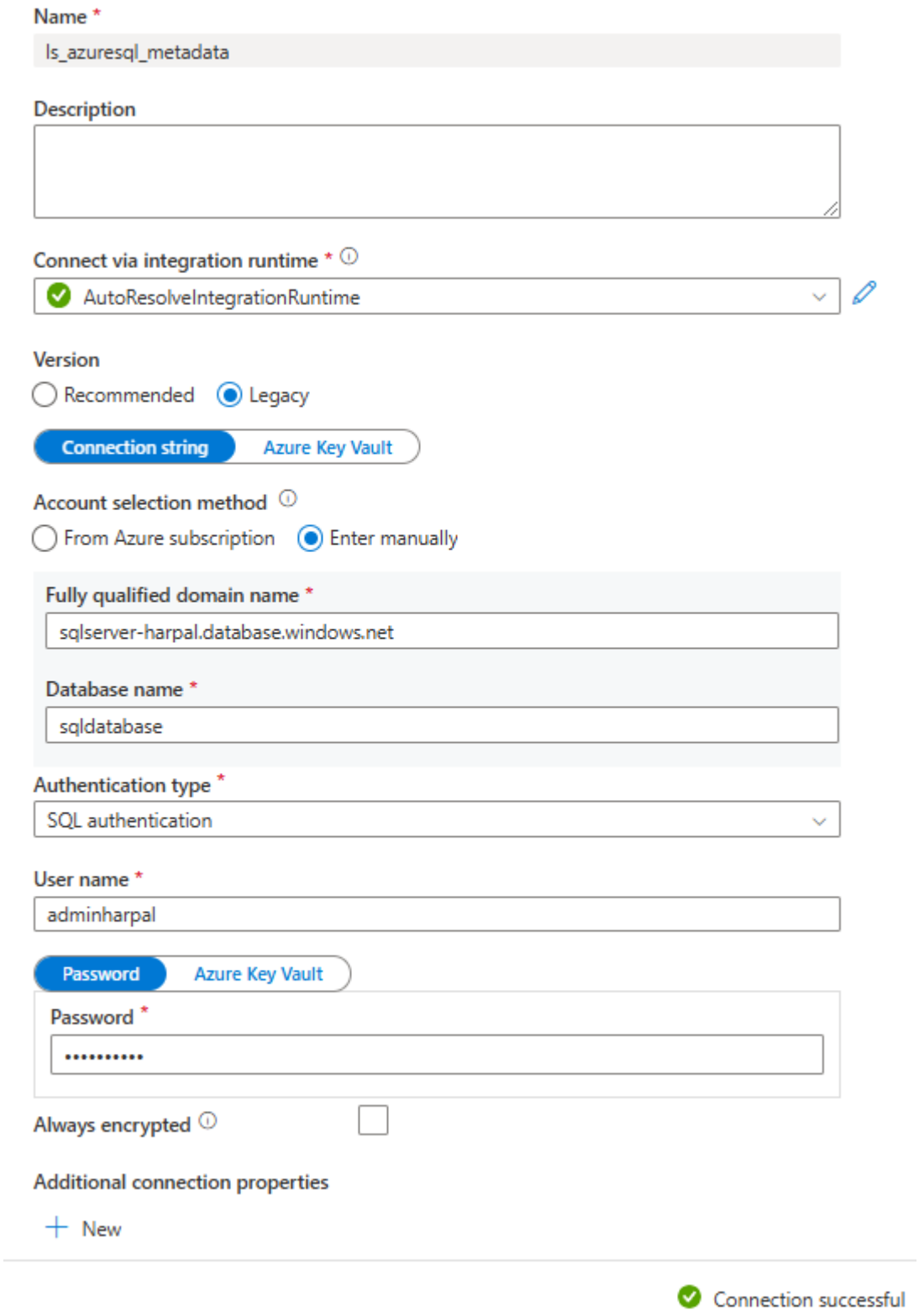


Drag and drop **Lookup Activity** and rename it **"Get Watermark2 Data"**



Go to the **Settings** tab in the lookup activity and select **the source dataset** if you already have it or select new.

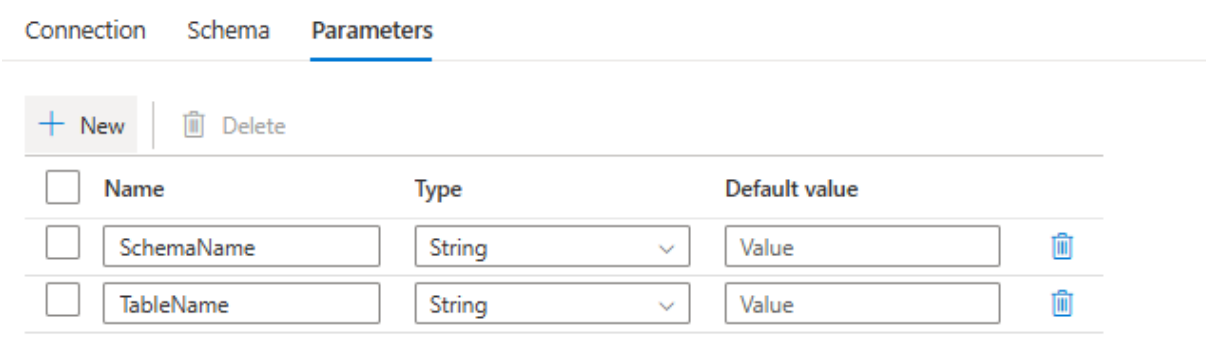
Select **Azure SQL Database** and Create **Linked Service** as shown below:



Test the Connection.

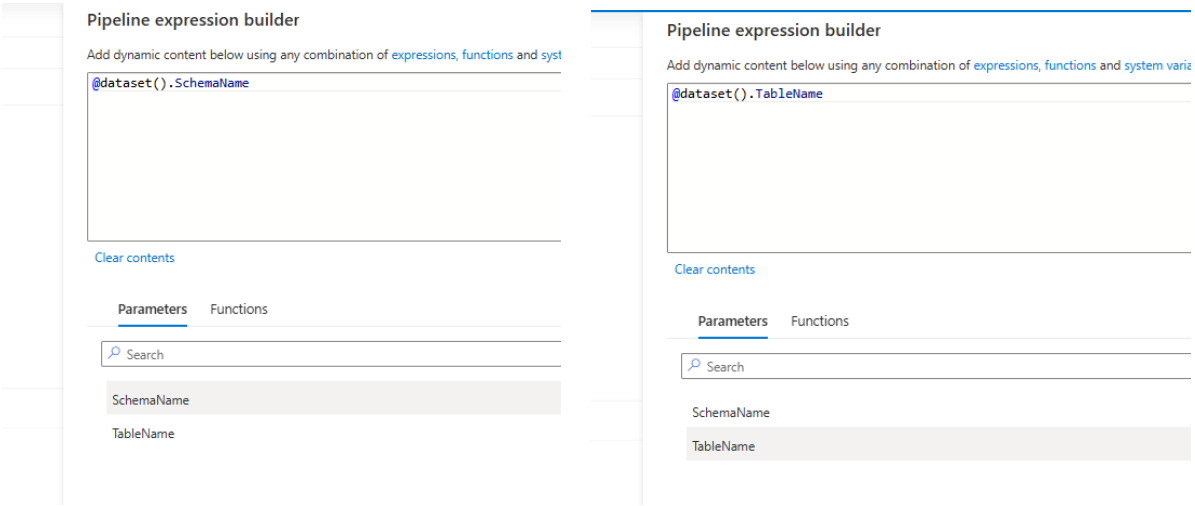
Go to the **Parameters tab** and write 2 parameters

- **SchemaName**
- **TableName**

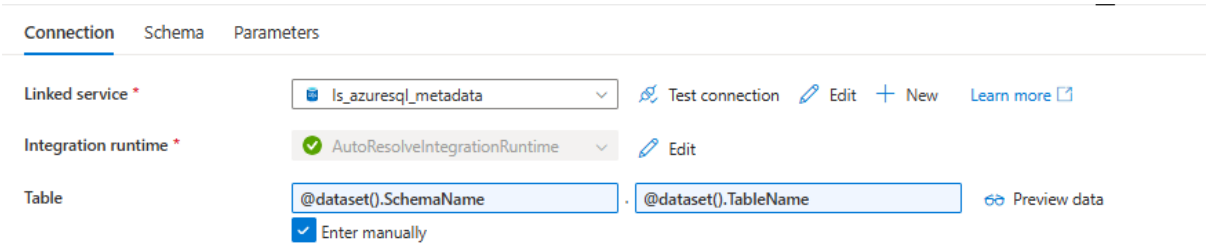


Go to the **Connection Tab** and Assign those 2 parameters

Click on **Enter Manually** and Select those parameters in Table files as shown below in the connection tab, step by step:

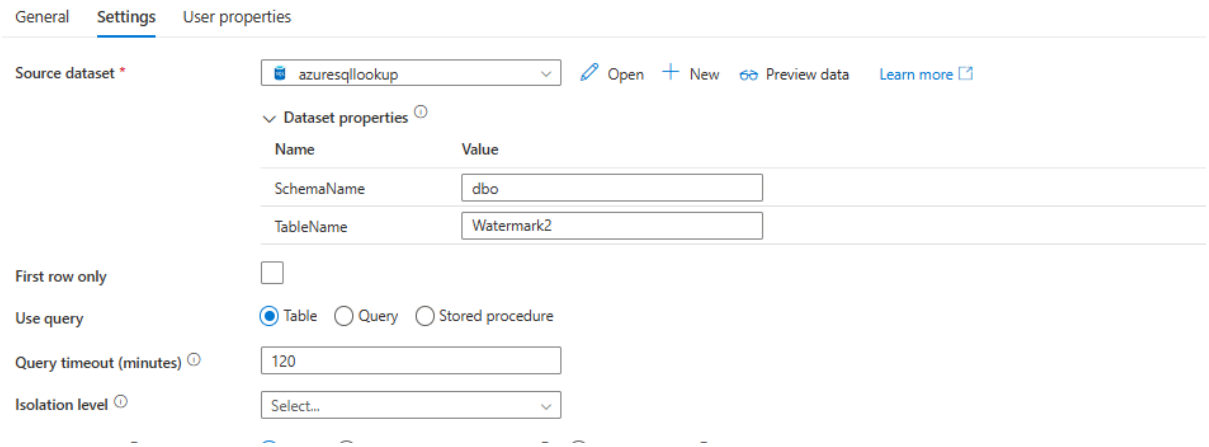


After assigning it will look like this:

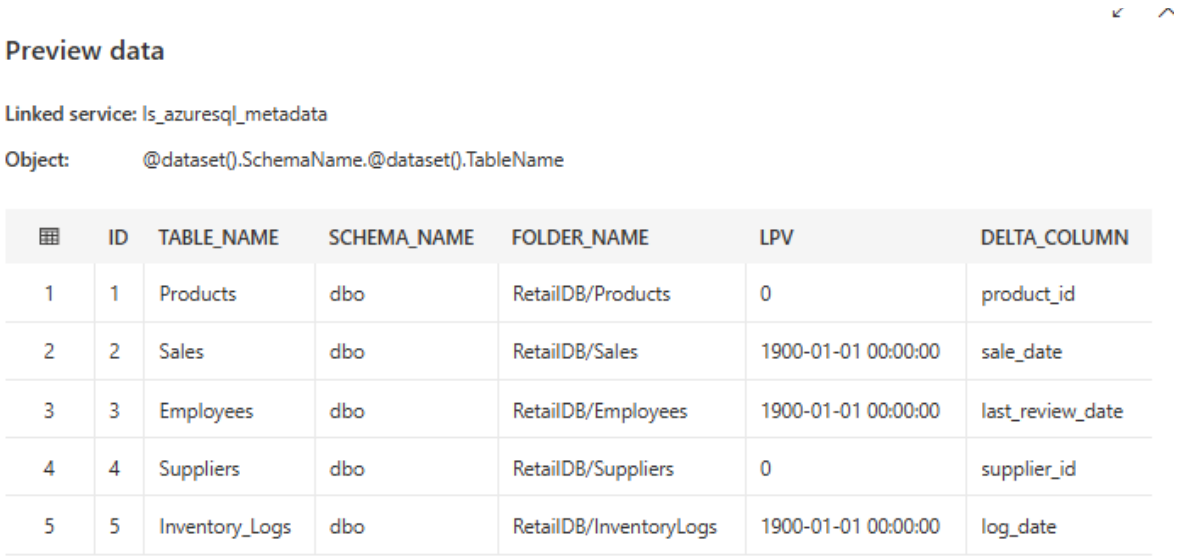


Go back to the **Settings tab** of the Lookup activity

Here, **Uncheck the First Row Only option** and Provide the Schema Name and Table name of the Watermark table which is in SSMS.



Click on Preview Data

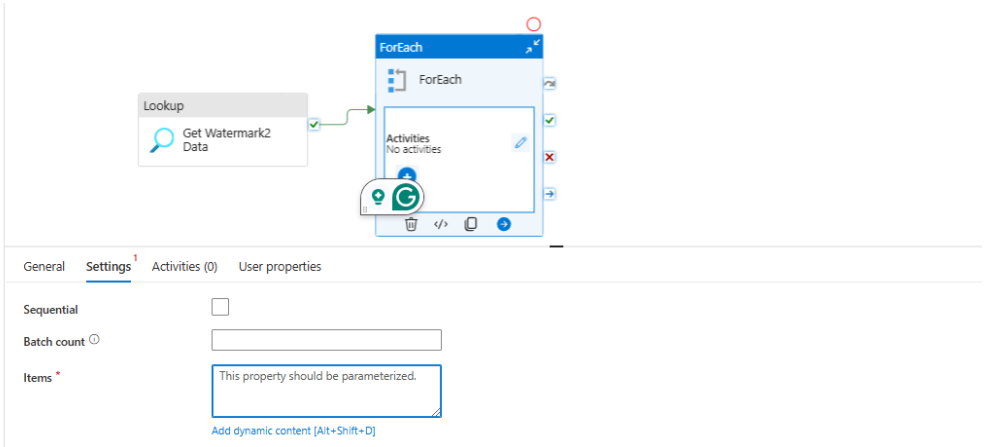


	ID	TABLE_NAME	SCHEMA_NAME	FOLDER_NAME	LPV	DELTA_COLUMN
1	1	Products	dbo	RetailDB/Products	0	product_id
2	2	Sales	dbo	RetailDB/Sales	1900-01-01 00:00:00	sale_date
3	3	Employees	dbo	RetailDB/Employees	1900-01-01 00:00:00	last_review_date
4	4	Suppliers	dbo	RetailDB/Suppliers	0	supplier_id
5	5	Inventory_Logs	dbo	RetailDB/InventoryLogs	1900-01-01 00:00:00	log_date

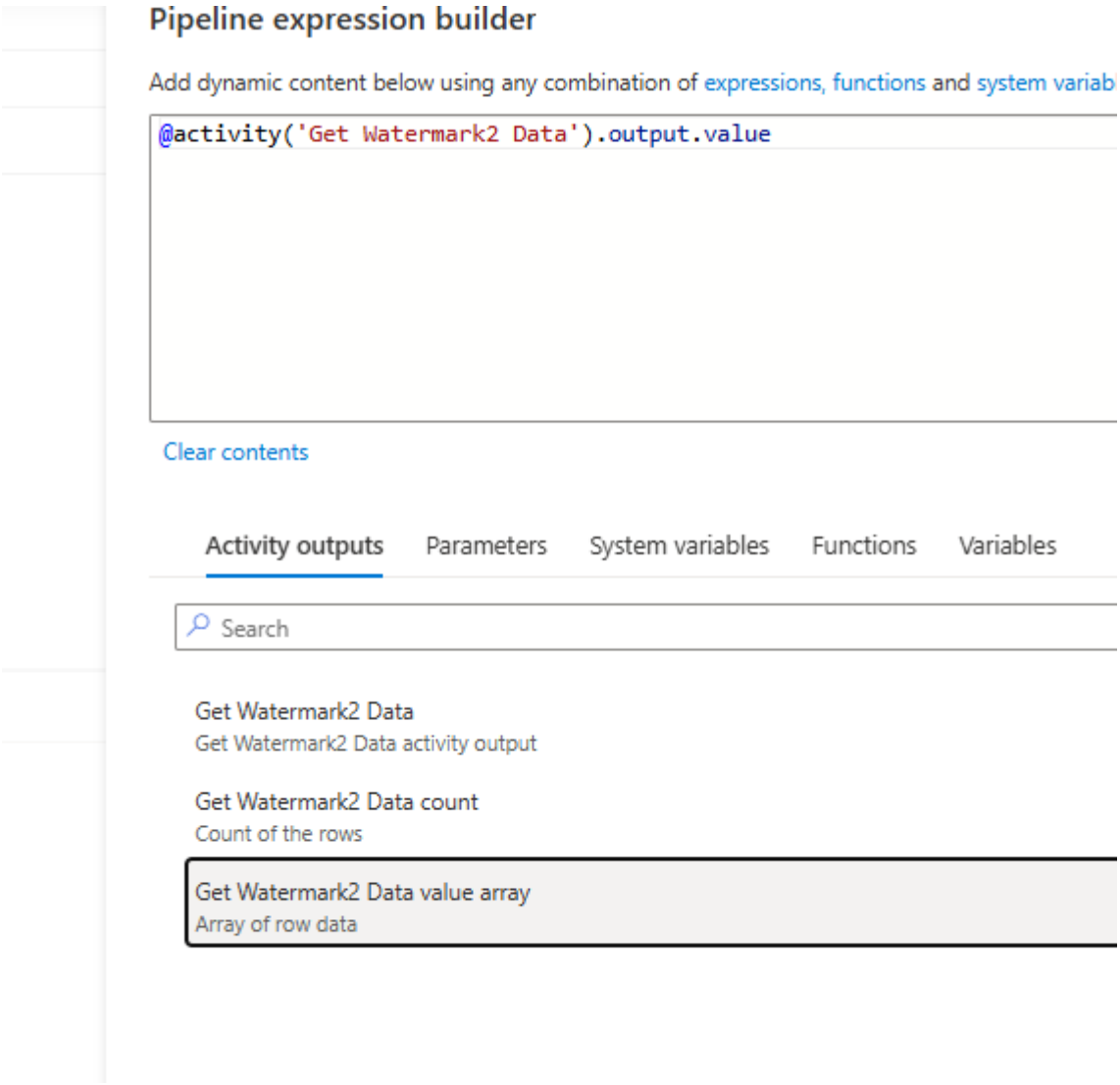
Here, we are done with this first lookup activity.

Now, Let's **ForEach Activity** in the pipeline design.

Connect Lookup with ForEach as shown below and Go to the **Settings** tab of **ForEach Activity**

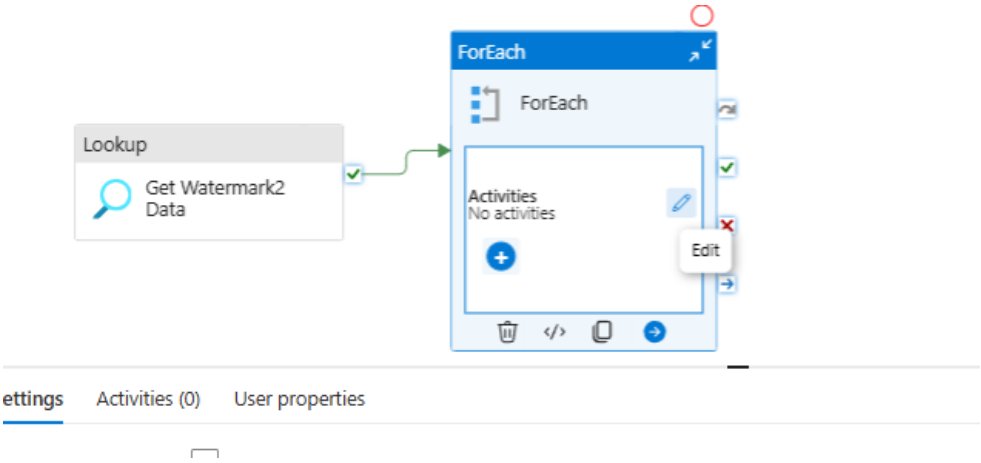


Check on **Items** and then there will be blue text as "**Add dynamic content**" Click on it and **assign the value array** as shown below:



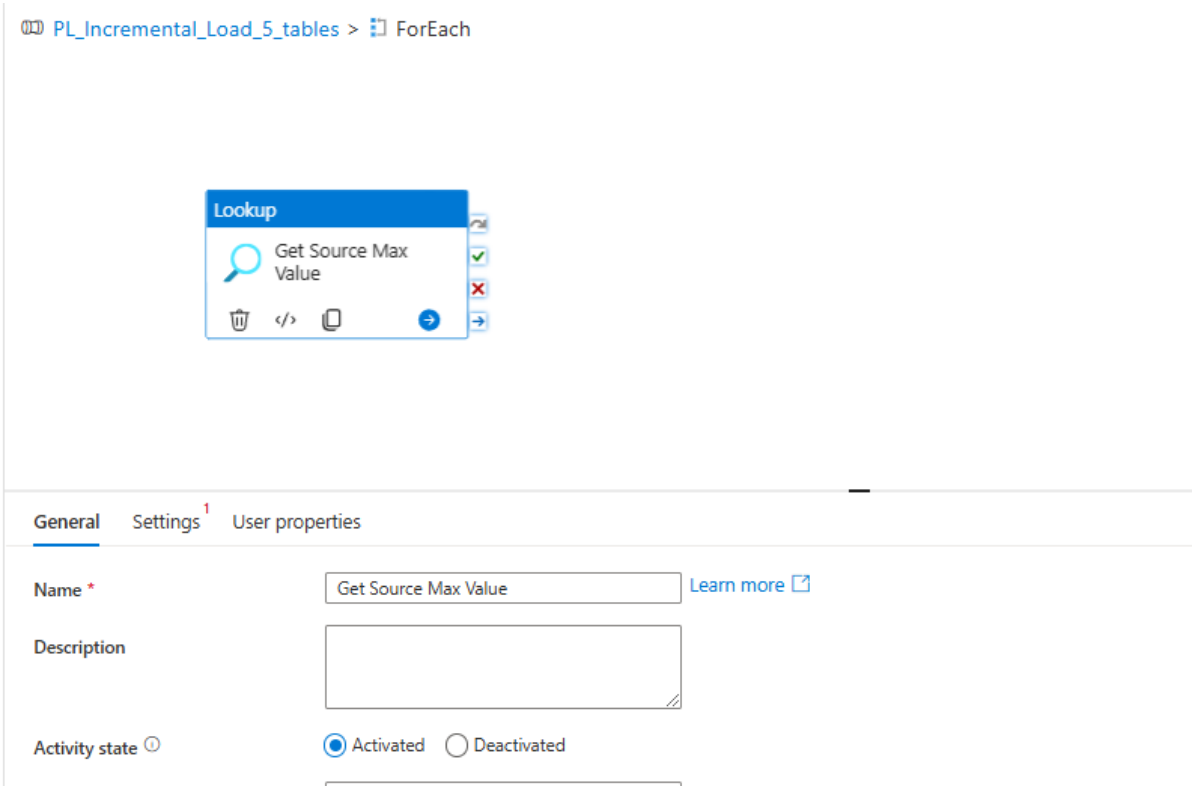
Click OK.

Now Click on the **Edit icon** of **ForEach Activity**



As we can see we are inside the ForEach Activity now.

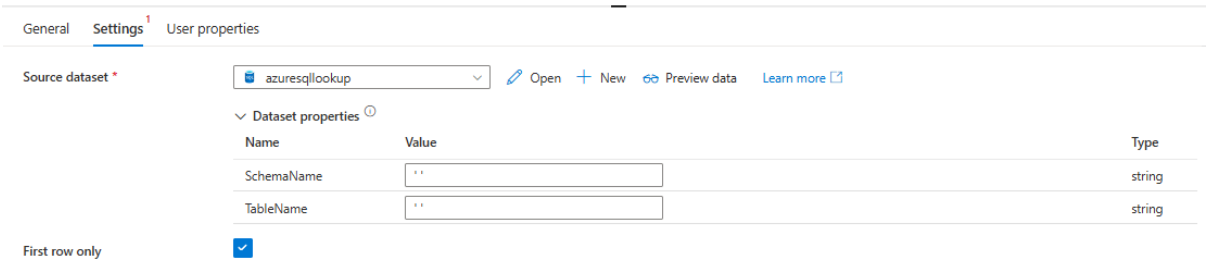
Drag and drop New **Lookup Activity** and rename it as **Get Source Max Value.**



Go to the **Settings tab** and select the source dataset which we have selected before for the first lookup activity from the dropdown list.

Enter a **single quotation** in the value fields for Schema and Table Name as indicated below. This effectively assigns a space value, signifying that no actual Schema and Table Names are needed**. Our objective is solely to obtain the maximum value for our LPV column, thus these fields can be left unspecified:**

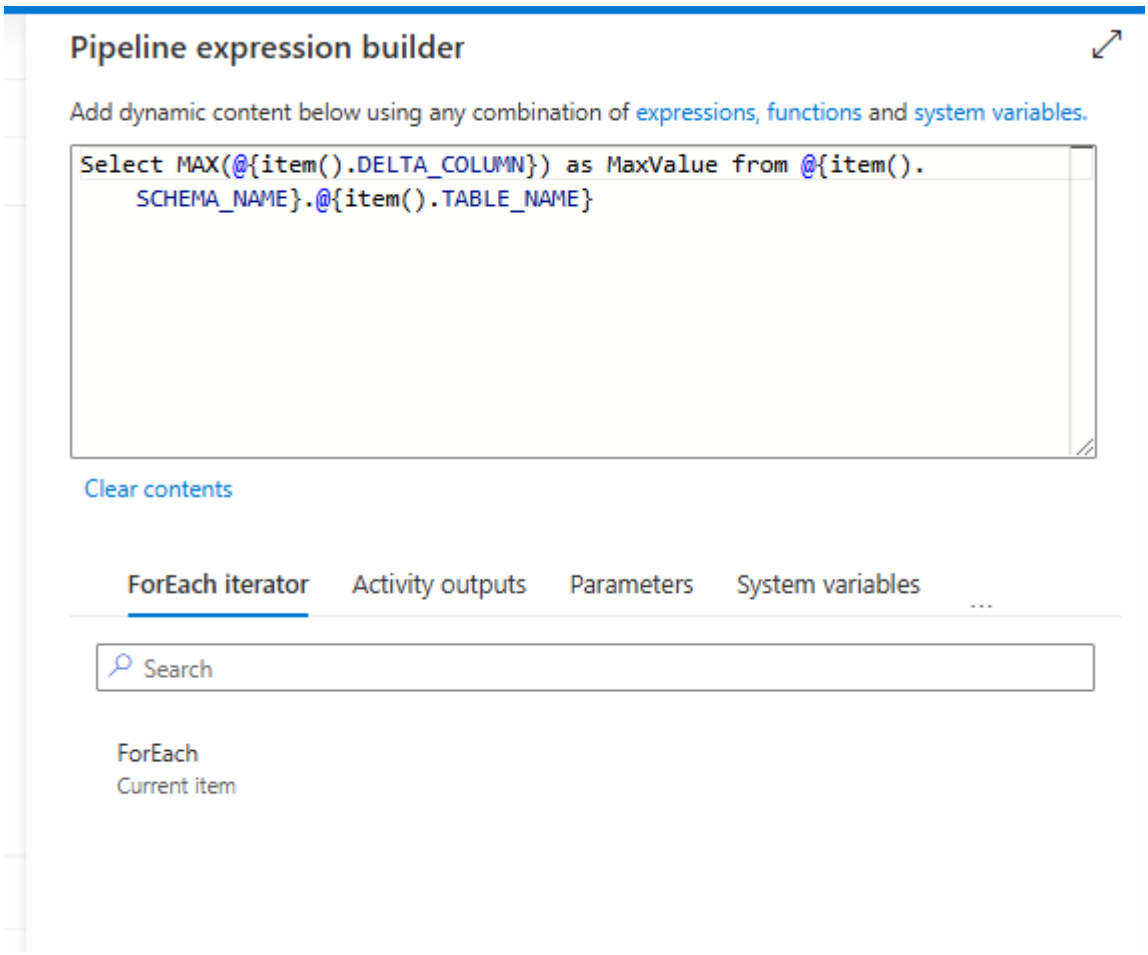
Make sure to tick mark on the **First row only** checkbox



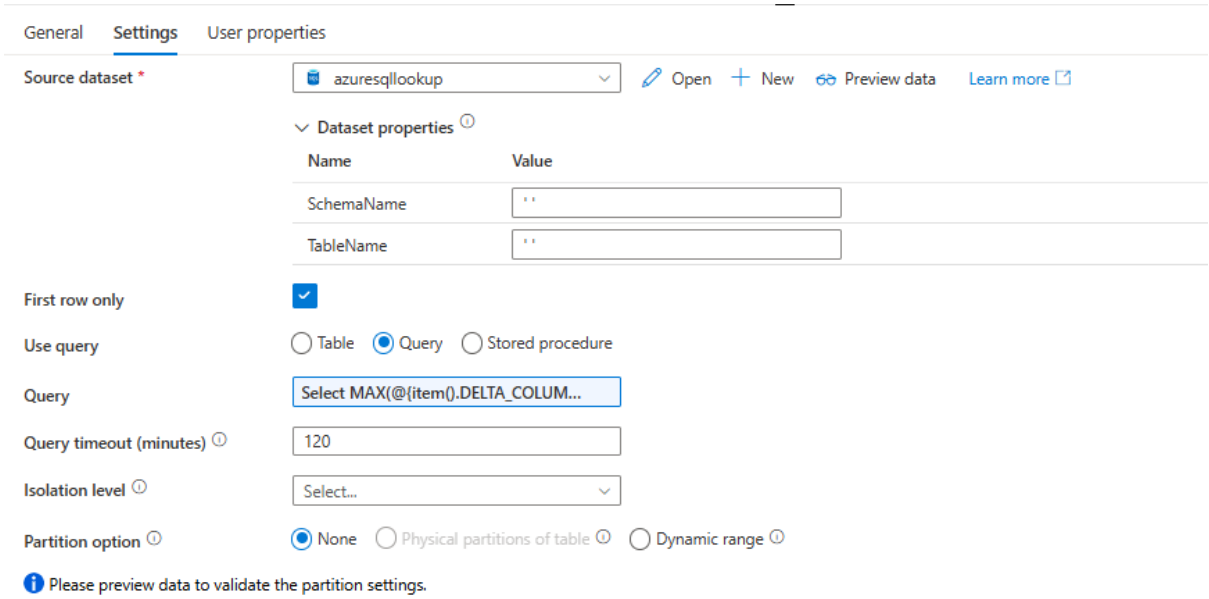
Go to Use query options and select the **Query** radio button

Click on **Add Dynamic Content** and write this query in the dialogue opened.

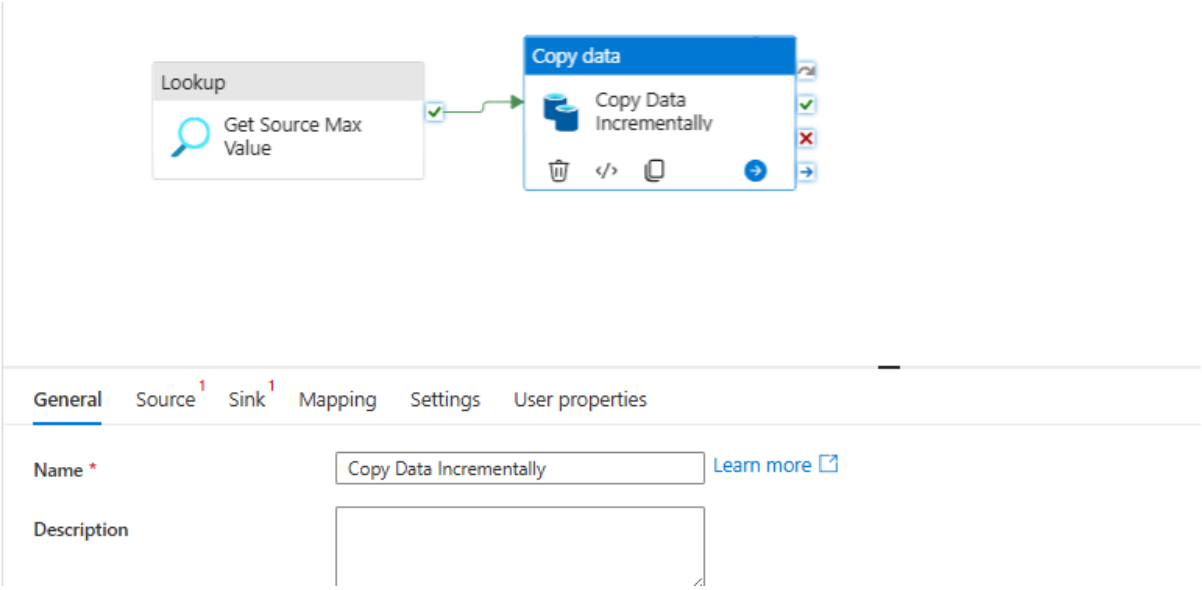
Select MAX(@{item().DELTA_COLUMN}) as MaxValue from @{item().SCHEMA_NAME}.@{item().TABLE_NAME}



After that, the settings tab of the lookup activity in ForEach Activity will look like this:



Now, drag and drop the **Copy Data Activity** in the design area and **connect it from the Lookup Activity**

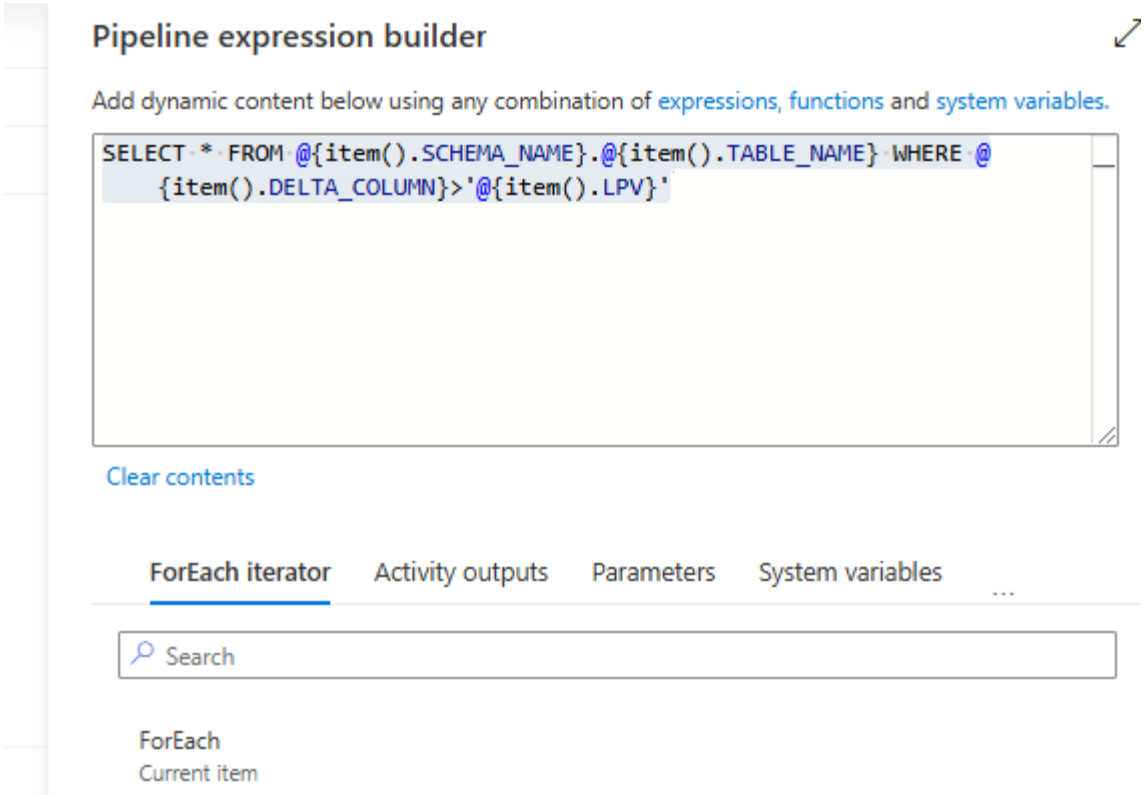


Go to the **Source Tab** and **select the source dataset** which we selected before in the lookup activity (Azure SQL Database)

- Provide a single quotation in schema name and table name (’ ‘)
- Go to Use query options and select the **Query** radio button
- Click on Add Dynamic Content and the dialog will open

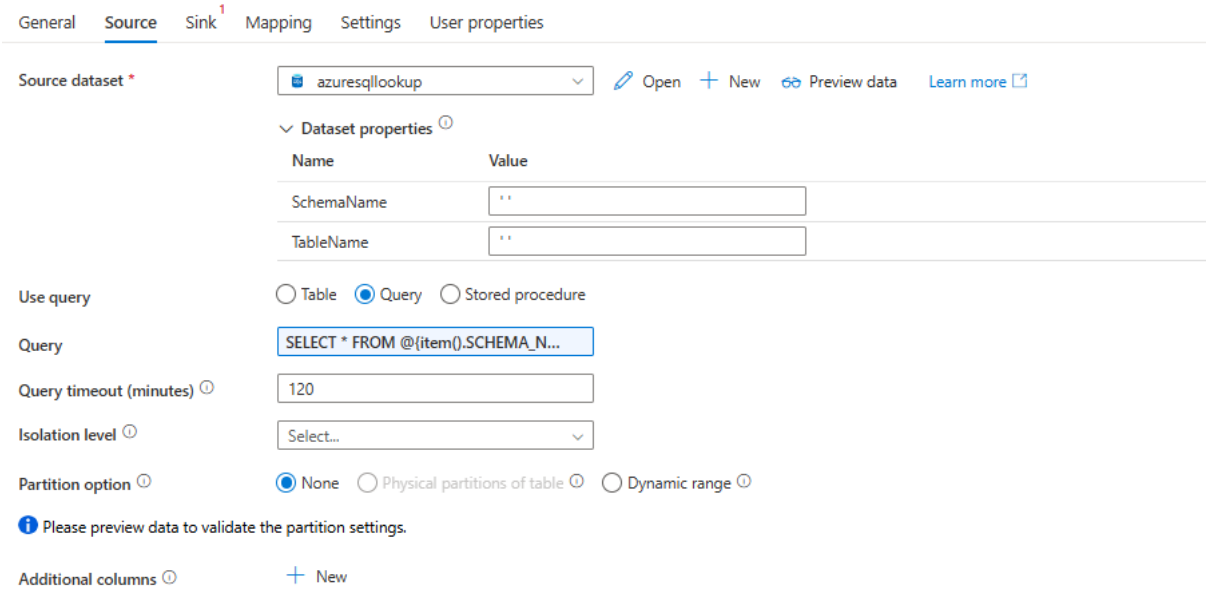
Write this below query in that dialog:

SELECT * FROM @{item().SCHEMA_NAME}.@{item().TABLE_NAME} WHERE @{item().DELTA_COLUMN}>'@{item().LPV}'



Click OK.

After that, the Source tab will look like this:

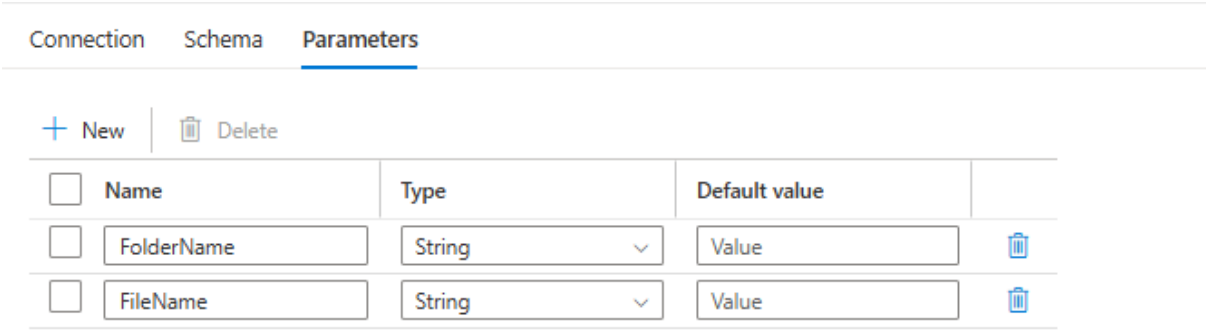


Go to the **Sink Tab** and Select **ADLS Gen 2** as the **sink dataset** and **DelimitedFile** for the destination file format

Go to Parameters Tab

Create two parameters

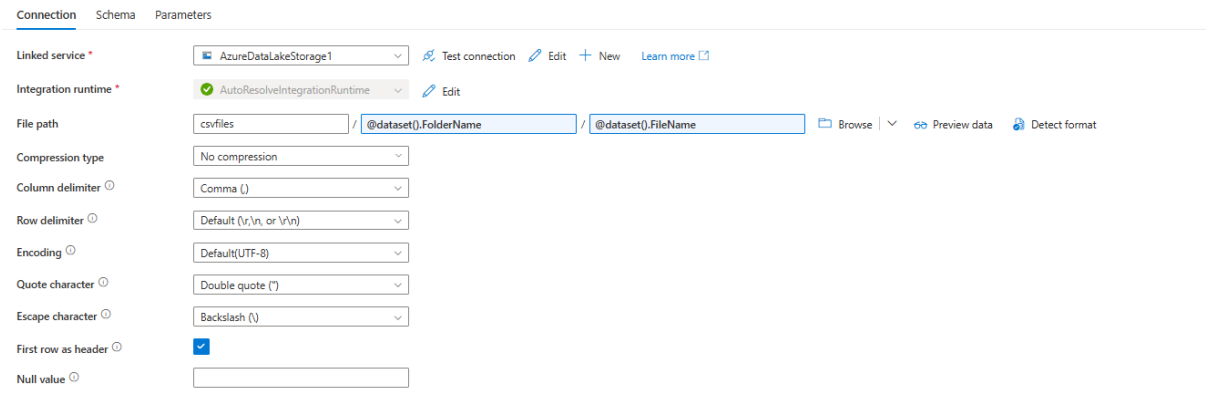
- **FolderName**
- **FileName**



Go to the Connection Tab

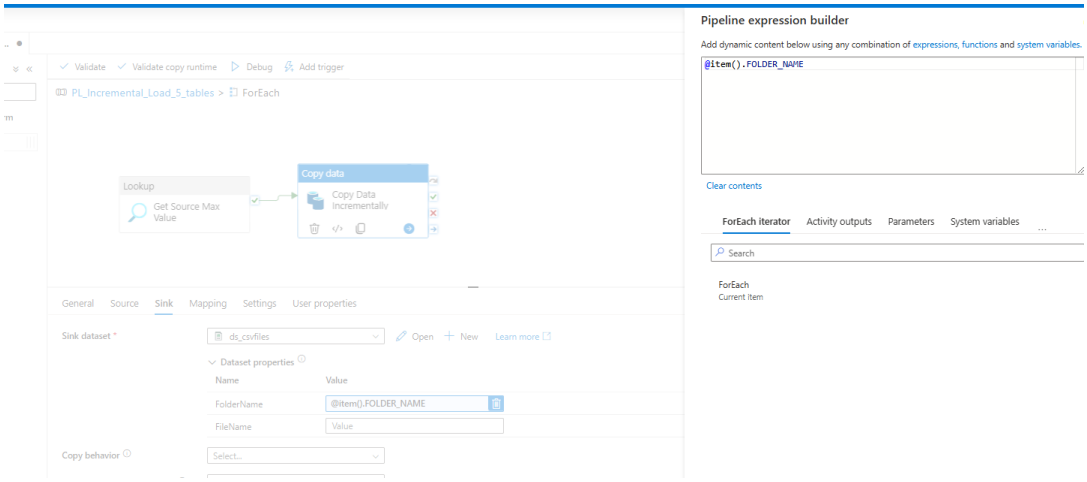
Assign those parameters by clicking on **Add Dynamic Content**

Connection tab will look like this.



Go back to the Sink Tab and provide Values for FolderName and FileName

So, for FolderName we will write as shown below by clicking on **Add Dynamic Content**

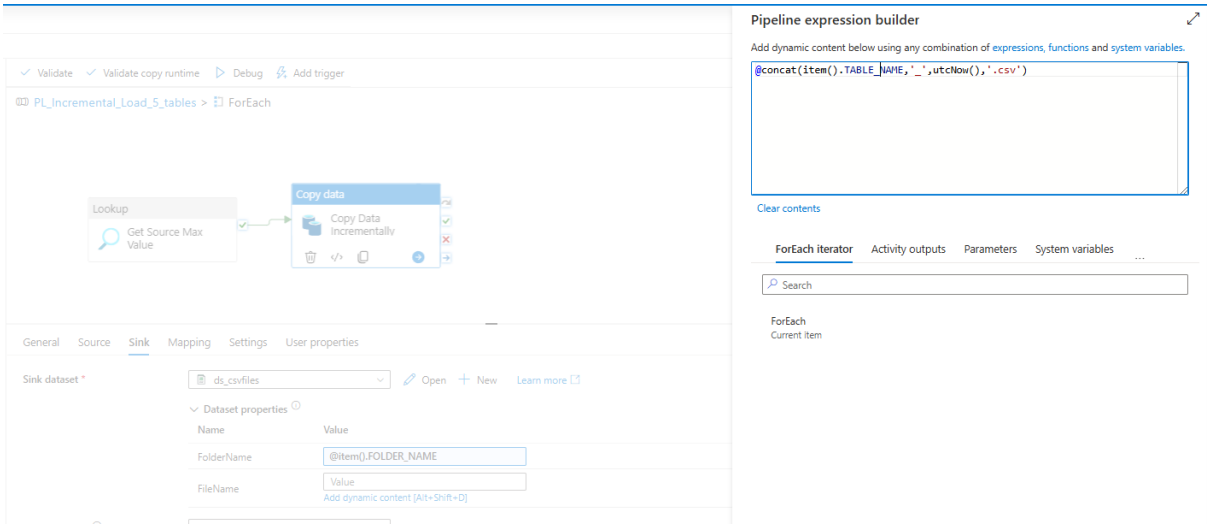


For the FileName, we will use this:

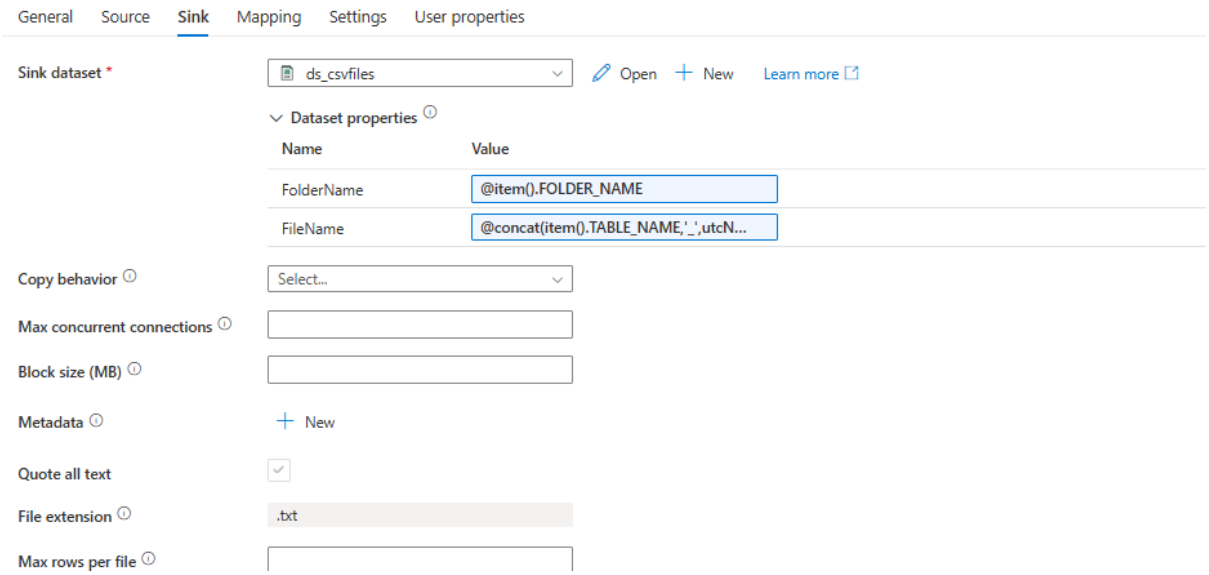
Concat function:

@concat(item().TABLE_NAME,'_',utcNow(),'.csv')

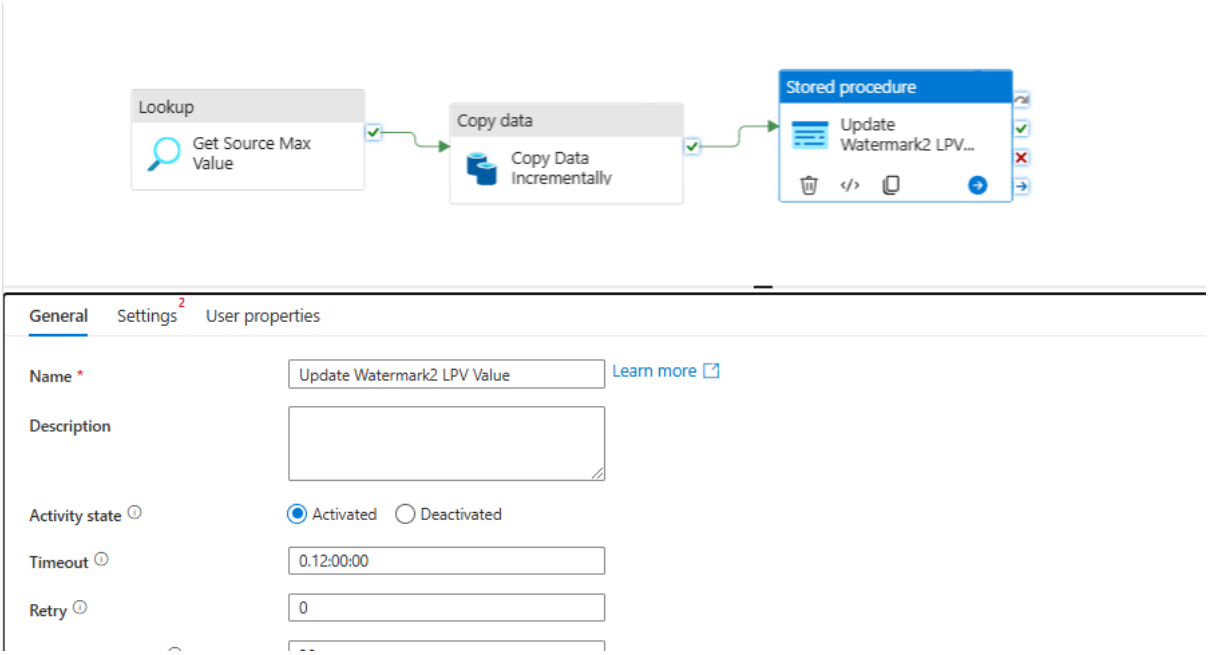
Why this: The expression in Azure Synapse dynamically constructs a CSV filename combining the table name with the current UTC timestamp, ensuring unique and time-stamped file names for data exports or logs.



After that, it will look like this in the Sink tab:

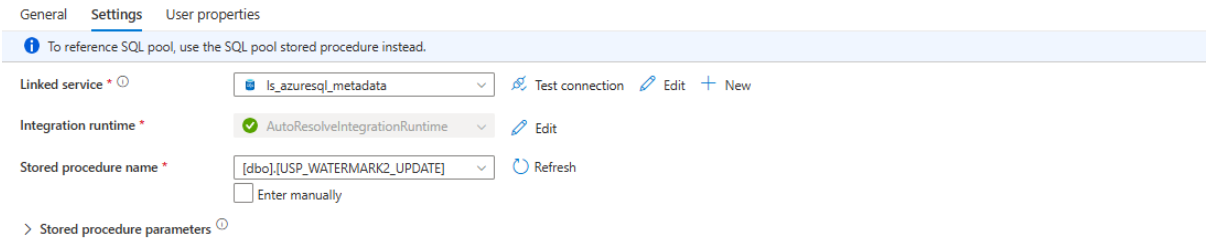


Now, drag and drop the **Stored Procedure** activity in the Design area and connect it from the Copy Data Activity as shown below:



Go to the **Settings tab** and select the **linked service** as selected before in the source/lookup activity (ADLS Gen 2)

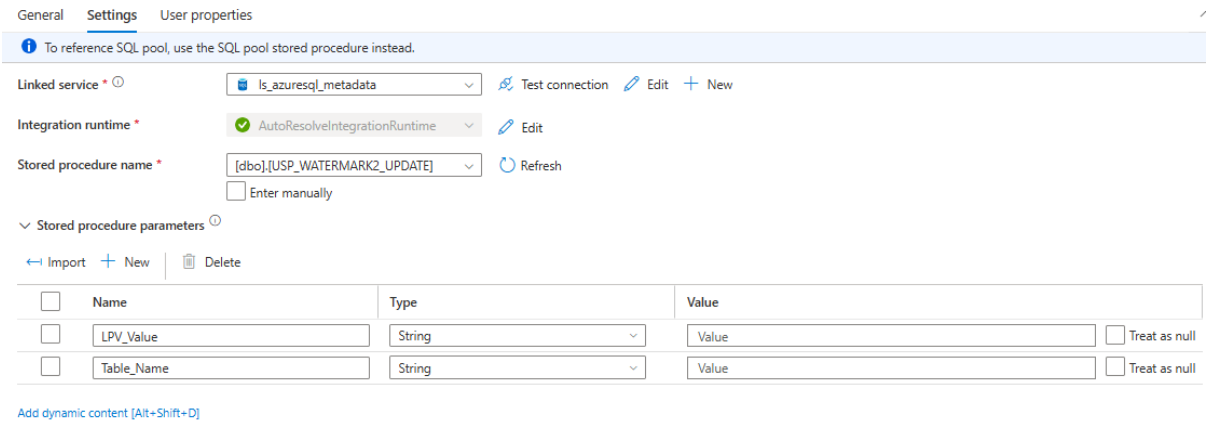
Select the **Stored Procedure name** from the dropdown



Click on **Stored Procedure Parameters**

And Click on **Import**

It will automatically take parameters from our stored procedure:



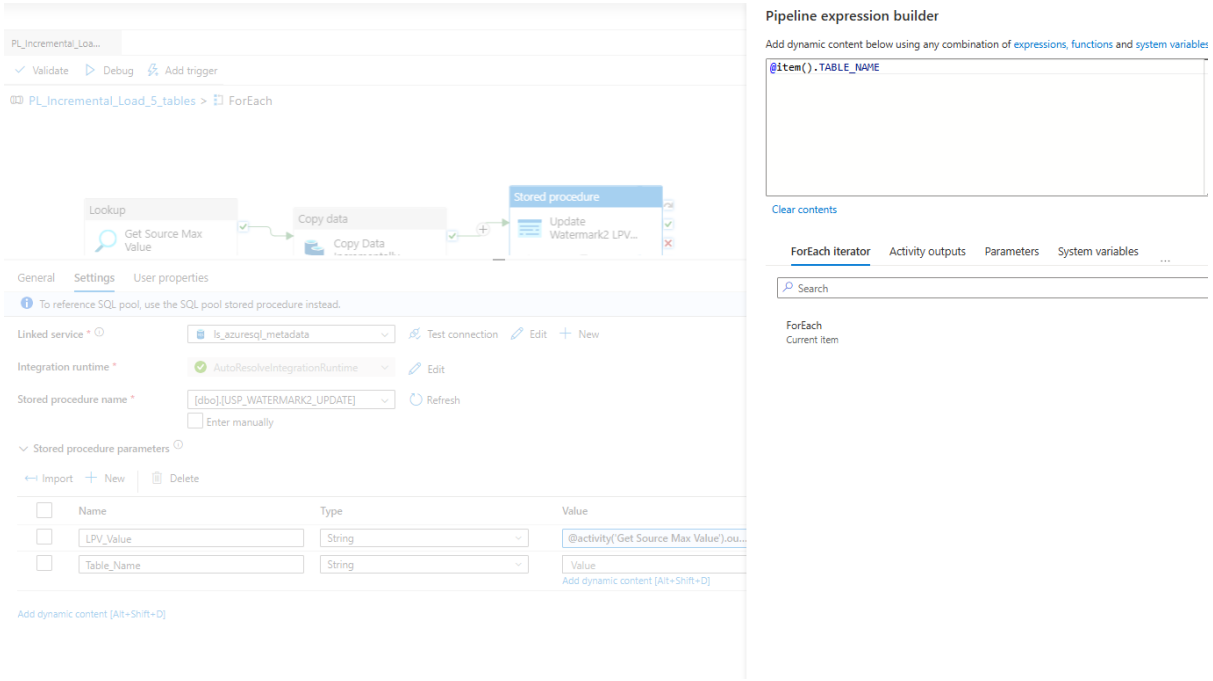
Now, Click on the value field for LPV_Value

Go to **Activity Output ->** Select **Get Source Max Value first row**

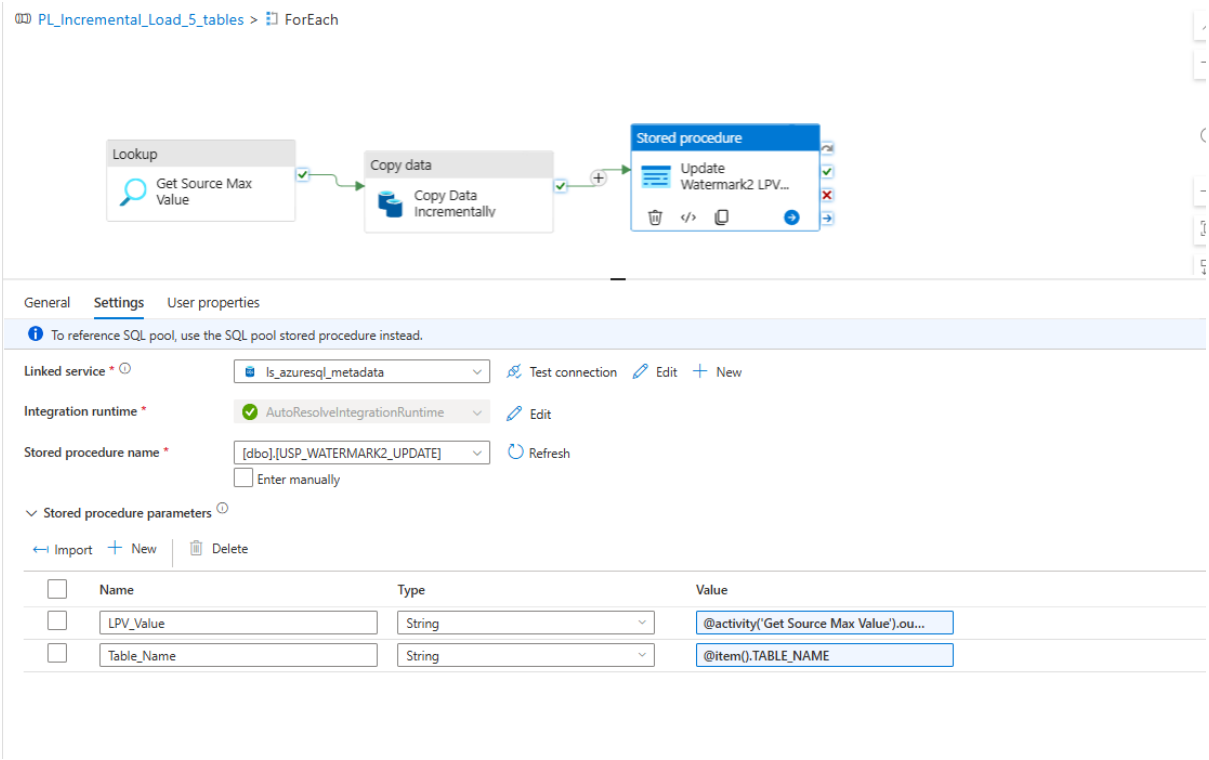
And it will automatically fill the above textbox with a query and we need to mention **MaxValue** (we did it in the second lookup activity inside ForEach) at the end of that query as shown below.

@activity('Get Source Max Value').output.firstRow.MaxValue

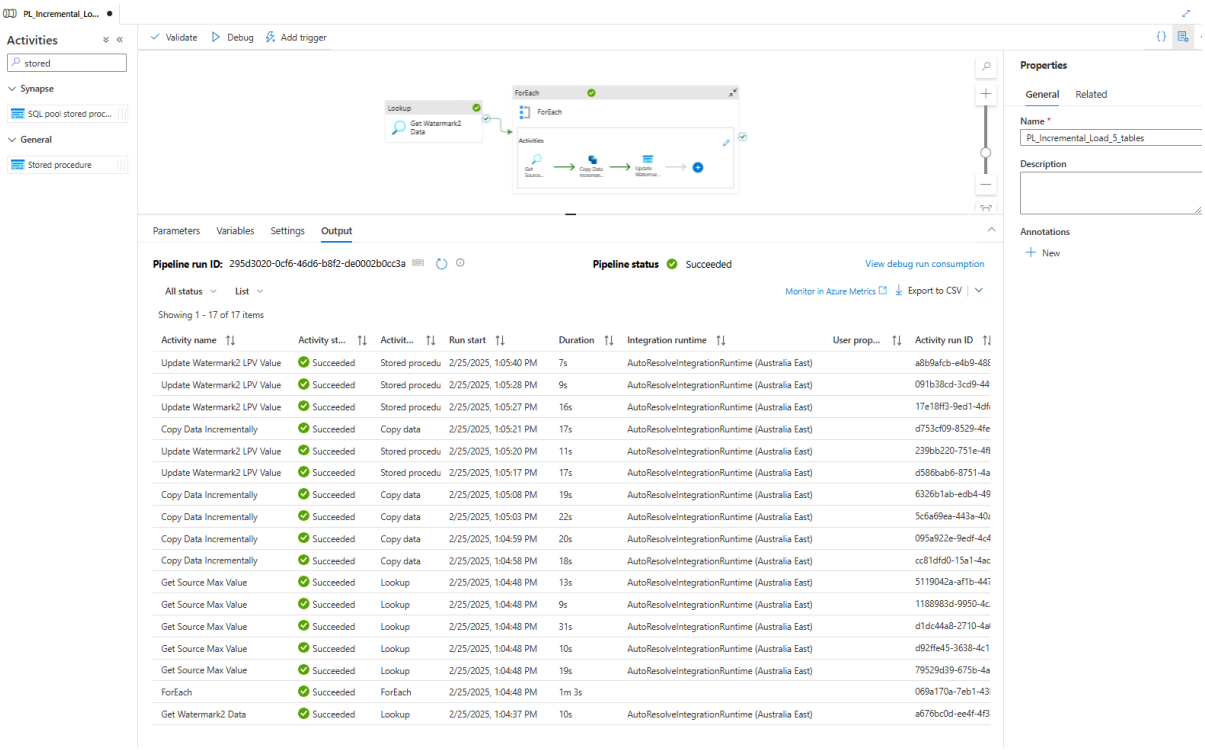
For the T**able name parameter**, select the ForEach current item from the dropdown, and match the SSMS Column name (TABLE_NAME in this case) of the Watermark2 table here, as shown below:



Stored procedure activity's settings tab It will look like this:



Debug the pipeline.



It was executed successfully.

Publish the Pipeline.

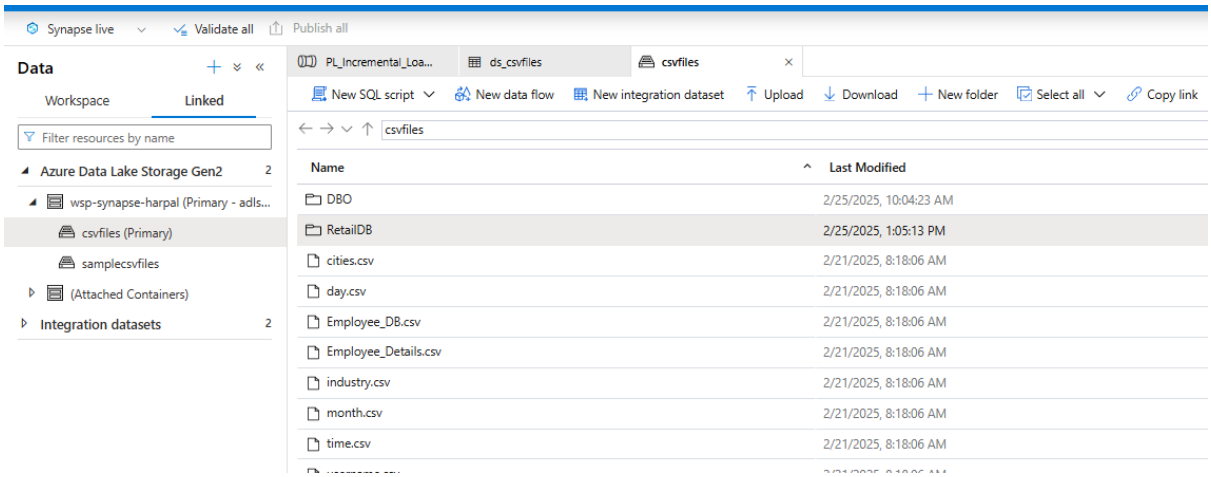
Now, let's check the Watermark2 table data

Results | Messages

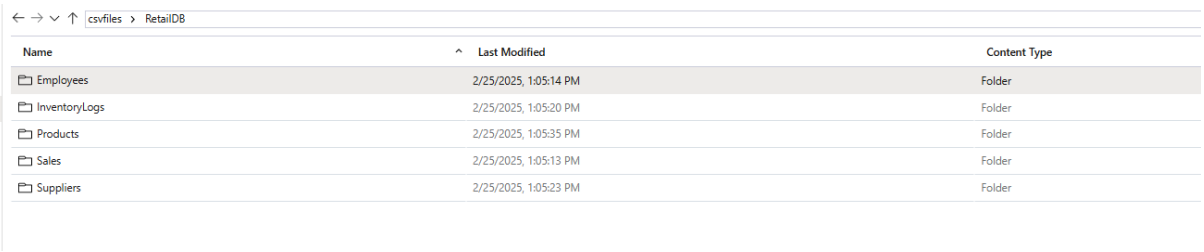
	ID	TABLE_NAME	SCHEMA_NAME	FOLDER_NAME	LPV	DELTA_COLUMN
1	1	Products	dbo	RetailDB/Products	4	product_id
2	2	Sales	dbo	RetailDB/Sales	2023-10-01T15:30:00	sale_date
3	3	Employees	dbo	RetailDB/Employees	2023-10-01T00:00:00	last_review_date
4	4	Suppliers	dbo	RetailDB/Suppliers	4	supplier_id
5	5	Inventory_Logs	dbo	RetailDB/InventoryLogs	2023-10-01T11:00:00	log_date

The **Watermark2 table has been updated correctly.**

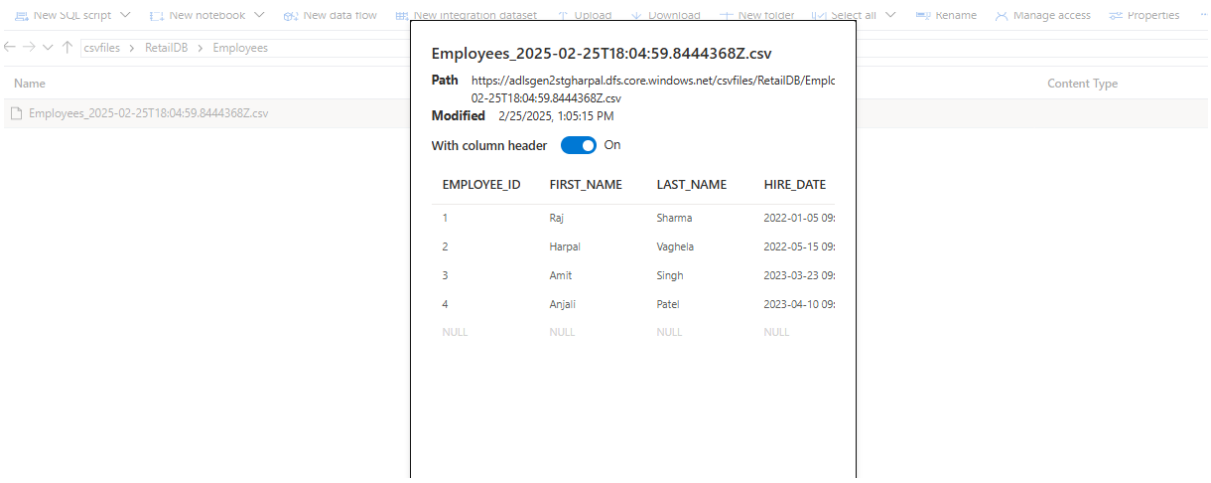
Now, let's check the data in the ADLS Gen2 storage



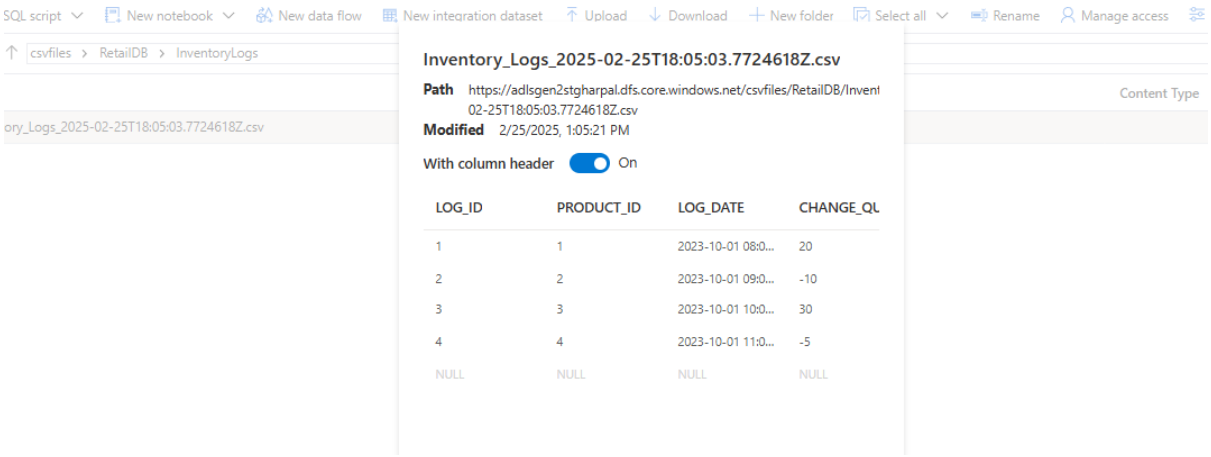
Go to ADLS Gen 2-> csvfiles -> RetailDB



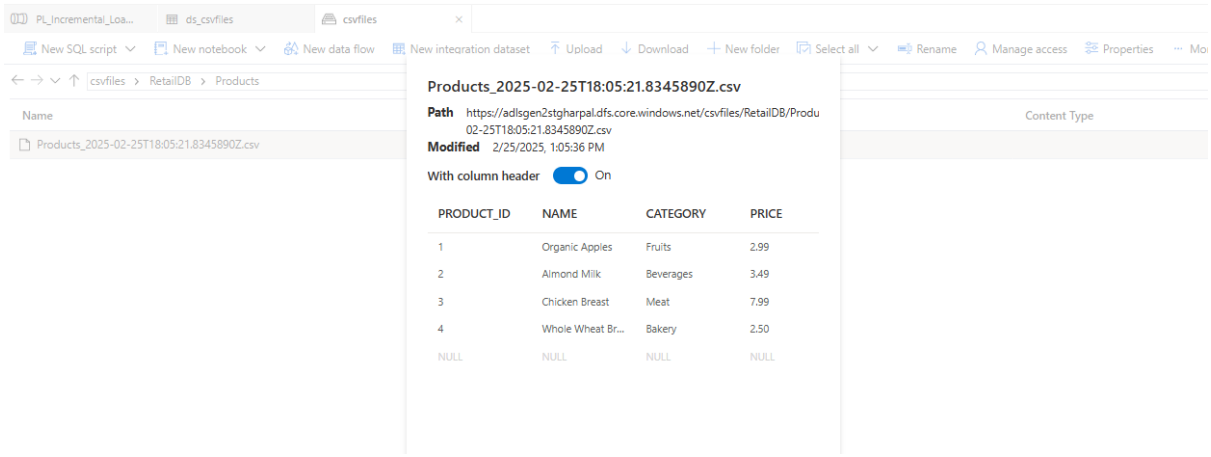
Employee Data:



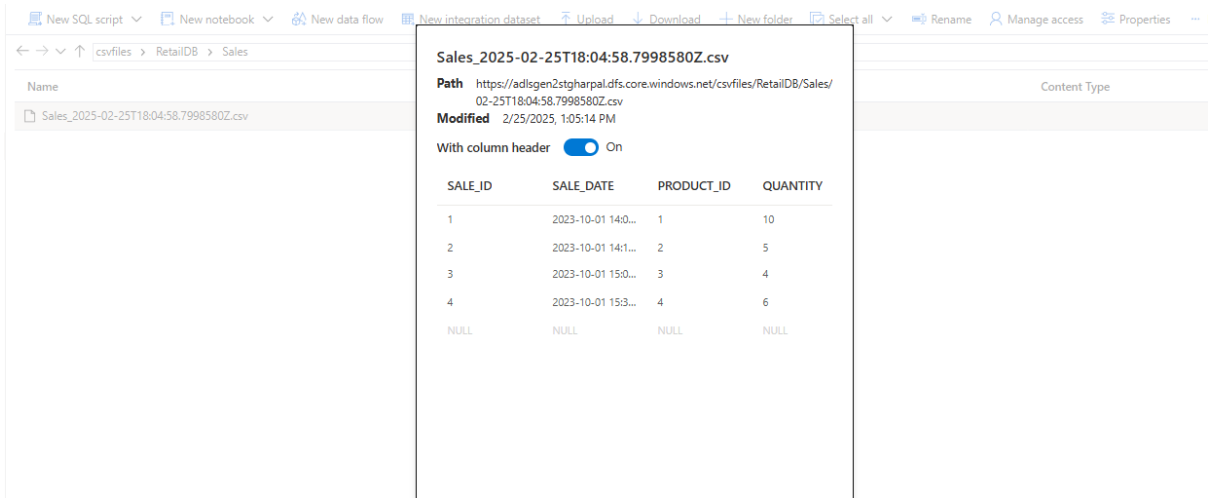
Inventory Logs Data:



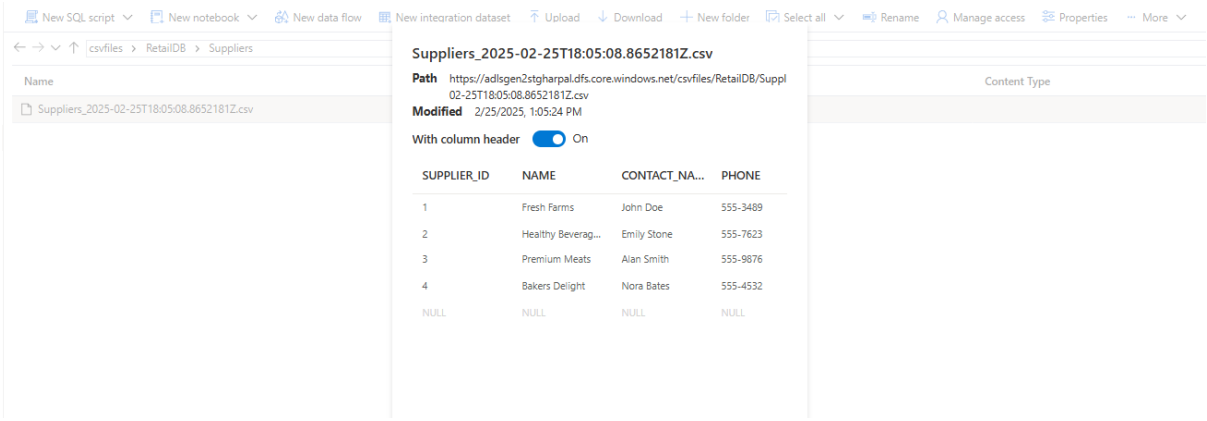
Products Data:



Sales Data:



Suppliers Data:



Now, let's insert some new records into tables:

----Insert new data into product table
INSERT INTO Products (product_id, name, category, price, stock, supplier_id) VALUES
(5, 'Vegetable Oil', 'Groceries', 4.50, 100, 2);

----Insert new data into Sales table
INSERT INTO Sales (sale_id, sale_date, product_id, quantity, total_amount, cashier_id) VALUES
(5, '2025-02-25 10:30:00', 5, 20, 90.00, 1);

----Insert new data into Employee table
INSERT INTO Employees (employee_id, first_name, last_name, hire_date, last_review_date, role) VALUES
(5, 'Nayan', 'Vaghela', '2025-01-01 09:00:00', '2025-02-20 00:00:00', 'Inventory Specialist');

```
132 INSERT INTO Products (product_id, name, category, price, stock, supplier_id) VALUES
133 (5, 'Vegetable Oil', 'Groceries', 4.50, 100, 2);
134
135 ----Insert new data into Sales table
136
137 INSERT INTO Sales (sale_id, sale_date, product_id, quantity, total_amount, cashier_id) VALUES
138 (5, '2025-02-25 10:30:00', 5, 20, 90.00, 1);
139
140 ----Insert new data into Employee table
141
142 INSERT INTO Employees (employee_id, first_name, last_name, hire_date, last_review_date, role) VALUES
143 (5, 'Nayan', 'Vaghela', '2025-01-01 09:00:00', '2025-02-20 00:00:00', 'Inventory Specialist');
144
145
146 Select * From Products
147 Select * From Employees
```

119 %

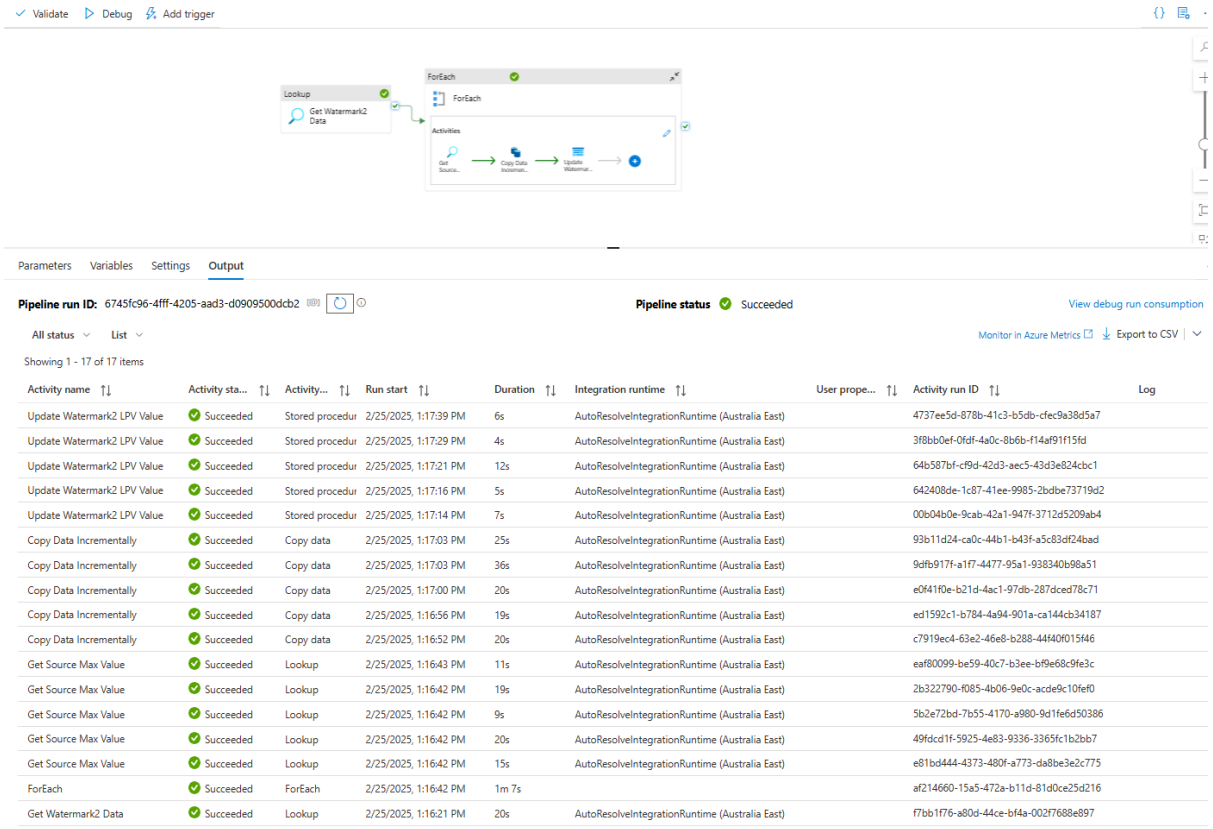
Results Messages

	product_id	name	category	price	stock	supplier_id
1	1	Organic Apples	Fruits	2.99	150	1
2	2	Almond Milk	Beverages	3.49	85	2
3	3	Chicken Breast	Meat	7.99	60	3
4	4	Whole Wheat Bread	Bakery	2.50	120	4
5	5	Vegetable Oil	Groceries	4.50	100	2

	employee_id	first_name	last_name	hire_date	last_review_date	role
1	1	Raj	Sharma	2022-01-05 09:00:00.000	2023-09-10 00:00:00.000	Cashier
2	2	Harpal	Vaghela	2022-05-15 09:00:00.000	2023-09-20 00:00:00.000	Cashier
3	3	Amit	Singh	2023-03-23 09:00:00.000	2023-09-30 00:00:00.000	Stock Manager
4	4	Anjali	Patel	2023-04-10 09:00:00.000	2023-10-01 00:00:00.000	Sales Manager
5	5	Nayan	Vaghela	2025-01-01 09:00:00.000	2025-02-20 00:00:00.000	Inventory Specialist

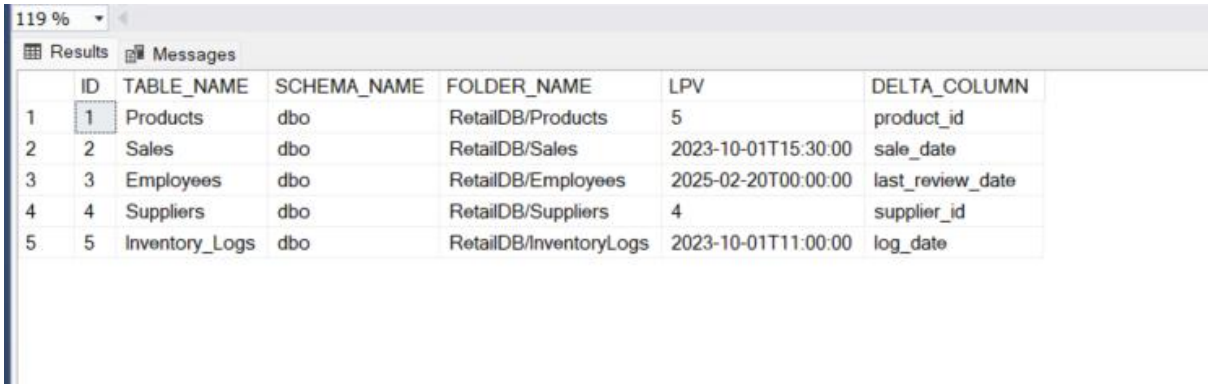
	sale_id	sale_date	product_id	quantity	total_amount	cashier_id
1	1	2023-10-01 14:00:00.000	1	10	29.90	1
2	2	2023-10-01 14:15:00.000	2	5	17.45	2
3	3	2023-10-01 15:00:00.000	3	4	31.96	1
4	4	2023-10-01 15:30:00.000	4	6	15.00	2

Now, let's run the pipeline again.



Activity name	Activity sta...	Activity...	Run start	Duration	Integration runtime	User prope...	Activity run ID	Log
Update Watermark2 LPV Value	Succeeded	Stored procedur	2/25/2025, 1:17:39 PM	6s	AutoResolveIntegrationRuntime (Australia East)		4737ee5d-878b-41c3-b5db-cfec9a38d5a7	
Update Watermark2 LPV Value	Succeeded	Stored procedur	2/25/2025, 1:17:29 PM	4s	AutoResolveIntegrationRuntime (Australia East)		3f8bb0ef-0fdf-4a0c-8b6b-f14af91f15fd	
Update Watermark2 LPV Value	Succeeded	Stored procedur	2/25/2025, 1:17:21 PM	12s	AutoResolveIntegrationRuntime (Australia East)		64b587bf-cf9d-42d3-aec5-43d3e824cbc1	
Update Watermark2 LPV Value	Succeeded	Stored procedur	2/25/2025, 1:17:16 PM	5s	AutoResolveIntegrationRuntime (Australia East)		642408de-1c87-41ee-9985-2bdbe73719d2	
Update Watermark2 LPV Value	Succeeded	Stored procedur	2/25/2025, 1:17:14 PM	7s	AutoResolveIntegrationRuntime (Australia East)		00b04b0e-9cab-42a1-947f-3712d5209ab4	
Copy Data Incrementally	Succeeded	Copy data	2/25/2025, 1:17:03 PM	25s	AutoResolveIntegrationRuntime (Australia East)		93b11d24-ca0c-44b1-b43f-a5c83df24bad	
Copy Data Incrementally	Succeeded	Copy data	2/25/2025, 1:17:03 PM	36s	AutoResolveIntegrationRuntime (Australia East)		9dfb917f-a1f7-4477-95a1-938340b98a51	
Copy Data Incrementally	Succeeded	Copy data	2/25/2025, 1:17:00 PM	20s	AutoResolveIntegrationRuntime (Australia East)		e0f41f0e-b21d-4ac1-97db-287dced78c71	
Copy Data Incrementally	Succeeded	Copy data	2/25/2025, 1:16:56 PM	19s	AutoResolveIntegrationRuntime (Australia East)		ed1592c1-b784-4a94-901a-ca144cb34187	
Copy Data Incrementally	Succeeded	Copy data	2/25/2025, 1:16:52 PM	20s	AutoResolveIntegrationRuntime (Australia East)		c7919ec4-63e2-46e8-b288-44f40f015f46	
Get Source Max Value	Succeeded	Lookup	2/25/2025, 1:16:43 PM	11s	AutoResolveIntegrationRuntime (Australia East)		eaf80099-be59-40c7-b3ee-bf9e68c9fe3c	
Get Source Max Value	Succeeded	Lookup	2/25/2025, 1:16:42 PM	19s	AutoResolveIntegrationRuntime (Australia East)		2b322790-f085-4b06-9e0c-acde9c10fef0	
Get Source Max Value	Succeeded	Lookup	2/25/2025, 1:16:42 PM	9s	AutoResolveIntegrationRuntime (Australia East)		5b2e72bd-7b55-4170-a980-9d1fe6d50386	
Get Source Max Value	Succeeded	Lookup	2/25/2025, 1:16:42 PM	20s	AutoResolveIntegrationRuntime (Australia East)		49fdcd1f-5925-4e83-9336-3365fc1b2bb7	
Get Source Max Value	Succeeded	Lookup	2/25/2025, 1:16:42 PM	15s	AutoResolveIntegrationRuntime (Australia East)		e81bd444-4373-480f-a773-da8be3e2c775	
ForEach	Succeeded	ForEach	2/25/2025, 1:16:42 PM	1m 7s			af214660-15a5-472a-b11d-81d0ce25d216	
Get Watermark2 Data	Succeeded	Lookup	2/25/2025, 1:16:21 PM	20s	AutoResolveIntegrationRuntime (Australia East)		f7bb1f76-a80d-44ce-bf4a-002f7688e897	

Let's check the data and watermark2 table:

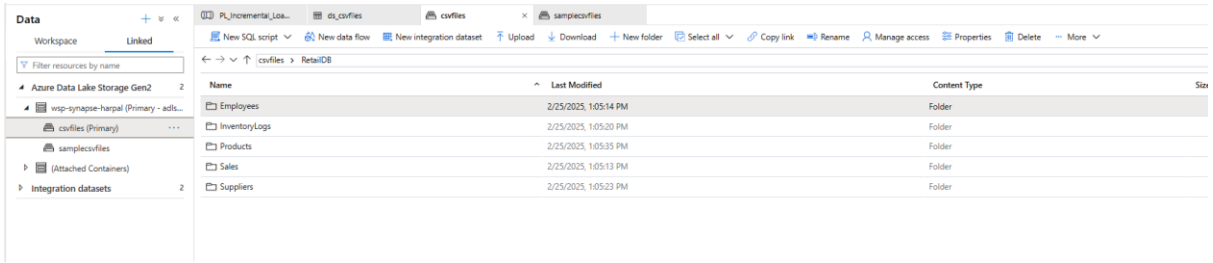


	ID	TABLE_NAME	SCHEMA_NAME	FOLDER_NAME	LPV	DELTA_COLUMN
1	1	Products	dbo	RetailDB/Products	5	product_id
2	2	Sales	dbo	RetailDB/Sales	2023-10-01T15:30:00	sale_date
3	3	Employees	dbo	RetailDB/Employees	2025-02-20T00:00:00	last_review_date
4	4	Suppliers	dbo	RetailDB/Suppliers	4	supplier_id
5	5	Inventory_Logs	dbo	RetailDB/InventoryLogs	2023-10-01T11:00:00	log_date

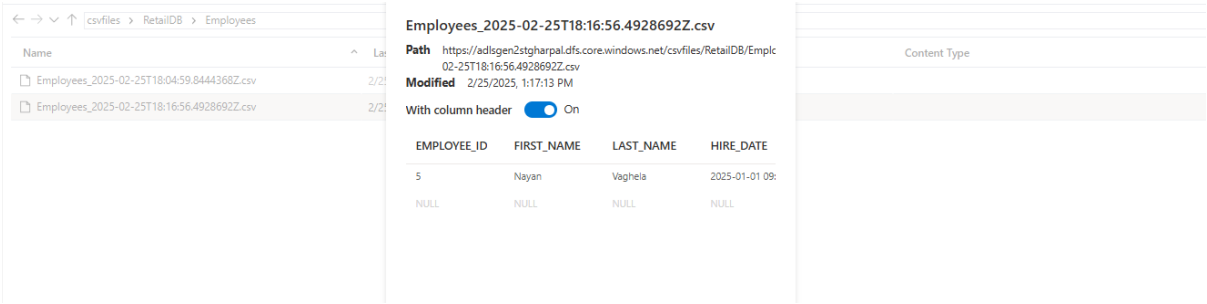
The **Watermark2 table has been updated correctly.**

Now, let's check the data in the ADLS Gen2 storage

Go to ADLS Gen 2-> csvfiles -> RetailDB

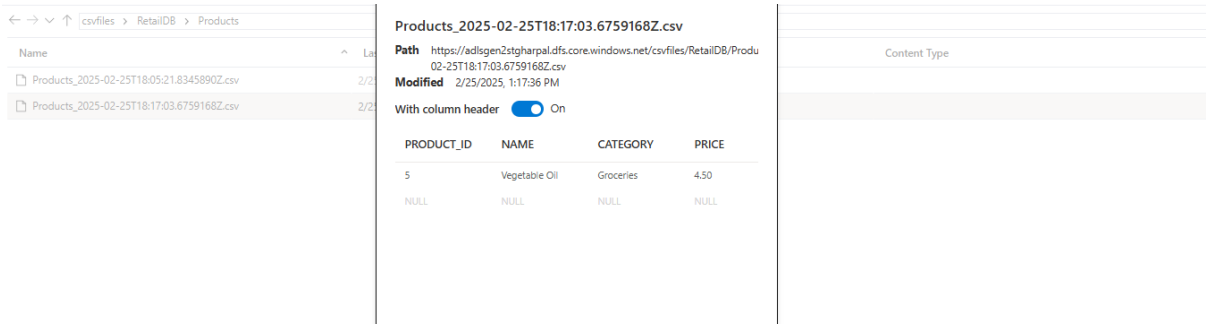


Employee Data:



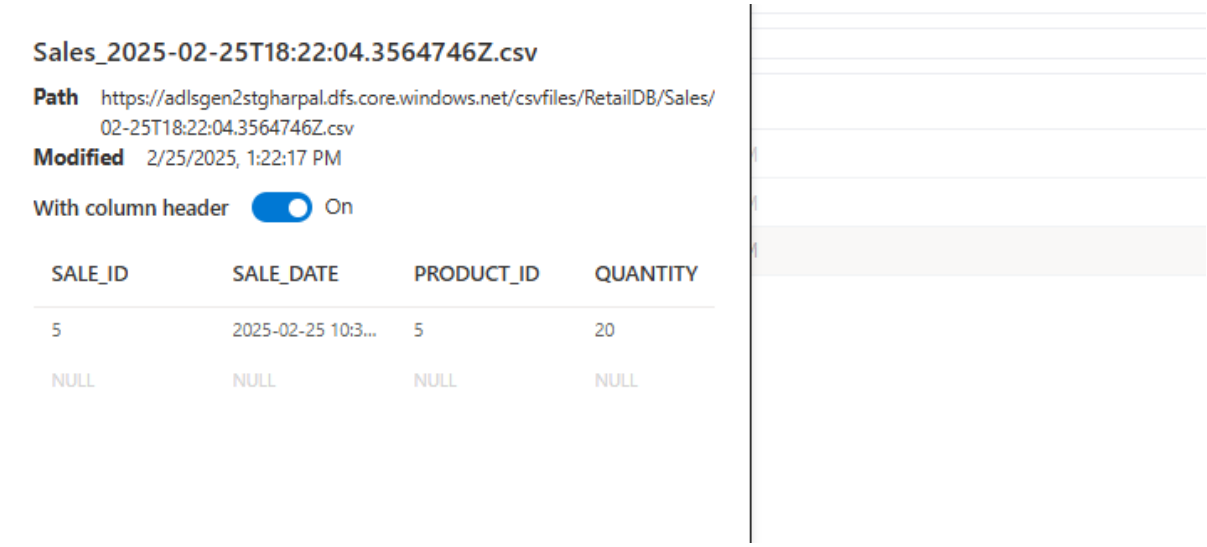
The Employee record is updated successfully in destination storage.

Products Data:



The Products table record is updated successfully in destination storage.

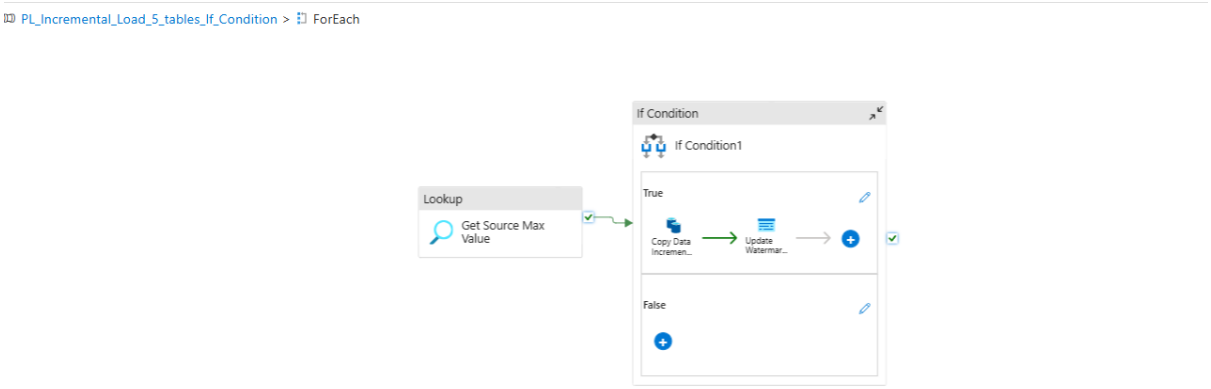
Sales Data:



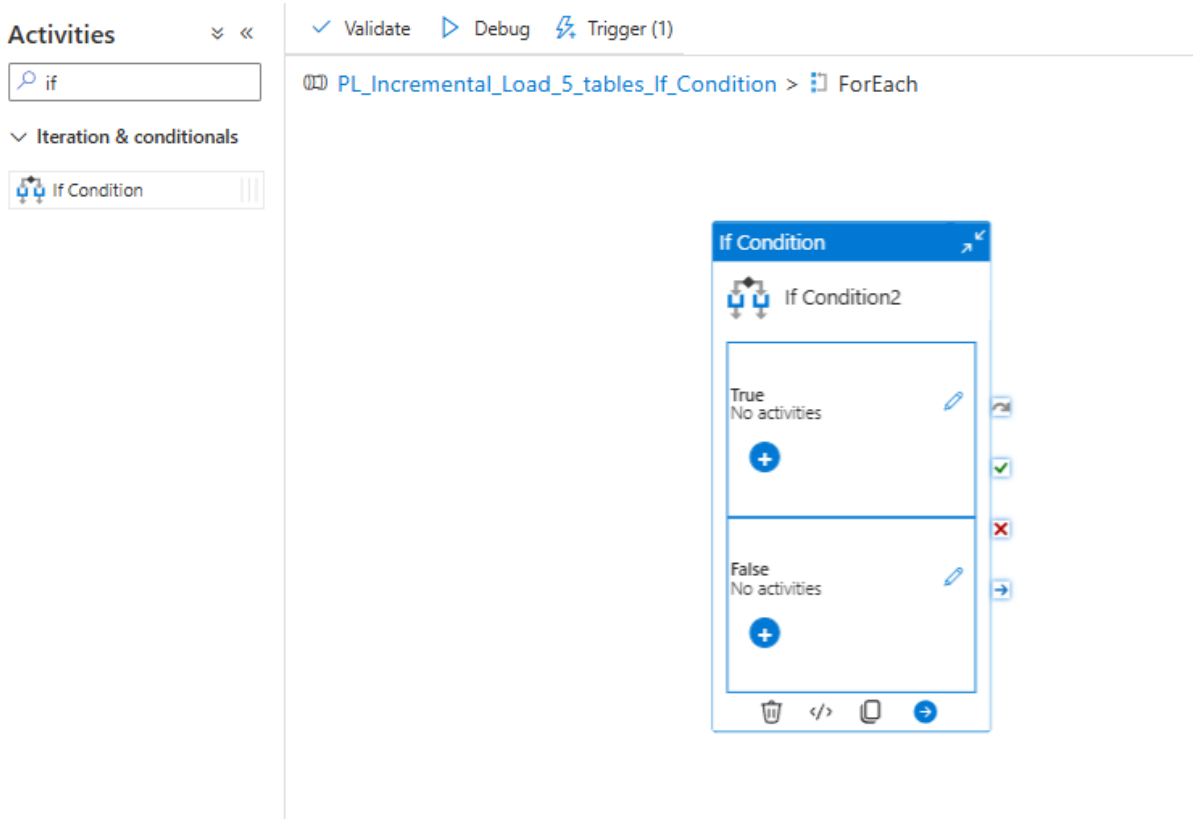
The Sales table record is updated successfully in destination storage.

Put If Condition, so no file with empty records will be created.

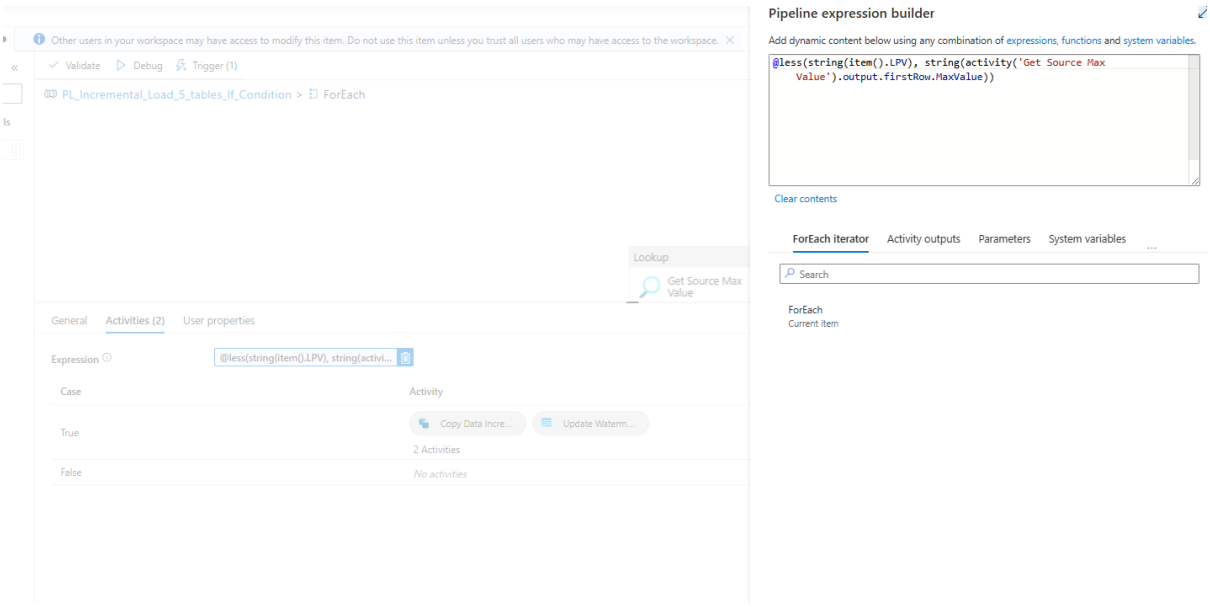
Let's work on this now,



Drag and drop **if condition** from activities tab.



Go to Activities in If condition and in Expression field write this expression:



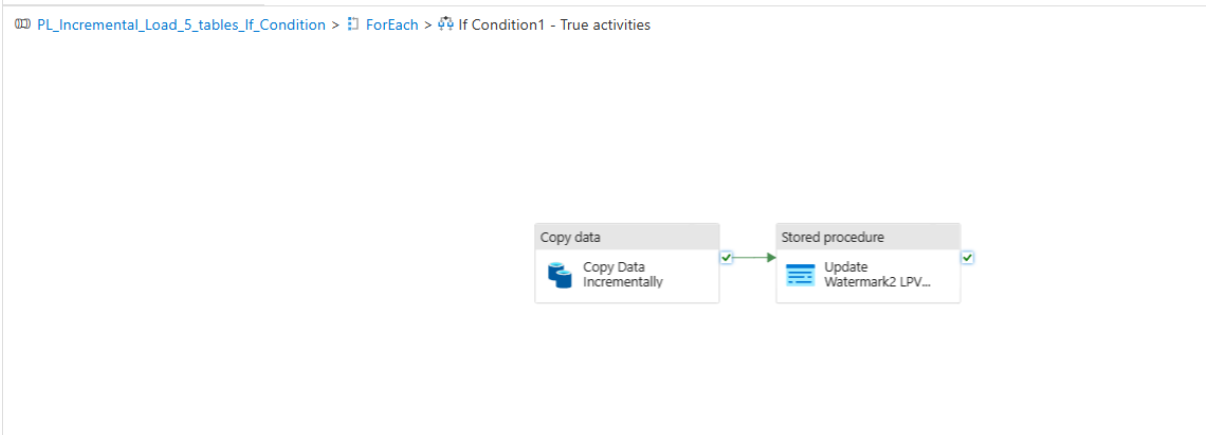
@less(string(item().LPV), string(activity('Get Source Max Value').output.firstRow.MaxValue))

Now, click on **Edit icon of "if condition" activity,**

Here, move previously created copy data and stored procedure activities.

It will look like this:

You can see we're in forEach-> If Condition -> True Activities



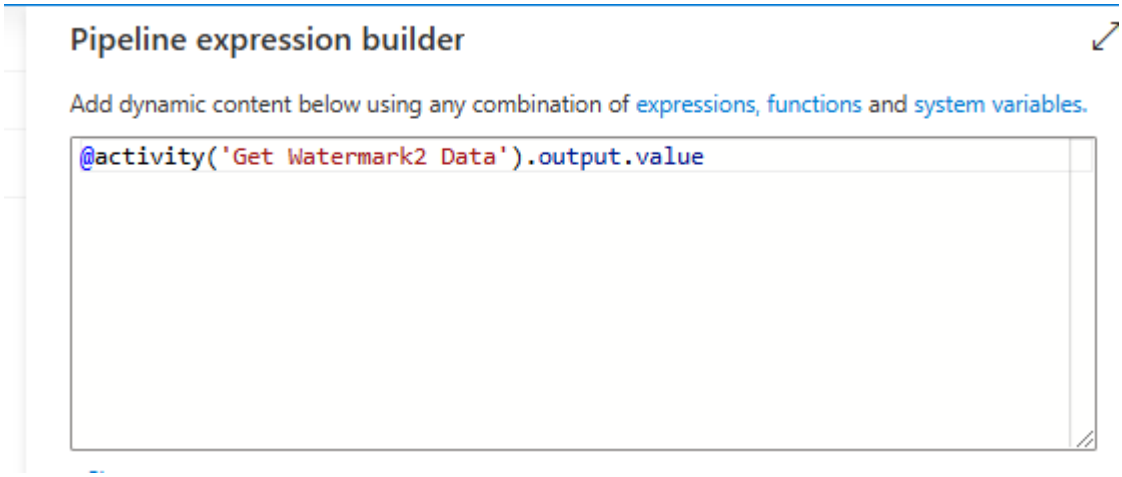
Publish the Pipeline again.

And Check the Data.

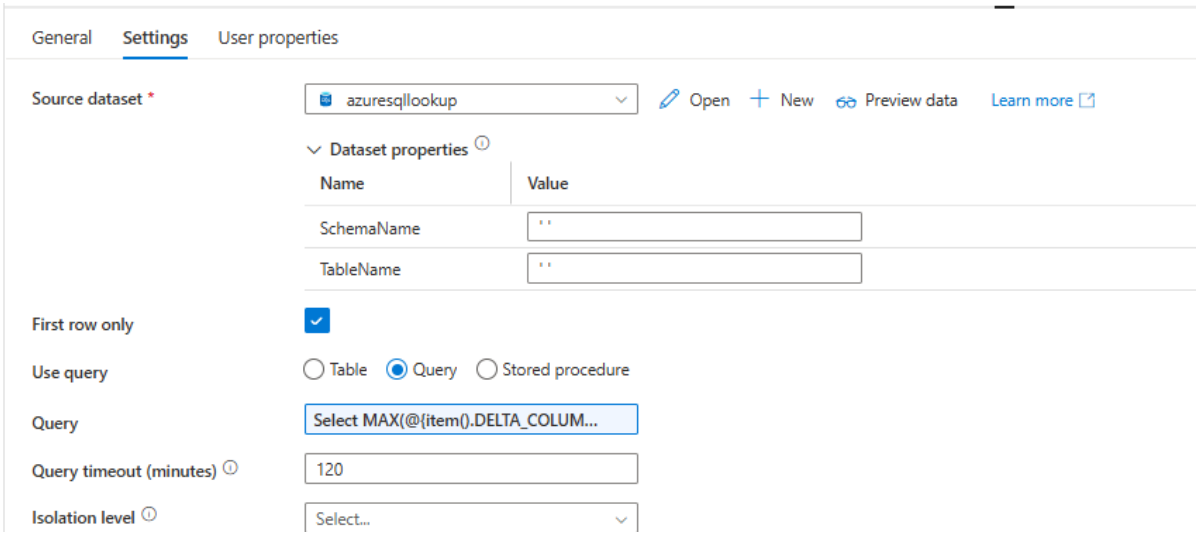
Now, it will not create any csv file with empty data.

Points to Remember:

- Expression in **ForEach** Activity should be a value array like this:

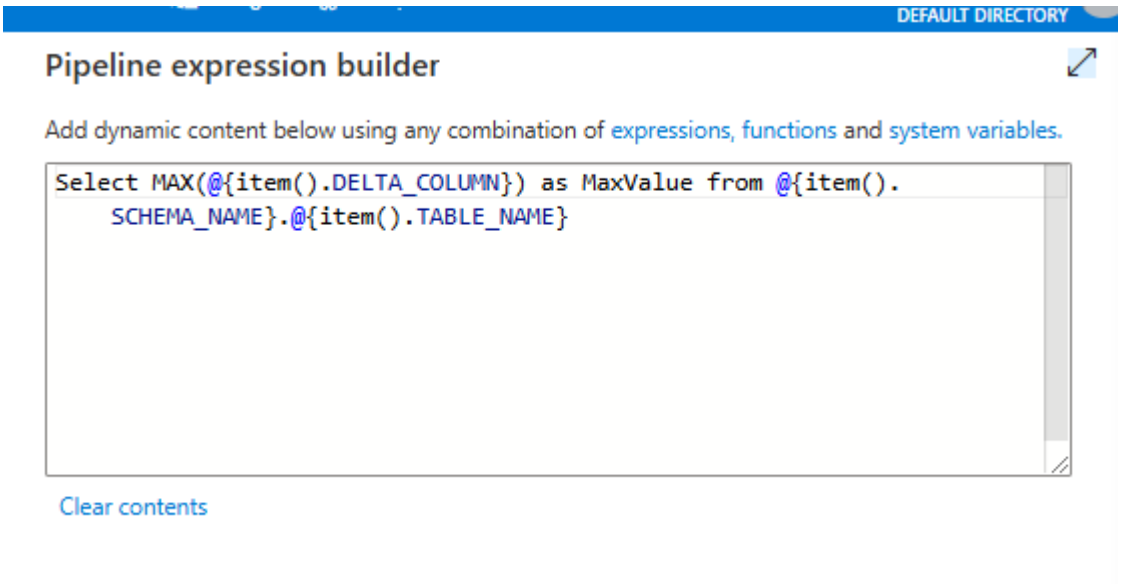


In the ForEach Activity, we utilize a Lookup table to retrieve the MaxValue from a source dataset previously employed, hence the system prompts for schema and table name inputs, which are not necessary in this context. The MaxValue is obtained through a custom query. When employing a new dataset, schema name and table name fields are absent. Conversely, when reusing an older dataset as indicated, it is essential to enclose value fields in single quotes to bypass them and execute the custom query.

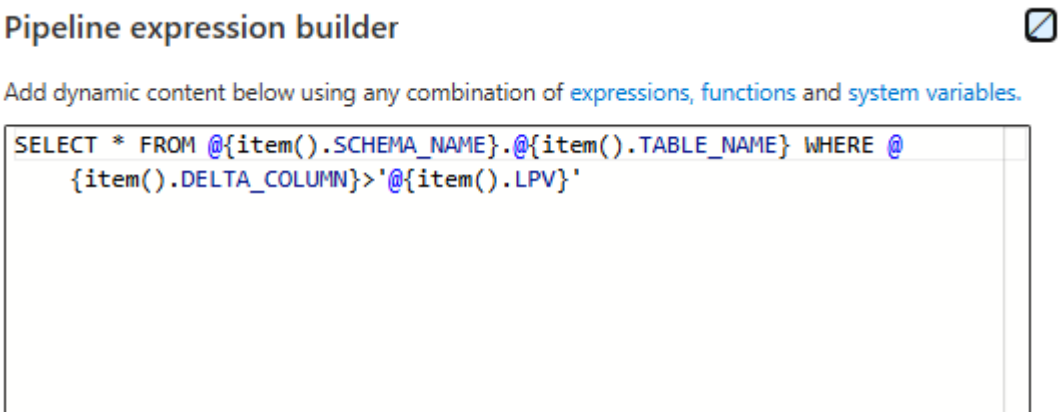


Custom Query:

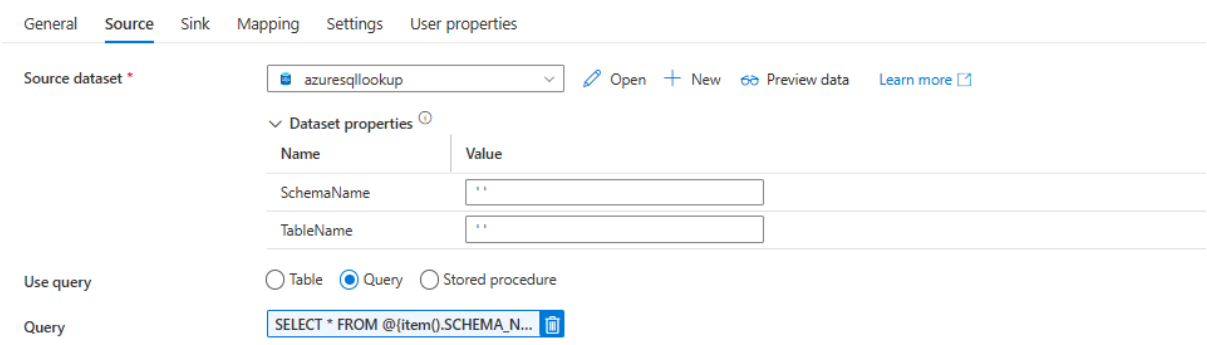
- Remember this alias name as MaxValue as we will need this variable in the stored procedure to update the LPV value.



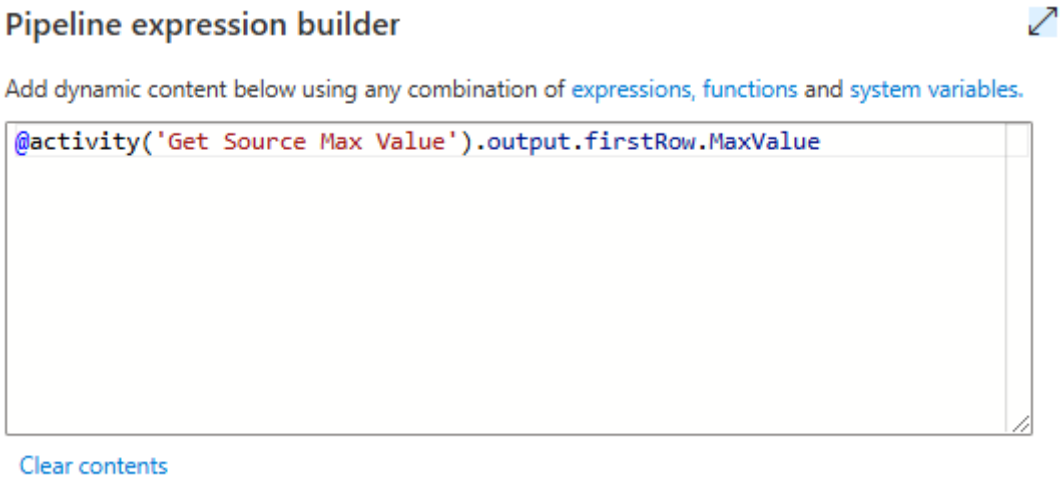
- Copy Data Activity inside If Condition
  - Make sure to write table, schema name, LPV and delta column name as same as SSMS watermark column names.



- In the Source configuration of the Copy Data Activity, we are using an older dataset, which prompts the request for parameter values. We are addressing this by enclosing these parameters in single quotation marks as shown below, effectively bypassing the request:



- In the Stored Procedure configuration, when parameters are imported, it prompts for the Table name and LPV value entries. For the LPV value, an expression should be used. Ensure that the variable name from the Lookup activity within the ForEach activity (MaxValue) is correctly specifie



Thank you for your time!

You can connect with me on:

LinkedIn: https://www.linkedin.com/in/harpalvaghela/