**Sentmail Email Report**

**Sources Details:**

**Azure SQL Database:**

**Server:** sql-m365-prod-ne-01.database.windows.net

**Database:** db-m365-logs-prod-ne-01

**Table:** 1. Sentitems.user\_sentitems

2. Sentitems.datelookup

**View:** VW\_EXTERNAL\_EMAIL\_REPORTING

**Azure Data factory:**

**Pipelines**

1. Copy\_0365\_Data

2. Sentitems\_Trans

3. user-details

4. userdetails\_Transformation

**Azure Storage Account:** strgpm365ne01

**Container:** m365-logs/sentitems

**General Email Domains:** [General Email Domains.xlsx](https://investcorp.sharepoint.com/:x:/s/tech/ETp94xGkc09OjDBf_QHVVpkBdxKVhL87nmRn338AdyOSQQ?e=sAAPZc)

**Microsoft 365 Admin Center:**

1. First, we need to create an active group that will contain the people in our organization who are allowed to approve the Microsoft data connect request.
2. To do so, Go to Microsoft 365 Admin center
3. Group-> Active Groups-> Add a Group->
4. Group Type: Mail-enabled Security and click on next
5. Provide a Group Name & Description: XXXXXX & XXXXXX and click on next
6. Provide a group email address: [xxxx@XXXX.com](mailto:xxxx@XXXX.com)
7. Review and create.
8. Once the group is available in the active group list simply click on it you will get the property pane. From that pane select Members tab -> View all & Manage members -> Add Members to the group who can approve the Azure data requests.

**Azure AD Registration:**

1. In this Step go to <https://portal.azure.com/>
2. Go to Home and search & select Azure active Directory -> App Registration
3. Click on New registration Button at the top & provide the Application name (**INVESTCORP**) and necessary details.
4. In the INVESTCORP, you will see the Application (client) ID and Directory (tenant) ID. Please make sure you make a note of the Application (client) ID and Directory (tenant) ID because we are going to need them later.
5. From there, click on Certifications and Secrets from the left navigation, In that click on a new client secret under Client Secrets. Provide the Description and Expires range. This Will create a new secret in our App & please make sure to take a note of value because we are going to use this later.

**Azure Storage account:**

Now, we need to create an azure storage account to store the files that are going to be exported from M365 Exchange. To do so, we are going to create a new Azure storage account. Search for storage accounts in the home page. Select Storage accounts.

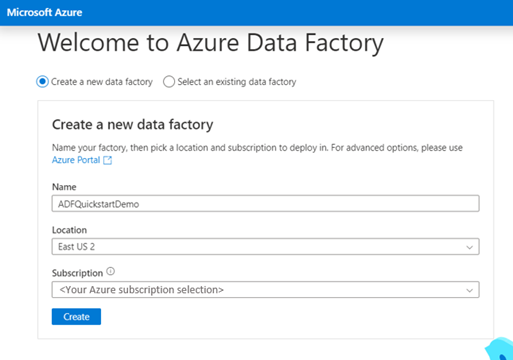
1. Click on the add button in the top menu.
2. From there we provide necessary details like **Storage account name** **& Resource group** and click on next
3. On the advanced tab click on **Enable Hierarchical namespace** and click on next until we get the **Review + Create** Option.
4. Once the account has been created. Click on **Go To resource** button which will bring your storage account page.
5. Now we are going to create our actual containers. Go to containers from the left menu.
6. From there click on the **+ Container** button in the top Menu to create a new one.
7. Give it a **Name (m365-logs)** and choose **Public access level type** and click on **Create.**
8. This will Create a new container called **m365-logs.** Click on **m365-logs** &open it.
9. We are going to add a new directory by clicking on the **+ Add Directory** Button. Give it a name as **sentitems** & click on Save. This will create a new directory inside the **m365-logs. Sentitems** is the actual directory where we going to store all the files exported from M365.
10. At the storage account level, we need to give grant access to our Azure AD Application access to write back to the storage account. From left menu, click on **Access Control (IAM)** -> **Add Role Assignments** Button. Select **Role** as Storage Blob Data Contributor, **Assign Access to** as User, group, or Service principal, **Select** asINVESTCORP and click save to grant the permissions to the application.

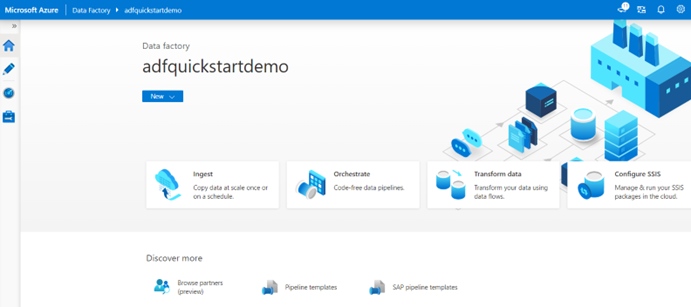
All the prerequisites are in place now we are ready to start the extract our data from office 365 first step to do an enable the Microsoft graph data connect at the tenant level.

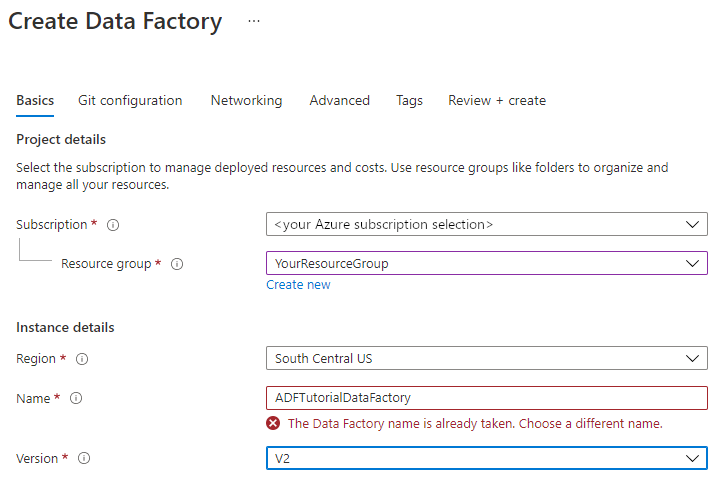
1. Go to Microsoft 365 Admin Center
2. From the left navigation select settings -> Org settings.
3. Select Microsoft Graph Data Connect which will open a panel in the right.
4. select the checkbox of Turn Microsoft Graph Data Connect On or off for your entire organization and choose a default approval group.
5. In the search group we need to select a group that we created earlier as a prerequisite; Approval group people are the one who is going to approve the requests coming from Microsoft Graph data connect.
6. Save the changes. Now the Microsoft Graph data connect is enabled from tenant level.

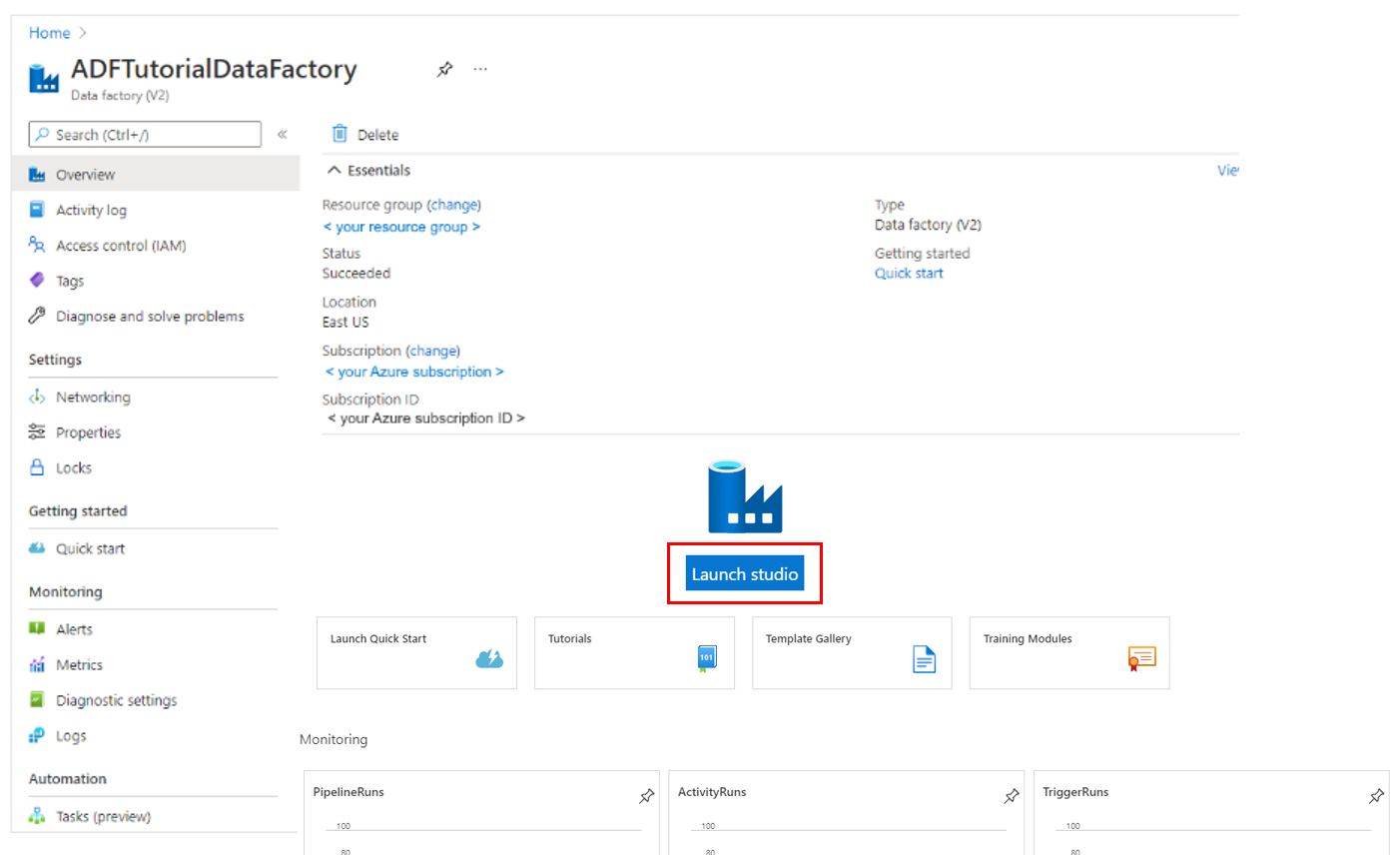
**Azure Data Factory:**

1. Now go to <https://portal.azure.com/>
2. In the Home page Search & select Data Factories
3. Click on + add and give it a name and resource group details and click on next until you get an option **Review + Create**
4. Once it is created click on **Go To resource.**









**Create a Pipeline(Copy\_0365\_Data):**

To copy the data from Microsoft 365 to Azure blob storage we need a pipeline.

1. Go to azure Data factories and click on Author from the left navigation icons.
2. Click on the + symbol just below the factory resources, you will get more options, select pipeline.
3. From the activities drag and drop a **copy data** activity into the canvas.
4. Click on the copy activity & select source tab.
5. Create a new **source dataset** by clicking on the **+New** symbol
6. In the search box search for office 365 and select office 365 from the results and click on continue.
7. Now we need to edit the dataset. Click on the open option just beside the source dataset.
8. Now we need to create a new Linked Service, click on **+New,** in that we need provide the information about Service principal ID and Service principal key and click on **create** button.
9. From the table dropdown menu choose the **BasicDataSet\_v0. Sentitem\_v1**

Graphical user interface, application

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, application

Description automatically generated

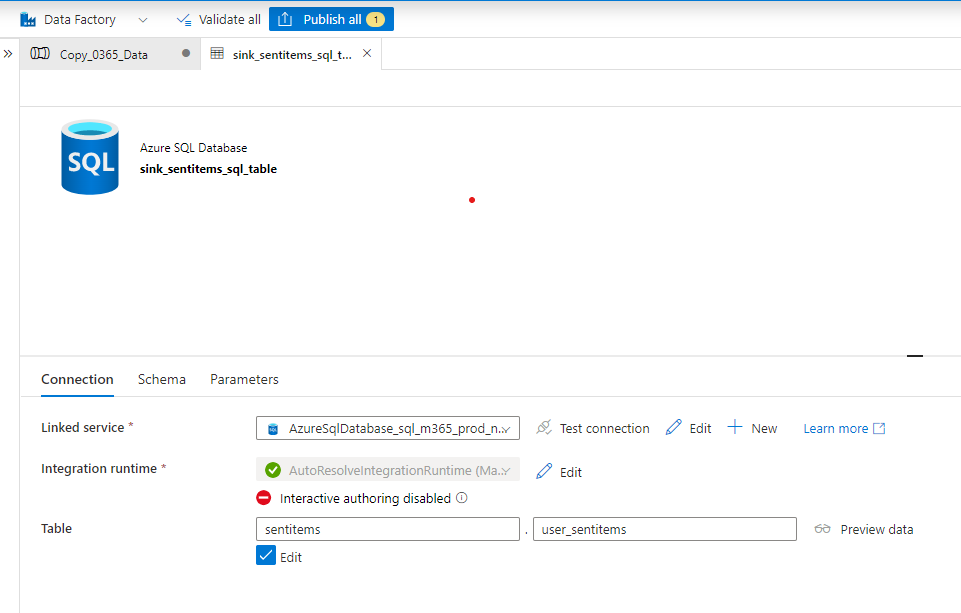
Graphical user interface, text, application, email

Description automatically generated

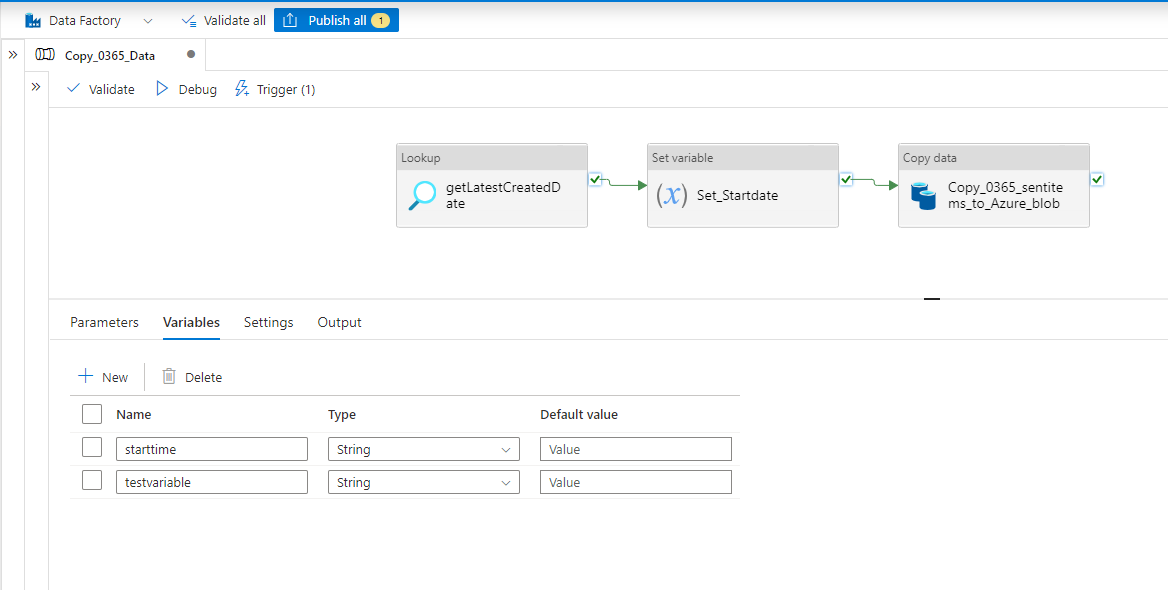
1. Now we can go back to pipeline tab from the top navigation. Go to canvas
2. Under activities search and drag the Lookup into canvas. Click on it, In the General tab Give it a name(**getLatestCreatedDate**) and description. Now go to setting tab choose a **source dataset** as **sink\_sentitems\_sql\_table** and click on **open** then,connection tab add a Linked service as **AzuresqlDatabase\_sql\_m365\_prod\_ne\_01** & Table as **Sentitems.user\_sentitems** and in theschema tab Import the schema.

Graphical user interface, text, application

Description automatically generated



1. Now go back to the Lookup settings, in that, choose **Query** as Use query, in the below query add the following sql query “**select MAX(startdate) as startdate from [Sentitems].[datelookup]**”, query timeout(minutes) as 120, isolation level as none and Partition Option as none.
2. Now click anywhere in the canvas empty space and go to variable tab create a variable as **starttime; string; value**

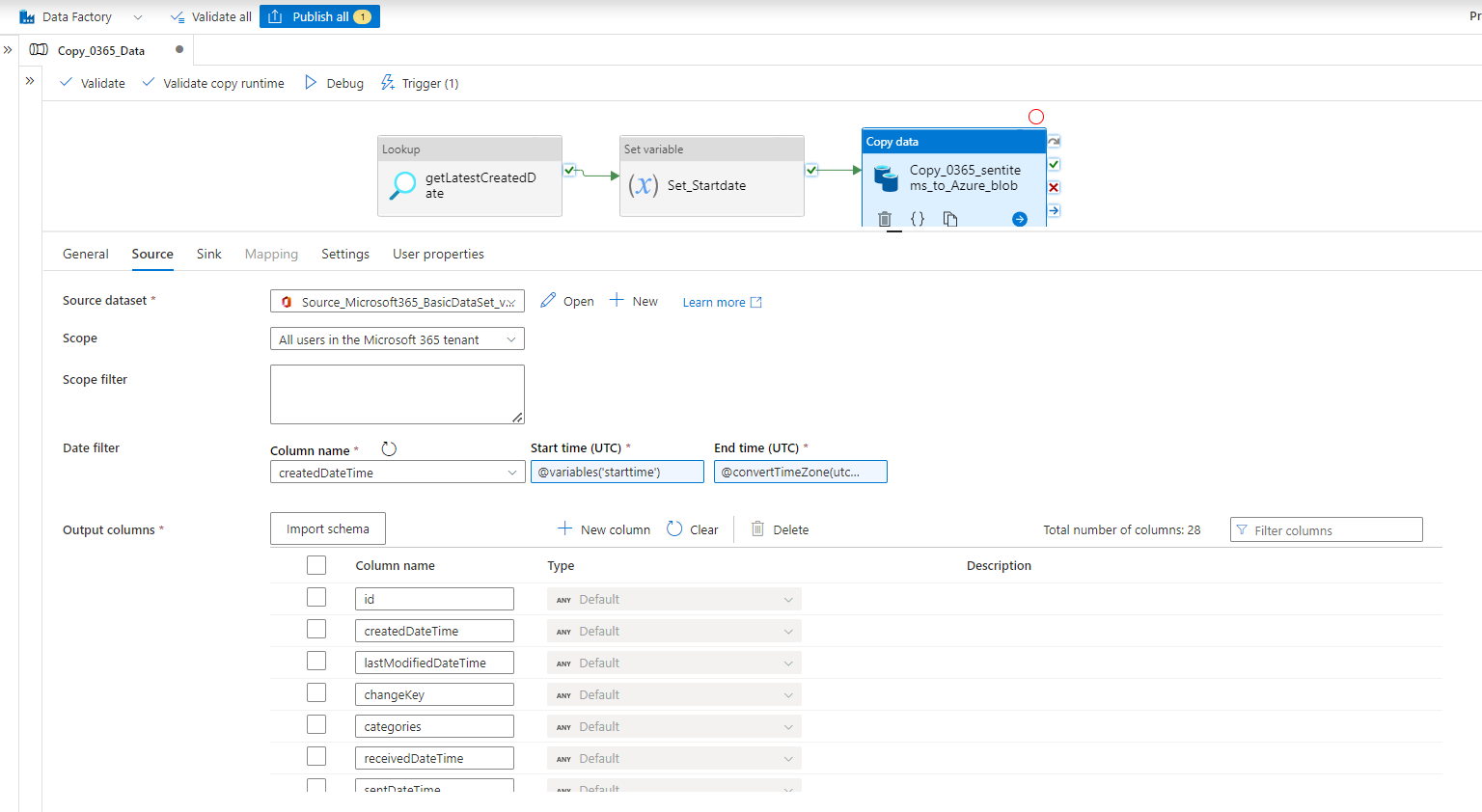


1. Now under the activities search and drag the Set Variable into the canvas. Click on it, in the General tab Give it a name (**set\_startdate** and description). Go to settings tab choose name as **starttime &** value as “**@activity('getLatestCreatedDate').output.firstRow.startdate”.**
2. Create a connection between activities like **getLatestCreatedDate(Lookup) -> set\_startdate(Set Variable) -> Copy\_o365\_sentitems\_to\_Azure\_blob(Copy data).**
3. Now click on the copy data and go to source tab, click on the import the schema In date filter choose column name as **createdDateTime** & start time (UTC) as **@variables('starttime')** & End time (UTC) as **@convertTimeZone(utcNow(),'UTC','Arab Standard Time').**

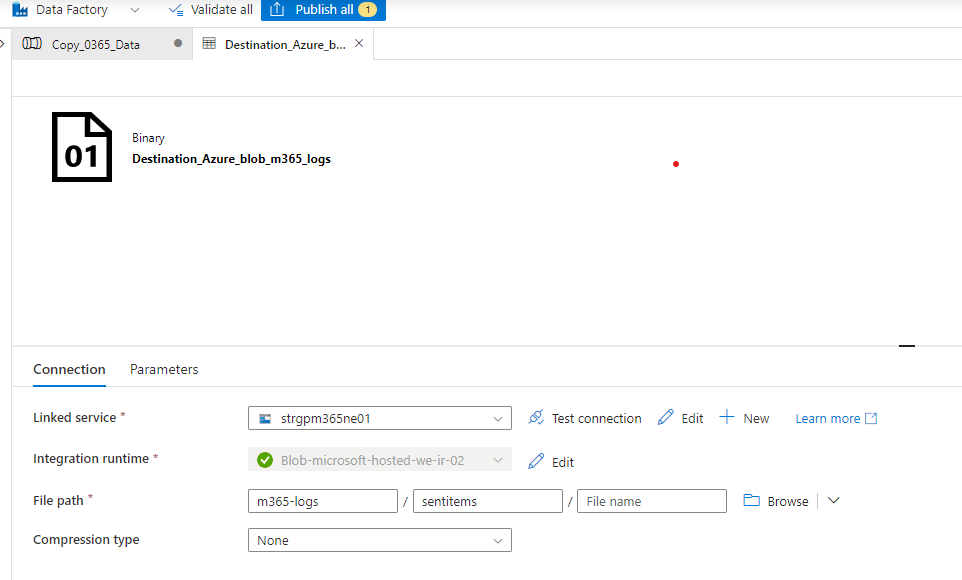
Graphical user interface, text

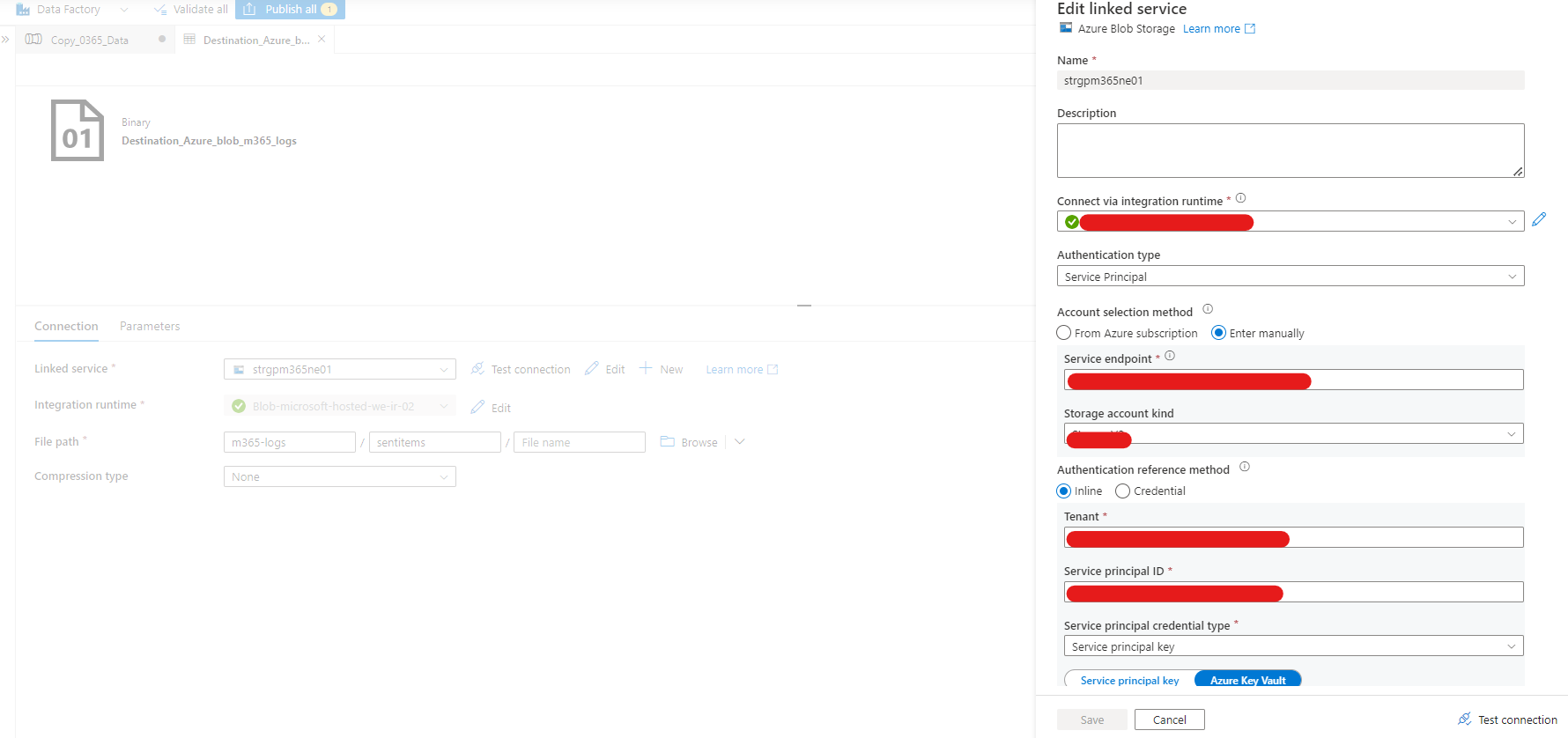
Description automatically generated

1. Click on import schema which will load all the columns available in the dataset, in the list of columns select the body and body preview columns and delete those because we don’t have access to see the actual content.



1. Next thing we need to do is create a sink, a sink is a location where the exported file is going to be located. Now go to sink tab. Click on **+new** to create a new sink connection choose azure blob storage, click on continue and choose the binary format, click on continue. From there in set properties choose a linked service. From the authentication method dropdown select service principal & select storage account name and service principal ID and service principal key and click on test connection to make we were able to connect. Click on the file path and select the location that we have already created in the storage account > m365-logs > sentitems.

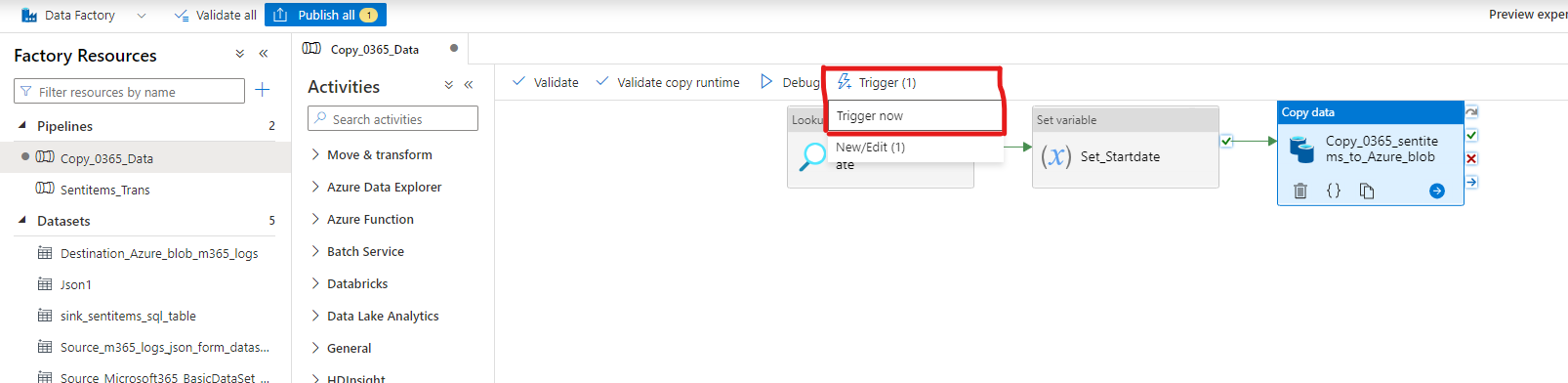




1. Now we are ready to go and publish the changes by clicking Publish all.

**Trigger the pipeline:**

1. From the canvas click on Trigger > Trigger Now this is going to automatically turn on the pipeline in order to monitor the execution click on the monitor button from the left navigation. From the pipeline runs we can see that our current pipeline is being executed; if we click on the entry for a pipeline, we can get the additional details like what’s going on with the execution.

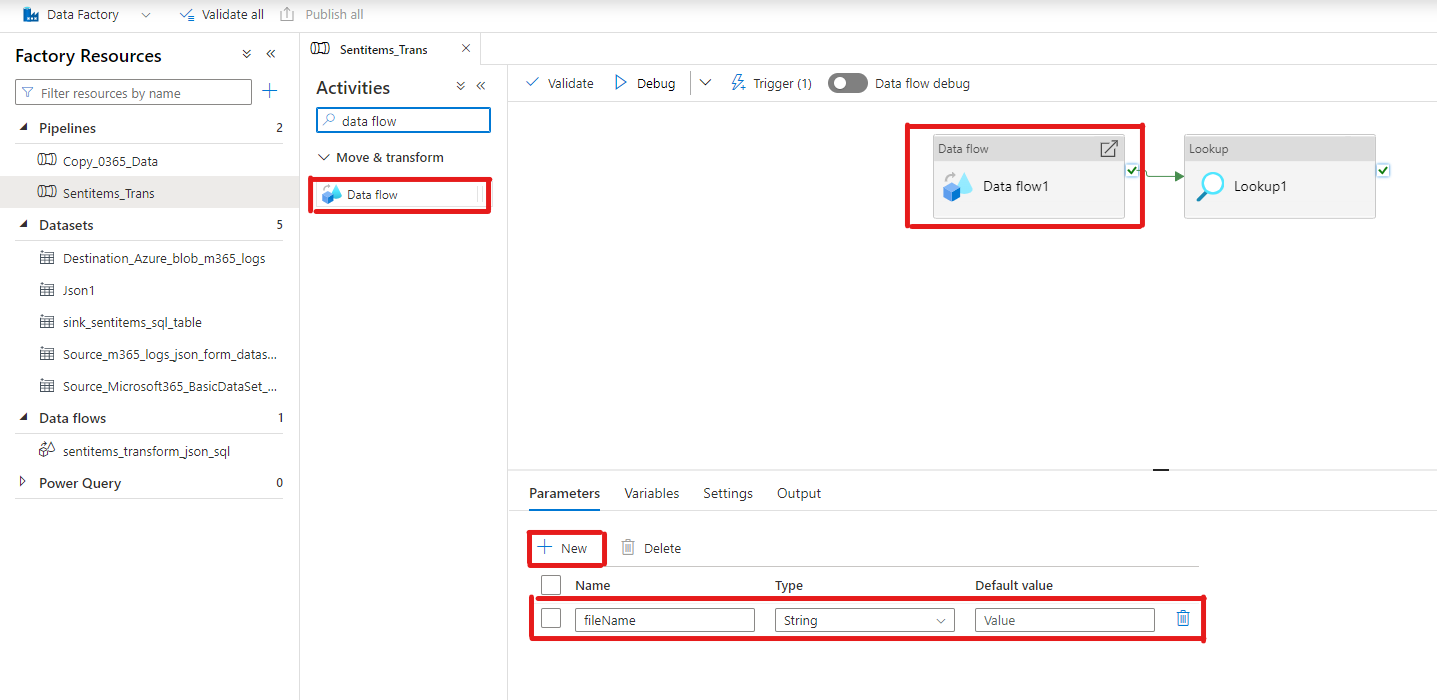


**Approve the Request:**

1. After a few mins into the execution of the pipeline, status changes from in progress to Consent Pending. This essentially means that we are now waiting for an approver is to go and grant the consent to the pipeline that can continue its execution.
2. In order to approve the consent anyone from the approval group of M365 admin center should do that.
3. Go to Microsoft 365 Admin center > Settings > Org Settings > Microsoft Graph Data Connect > Privileged Access Management Portal > Data Access Request > Approve.
4. Then we are going to see our data access request has been approved and the execution of the pipeline can continue. After a few mins we are going to see that our pipeline executed successfully.
5. To access exported files, Navigate to storage accounts > containers > m365-logs > sentitems.

**Create a Dataflow (Sentitems\_Trans):**

1. Go to azure Data factories and click on Author from the left navigation icons.
2. Click on the + symbol just below the factory resources, you will get more options, select pipeline.
3. From the activities drag and drop a **Data flow** activity into the canvas.
4. Now click on the Canvas & go to Parameters tab, click on +New -> Name: fileName; Type: String; Default Value: Value.

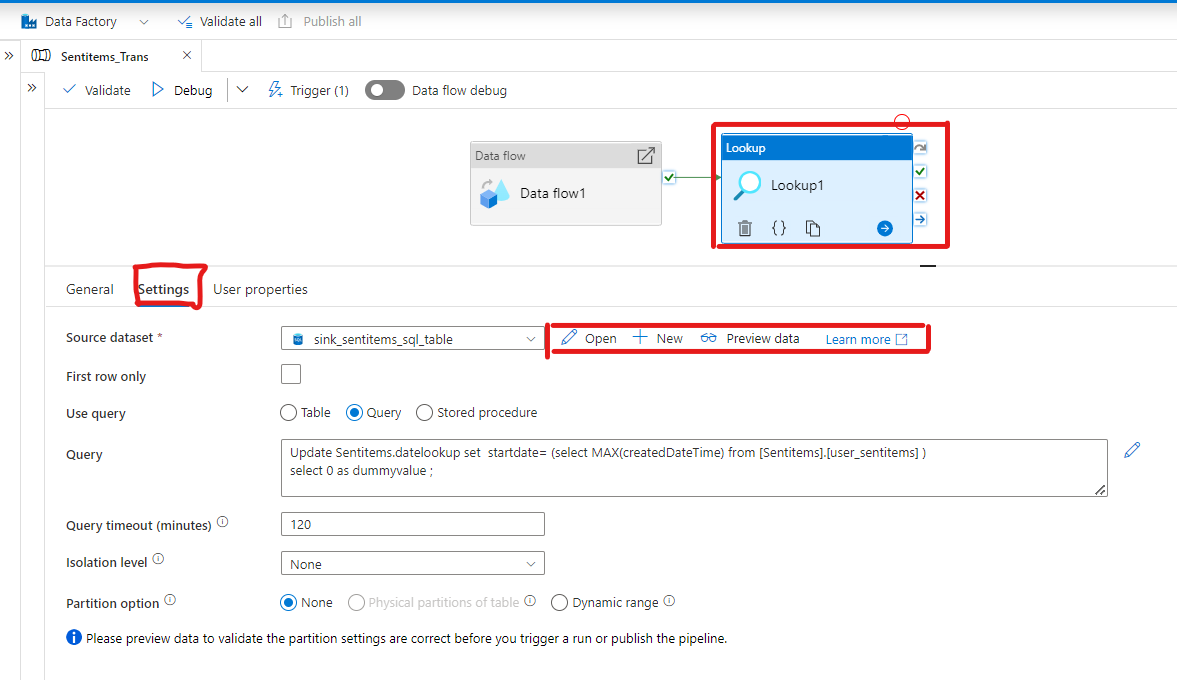


1. Click on the Data Flow activity & select source tab.
2. Create a new **source dataset** by clicking on the **+New** symbol.

Graphical user interface, application

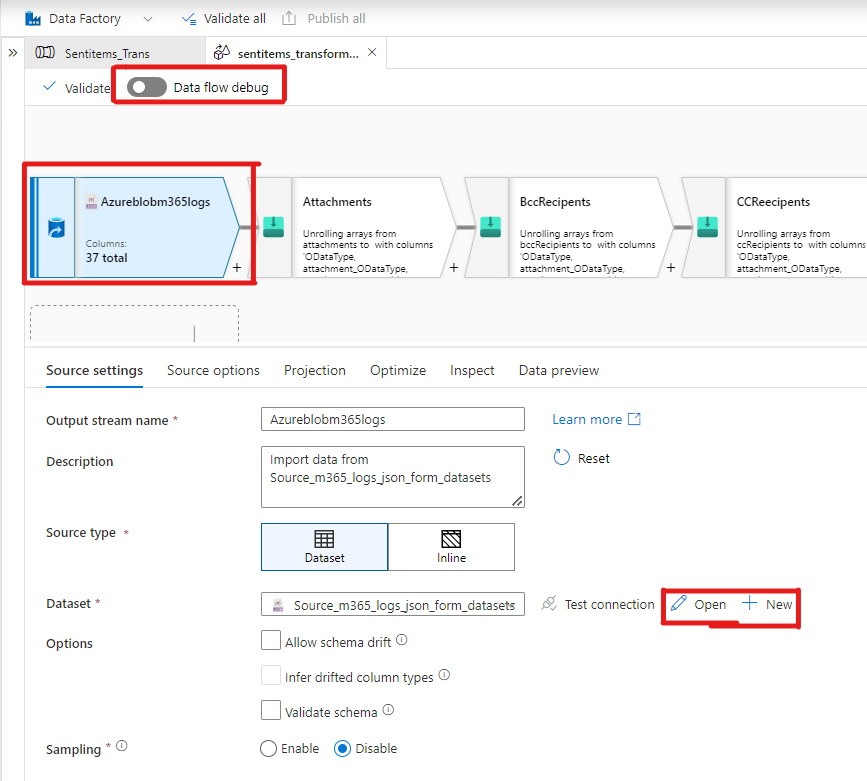
Description automatically generated

1. Click On the canvas and drag and drop a lookup activity onto the canvas.
2. Create a connection from Dataflow to lookup activity.
3. Click on the lookup activity and go to settings tab, click on the Source dataset. Create a new or open connection if you have an existing connection.
4. In our case we have already created an azure SQL database (**AzureSqlDatabase\_sql\_m365\_prod\_ne\_01**)
5. Now in the Lookup settings choose Use query as Query & In the query text box write the following SQL Query (**Update Sentitems.datelookup set startdate= (select MAX(createdDateTime) from [Sentitems].[user\_sentitems] )**
6. **select 0 as dummyvalue ;**).
7. Select query timeout 120 mins & Isolation level as None & partition option as none.



**Data Flows:**

1. To create a data flows, go to azure data factory > Author > Data flows
2. Turn on the Data flow Debug toggle button.
3. Right click on data flows > New data flow, click on the transformation > Source Settings tab, give a name to output stream name as **Azureblobm365logs** & Description.
4. Choose source type: dataset; sampling: disable.



Graphical user interface, application

Description automatically generated

1. Create a dataset by clicking on the **+New** Provide the necessary info like name, linked service and service principal, Integration runtime & choose file path, at the place of the file name write a function as “**@dataset().filename”,** comparison type: none, encoding: Default(UTF-8) **&** click on preview data to see the sample data.
2. Now go back to the transformation in the source settings tab you will see dataset as **source\_365logs\_json\_form\_datasets.** Click on Open the dataset in the connection tab, click on the test connection and go to Schema tab click on **import schema** to import the available fields and Parameters tab create a parameter by clicking on **+New,** Name: filename; Type: string; Default value: Value.

Graphical user interface, text, application, email

Description automatically generated

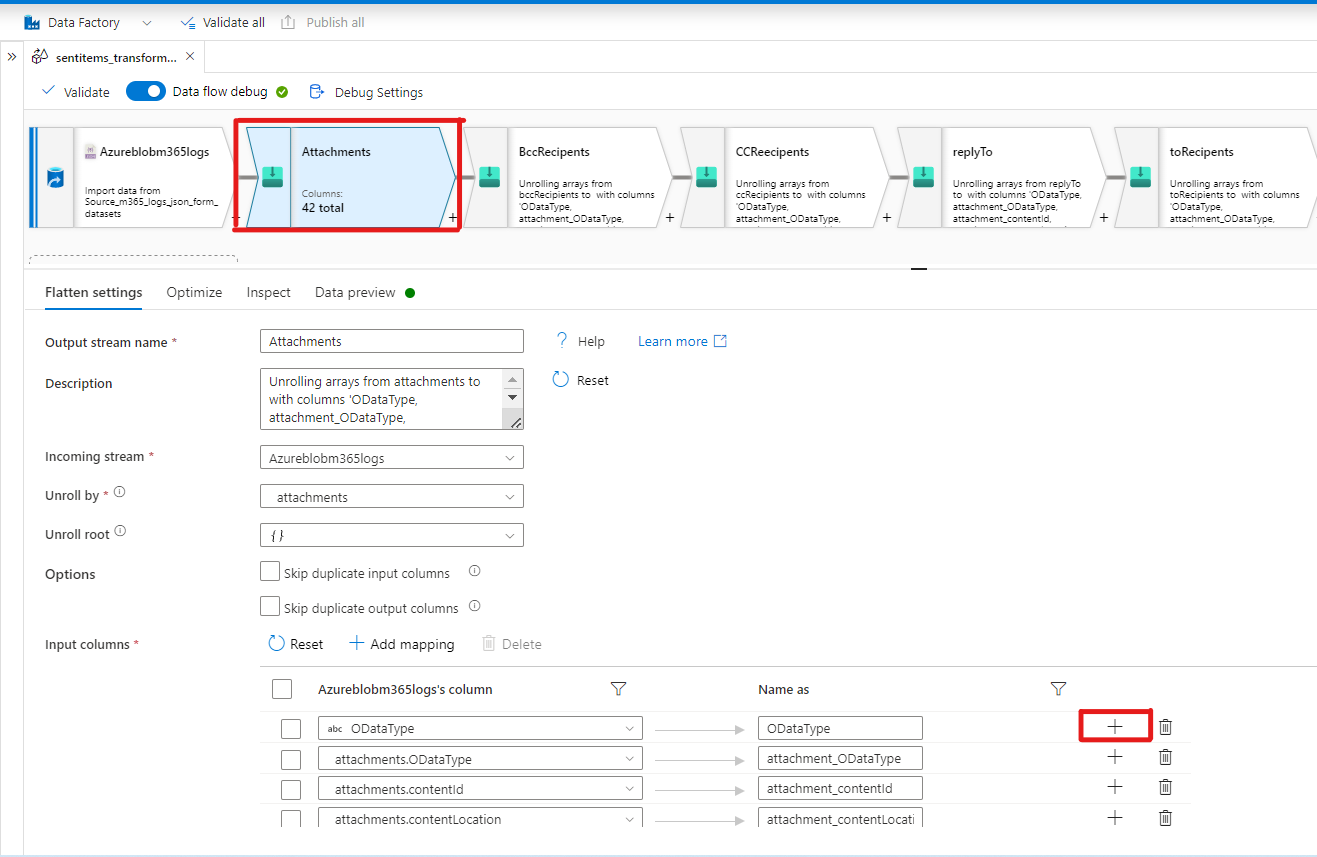
1. Now go back to transformation (**Azureblobm365logs**) >Source Options > JSON Settings > Document Per line.
2. Go to Projection tab click on Import Projection to import the schema.
3. Go to Optimize tab choose Partition Option as Use current Partitioning.
4. Go to Inspect Tab look for any array data type fields to massage the data in the next transformations.
5. Go to Data preview tab and check the data is available.

**Attachments:**

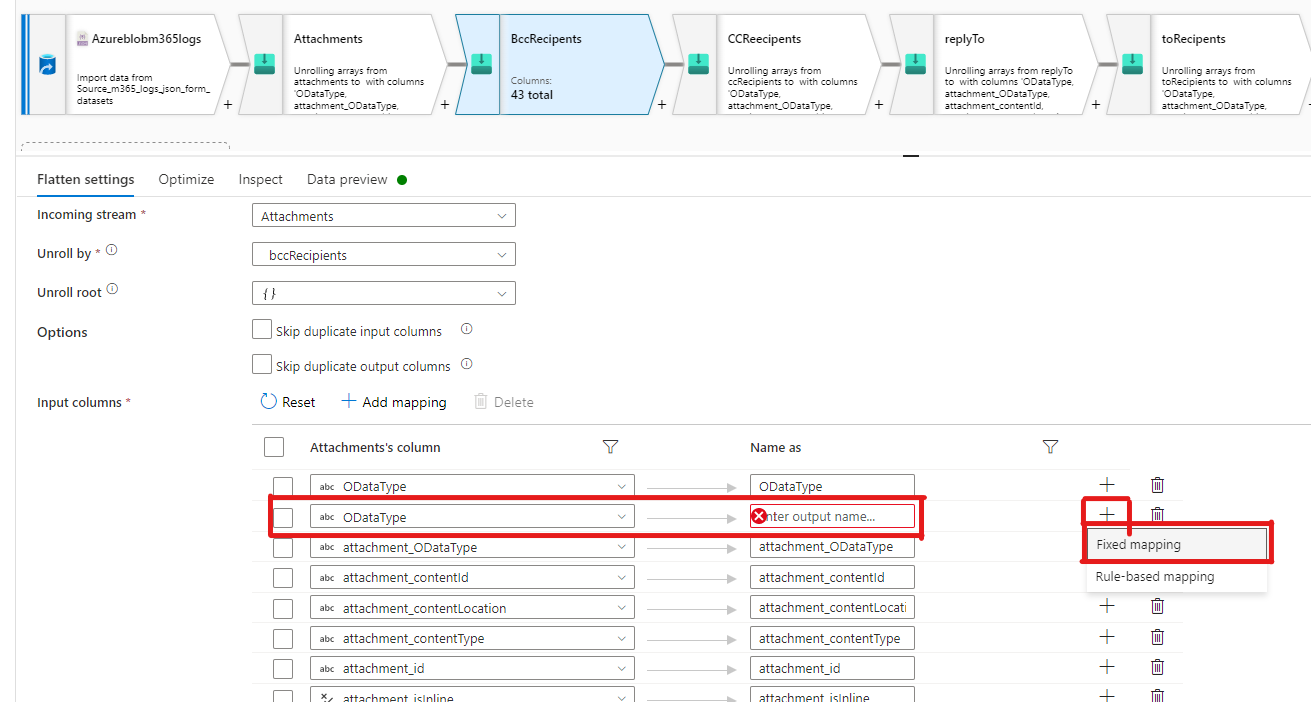
**Flatten Transformation:**

The flatten transformation takes array values inside hierarchical structures such as JSON and converts them into individual rows.

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **Azureblobm365logs** Transformation and choose **Flatten** Option You will get new Transformation connected to **Azureblobm365logs**.
2. Now go to the Flatten settings give a name (**Attachments**)
3. Choose incoming stream as **Azureblobm365logs**, unroll by as attachments & Unroll root as attachments [].



1. Under Input columns Add a new field by clicking on the Plus+ symbol choose fixed mapping as an option to create a new field. So, you will get two empty fields
2. Under Azureblobm365logs’s Columns choose the Attachment\_ODataType & in the other field choose the Name as Attachment\_ODataType. Add more fields like attachment\_contentId, attachment\_contentLocation, attachment\_contentType, attachment\_id, attachment\_isInline, attachment\_lastModifiedDateTime, attachment\_name, attachment\_size using the same procedure.



1. Go to inspect tab and check the fields those which are added in the Flatten settings tab. If not Go back and change them into string using above procedure.
2. Go to Data Preview tab, you will see the columns created & make sure the data is loaded.

**BCC Recipients:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **Attachments** Transformation and choose **Flatten** Option You will get new Transformation connected to **Attachments**.
2. Now go to the Flatten settings give a name (**BccRecipients**) & choose incoming stream as **Attachments**, unroll by as bccRecipients & Unroll root as bccRecipients [].
3. Under Attachment’s column, create two new fields (add fields according to the need) click on the plus (+) and click fixed mapping.
4. Under attachment’s columns in the created empty field choose bccRecipients.emailAddress.address and in the equivalent field choose the same name or bcc\_address and In the other created field choose bccRecipients.emailAddress.address under Attachment’s column & bcc\_name in the equivalent field.
5. Go to Inspect tab, check the fields and data types.
6. Now go to Data Preview, check for the data. You will see the bcc\_address and bcc\_name columns in the table.

**CC Recipients:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **bccReecipents** Transformation and choose **Flatten** Option You will get new Transformation connected to **bccReecipents**.
2. Now go to the Flatten settings give a name (**CCReecipents**) & choose incoming stream as **BCCReecipents**, unroll by as ccRecipients & Unroll root as {}.
3. Under Attachment’s column, create two new fields (add fields according to the need) click on the plus (+) and click fixed mapping.
4. Under attachment’s columns in the created empty field choose ccRecipients.emailAddress.address and in the equivalent field choose the same name or cc\_address and In the other created field choose ccRecipients.emailAddress.name under bccReecipents’s column & cc\_name in the equivalent field.
5. Go to Inspect tab, check the fields and data types.
6. Now go to Data Preview, check for the data. You will see the cc\_address and cc\_name columns in the table.

**replyTo:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **ccReecipents** Transformation and choose **Flatten** Option You will get new Transformation connected to **CCReecipents**.
2. Now go to the Flatten settings give a name (**replyTo**) & choose incoming stream as **CCReecipents**, unroll by as replyTo & Unroll root as {}.
3. Under **CCReecipents**‘s column, create two new fields (add fields according to the need) click on the plus (+) and click fixed mapping.
4. Under **CCReecipents**’s columns in the created empty field choose replyTo.emailAddress.address and in the equivalent field choose the same name or replyTo\_address and In the other created field choose replyTo.emailAddress.name under bccReecipents’s column & replyTo\_name in the equivalent field.
5. Go to Inspect tab, check the fields and data types.
6. Now go to Data Preview, check for the data. You will see the replyTo\_address and replyTo\_name columns in the table.

**toRecipents:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **replyTo** Transformation and choose **Flatten** Option You will get new Transformation connected to **replyTo**.
2. Now go to the Flatten settings give a name (**toRecipents**) & choose incoming stream as **replyTo**, unroll by as toRecipients & Unroll root as {}.
3. Under **replyTo**‘s column, create two new fields (add no of fields according to the need) click on the plus (+) and click fixed mapping.
4. Under **replyTo**’s columns in the created empty field choose toRecipents.emailAddress.address and in the equivalent field choose the same name or to\_address and In the other created field choose toRecipents.emailAddress.name under replyTo’s column & to\_name in the equivalent field.
5. Go to Inspect tab, check the fields and data types.
6. Now go to Data Preview, check for the data. You will see the to\_address and to\_name columns in the table.

**LoadTime:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **toReecipents** Transformation and choose **Derived Column** Option You will get new Transformation connected to **toRecipents**.

Graphical user interface, application, table, email

Description automatically generated

1. Now go to the **Derived Column** settings give a name (**LoadTime**) & choose incoming stream as **toRecipents**, unroll by as toRecipients & Unroll root as {}.
2. Under **columns**, add a new field (add no of fields according to the need) click on the plus (+) and click fixed mapping.
3. Under **column** in the created empty field, give it a name as Load Time & in the expression field provide the following ‘**currentTimestamp() + hours(3)**’ ( to get Bahrain Time zone)
4. Do the following

**Column - Expression**

LoadTime currentTimestamp() + hours(3)

update\_time currentTimestamp() + hours(3)

createdDateTime toTimestamp(createdDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")

sentDateTime toTimestamp(sentDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")

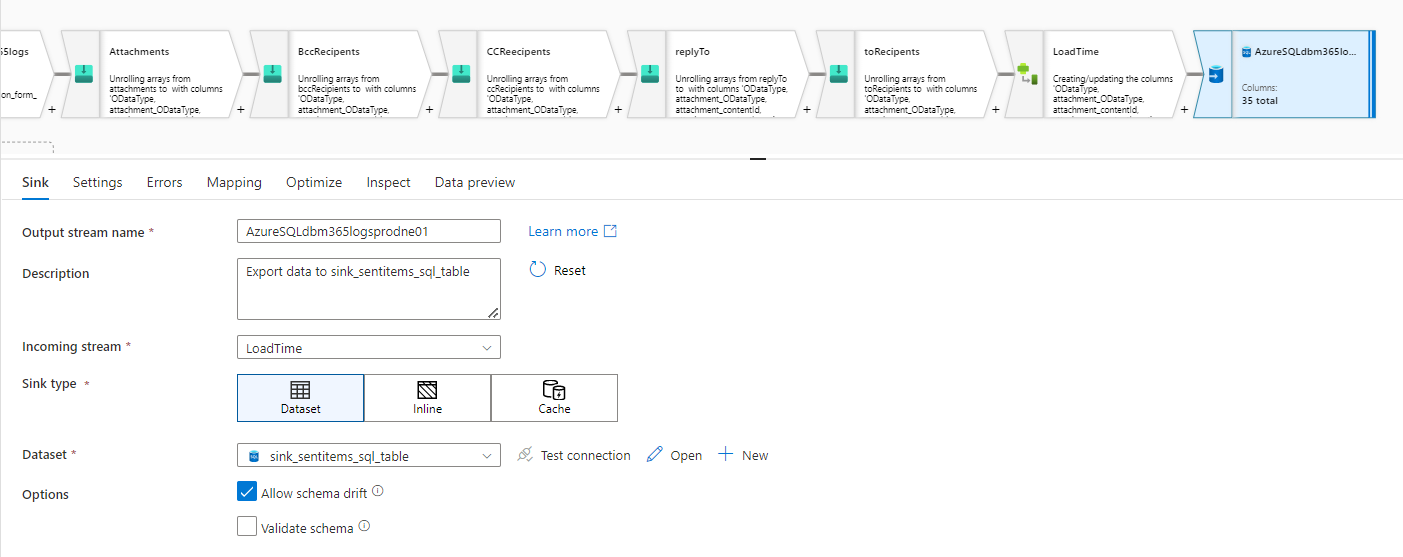
lastModifiedDateTime toTimestamp(lastModifiedDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")

receivedDateTime toTimestamp(receivedDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")

1. Go to Inspect tab, check the fields and data types.
2. Now go to Data Preview, check for the data. You will see the to\_address and to\_name columns in the table.

**AzureSQLdbm365logsprodne01:**

1. Now in the canvas click on the **Plus** **(+)** symbol which is located just below the **LoadTime** Transformation and choose **Sink** Option You will get new Transformation connected to **LoadTime**.
2. Now go to the sink tab give a name (**AzureSQLdbm365logsprodne01**), Description, Incoming Stream as LoadTime, sink type as Dataset, & options as Allow Schema Drift. dataset as sink\_sentitems\_sql\_table (click on new and add a linked service & give a table name as sentitems.user\_sentitems)

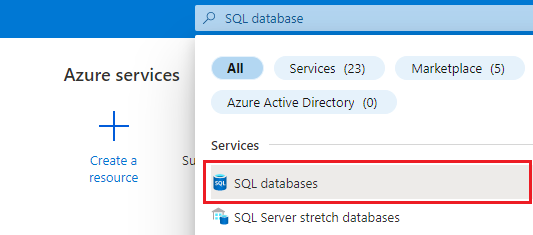


1. Go to settings tab check Update Method as Allow Insert, table action as None, check Use TempDB.
2. Go to mapping tab check the mapping names associated with columns.
3. Go to Optimize tab select Use current partitioning.
4. Go to Data Preview tab check the final output data.

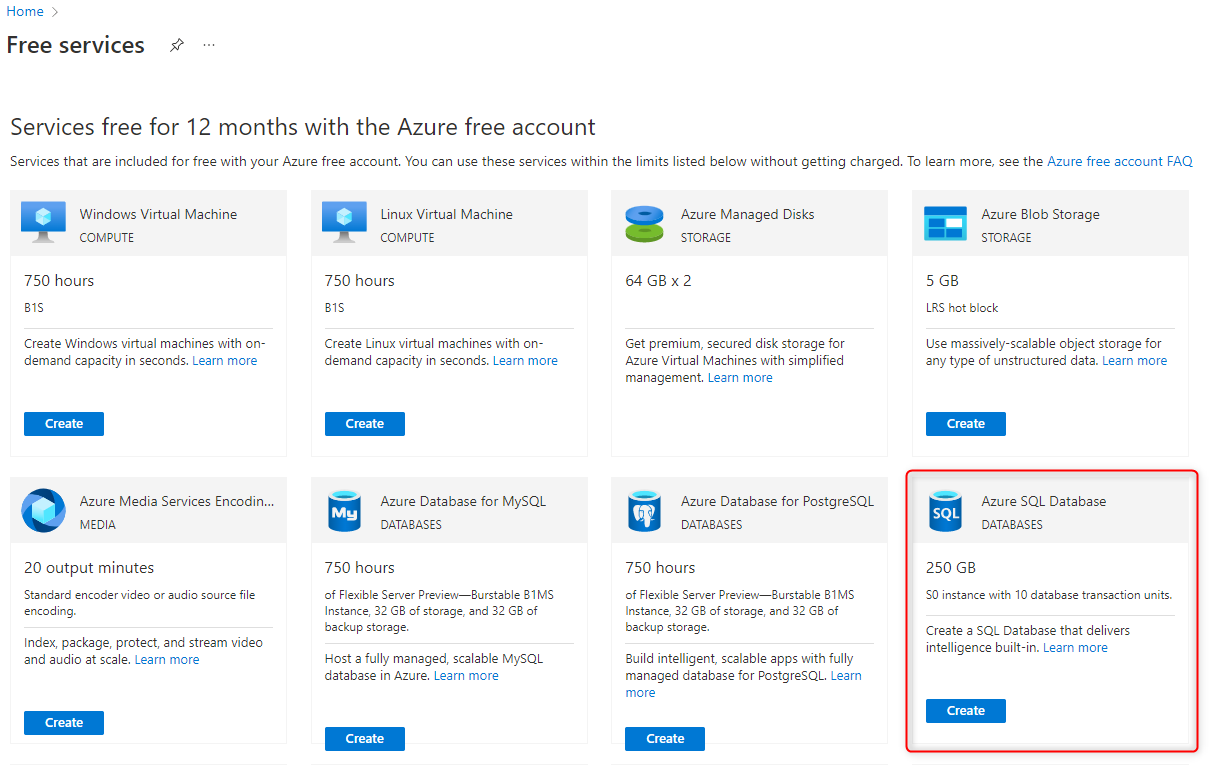
**Azure SQL Database:**

To create your database, follow these steps:

1. Sign in to the [Azure portal](https://portal.azure.com/) with your Azure free account.
2. Search for and select **SQL databases**:



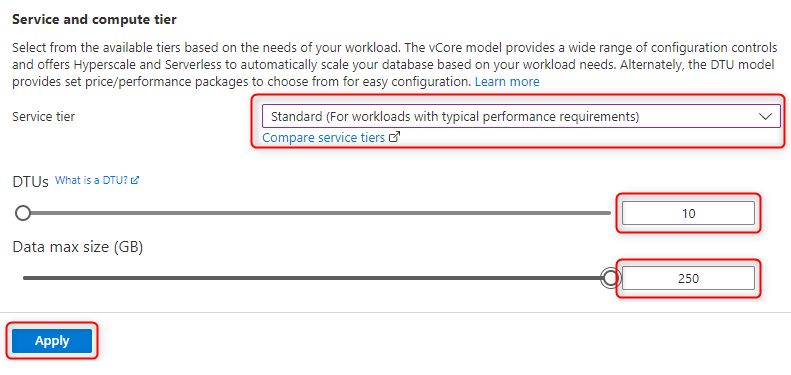
Alternatively, you can search for and navigate to **Free Services**, and then select the **Azure SQL Database** tile from the list:



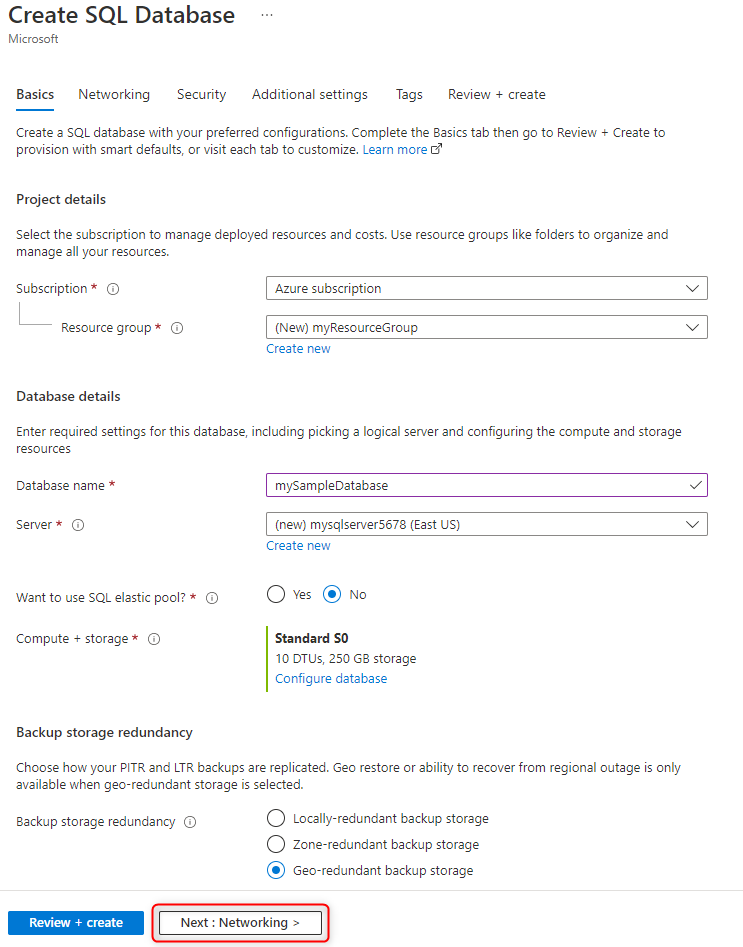
1. Select **Create**.
2. On the **Basics** tab of the **Create SQL Database** form, under **Project details**, select the free trial Azure Subscription, **Azure subscription 1**.
3. For **Resource group**, select **Create new**, enter myResourceGroup, and select **OK**.
4. For **Database name**, enter mySampleDatabase.
5. For **Server**, select **Create new**, and fill out the **New server** form with the following values:
   * **Server name**: Enter mysqlserver, and add some characters for uniqueness. We can't provide an exact server name to use because server names must be globally unique for all servers in Azure, not just unique within a subscription. So enter something like mysqlserver12345, and the portal lets you know if it's available or not.
   * **Server admin login**: Enter azureuser.
   * **Password**: Enter a password that meets complexity requirements, and enter it again in the **Confirm password** field.
   * **Location**: Select a location from the dropdown list.

Select **OK**.

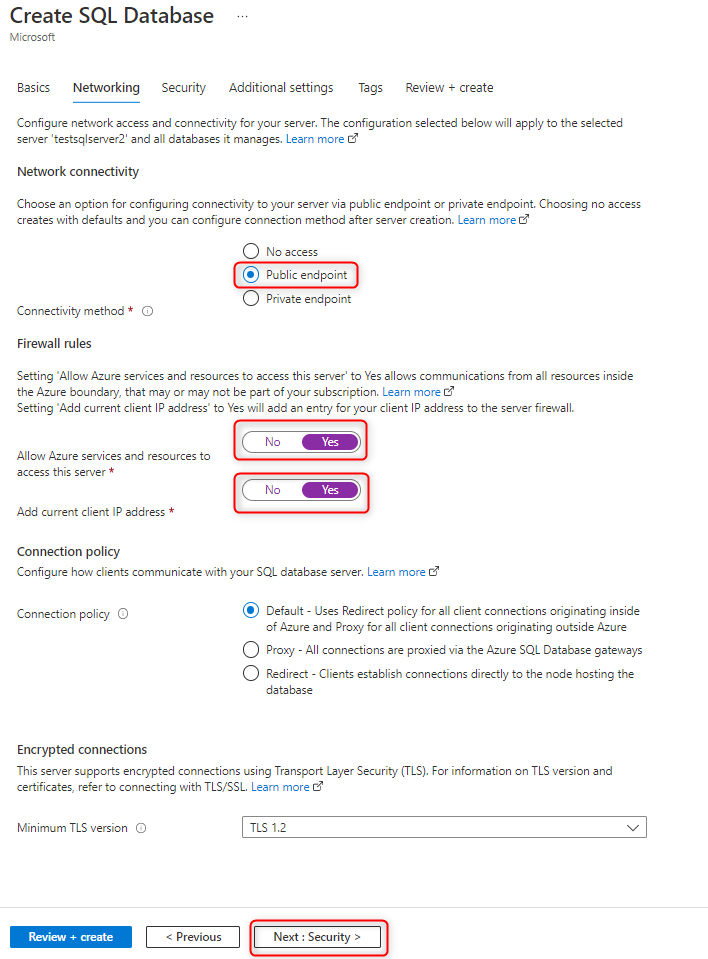
1. Leave **Want to use SQL elastic pool** set to **No**.
2. Under **Compute + storage**, select **Configure database**.
3. For the free trial, under **Service Tier** select **Standard (For workloads with typical performance requirements)**. Set **DTUs** to **10** and **Data max size (GB)** to **250**, and then select **Apply**.



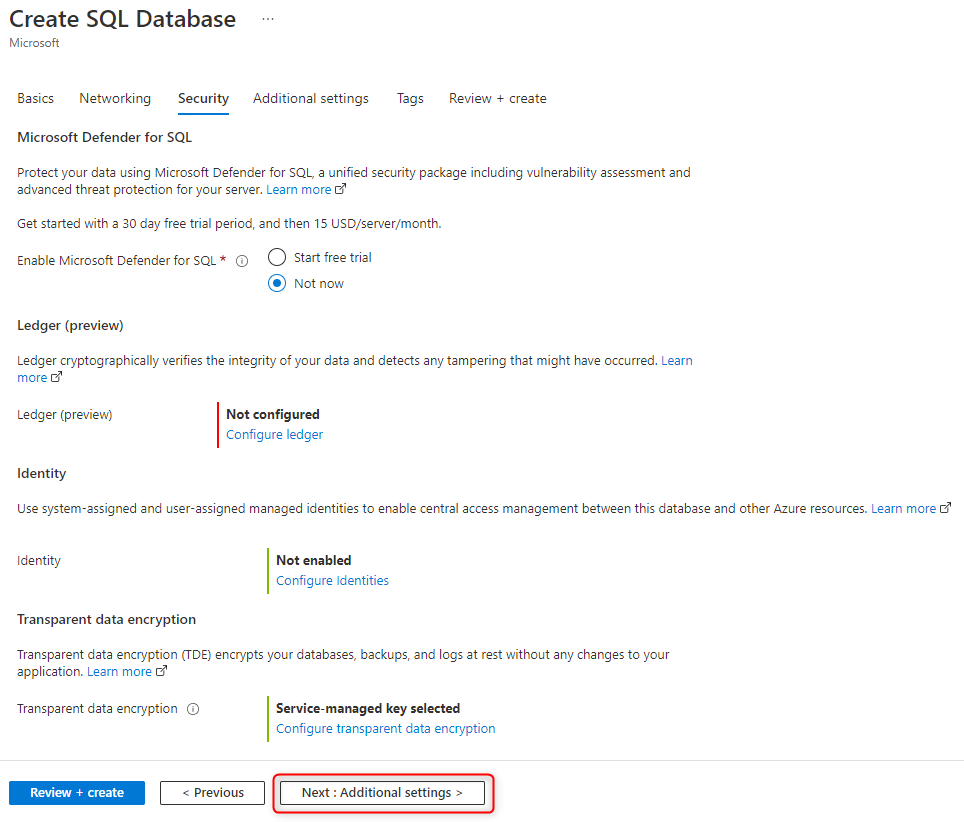
1. Leave **Backup storage redundancy** set to **Geo-redundant backup storage**
2. Select **Next: Networking** at the bottom of the page.



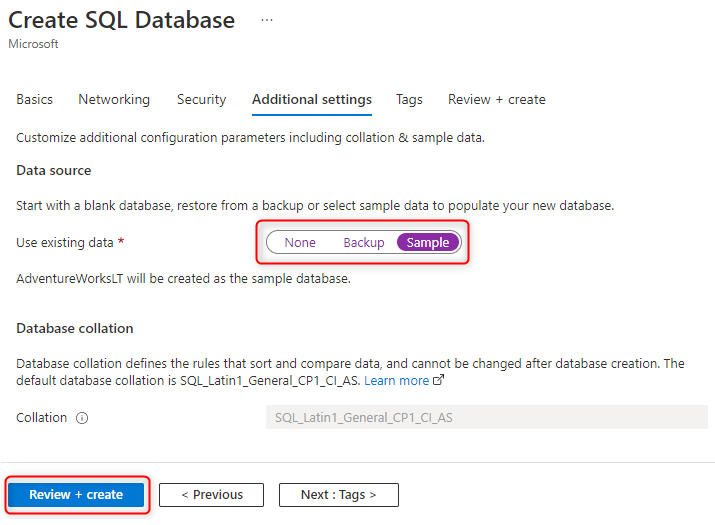
1. On the **Networking** tab, for **Connectivity method**, select **Public endpoint**.
2. For **Firewall rules**, set **Allow Azure services and resources to access this server** set to **Yes** and set **Add current client IP address** to **Yes**.
3. Leave **Connection policy** set to **Default**.
4. For **Encrypted Connections**, leave **Minimum TLS version** set to **TLS 1.2**.
5. Select **Next: Security** at the bottom of the page.



1. Leave the values unchanged on **Security** tab.



1. Select **Review + create** at the bottom of the page.



1. On the **Review + create** page, after reviewing, select **Create**.

**Create a schema:**

**Create schema sentitems;**

**Grant create table to dfm365pazne01**

**select top 10 \* from [Sentitems].[user\_sentitems]**

Now go to the Azure SQL databases create Two tables with in it. one as **Sentitems.user\_items** and the other as **Sentitems.datelookup**

**Sentitems.user\_items:**

**/\*\*\*\*\*\* Object: Table [Sentitems].[user\_sentitems] Script Date: 11/16/2022 10:59:29 PM \*\*\*\*\*\*/**

**SET ANSI\_NULLS ON**

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

**CREATE TABLE [Sentitems].[user\_sentitems](**

**[attachment\_ODataType] [nvarchar](max) NULL,**

**[attachment\_contentId] [nvarchar](max) NULL,**

**[attachment\_contentLocation] [nvarchar](max) NULL,**

**[attachment\_contentType] [nvarchar](max) NULL,**

**[attachment\_id] [nvarchar](max) NULL,**

**[attachment\_isInline] [bit] NULL,**

**[attachment\_lastModifiedDateTime] [datetime2](7) NULL,**

**[attachment\_name] [nvarchar](max) NULL,**

**[attachment\_size] [int] NULL,**

**[bcc\_address] [nvarchar](max) NULL,**

**[bcc\_name] [nvarchar](max) NULL,**

**[cc\_address] [nvarchar](max) NULL,**

**[cc\_name] [nvarchar](max) NULL,**

**[createdDateTime] [datetime2](7) NULL,**

**[flagStatus] [nvarchar](max) NULL,**

**[from\_emailaddress] [nvarchar](max) NULL,**

**[from\_name] [nvarchar](max) NULL,**

**[hasAttachments] [bit] NULL,**

**[id] [nvarchar](max) NULL,**

**[internetMessageId] [nvarchar](max) NULL,**

**[lastModifiedDateTime] [datetime2](7) NULL,**

**[ptenant] [nvarchar](max) NULL,**

**[puser] [nvarchar](max) NULL,**

**[receivedDateTime] [datetime2](7) NULL,**

**[repltTo\_address] [nvarchar](max) NULL,**

**[replyTo\_name] [nvarchar](max) NULL,**

**[sender\_emailaddress] [nvarchar](max) NULL,**

**[sender\_name] [nvarchar](max) NULL,**

**[sentDateTime] [datetime2](7) NULL,**

**[subject] [nvarchar](max) NULL,**

**[to\_address] [nvarchar](max) NULL,**

**[to\_name] [nvarchar](max) NULL,**

**[load\_time] [datetime2](7) NULL,**

**[update\_time] [datetime2](7) NULL,**

**[src\_file] [nvarchar](max) NULL**

**) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]**

**GO**

**Sentitems.datelookup:**

**select MAX(createdDateTime) as startdate into sentitems.datelookup from [Sentitems].[user\_sentitems]**

Graphical user interface, application

Description automatically generated

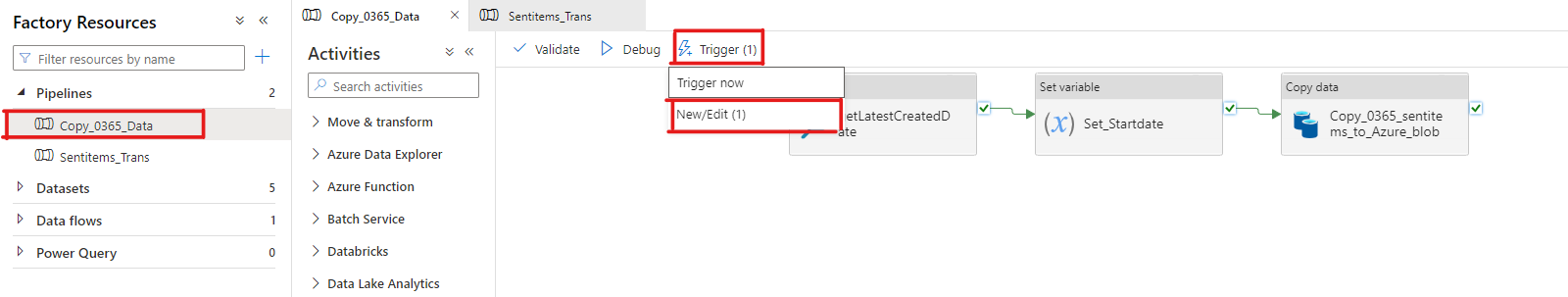
The table **Sentitems.user\_items** was connected in the data flow so that we can get the data in the table from container. Once the data comes to the table the Sentitems.datelookup will be updated with latest created date Sentitems.datelookup table will help us to load newly imported data from M365 into containers.

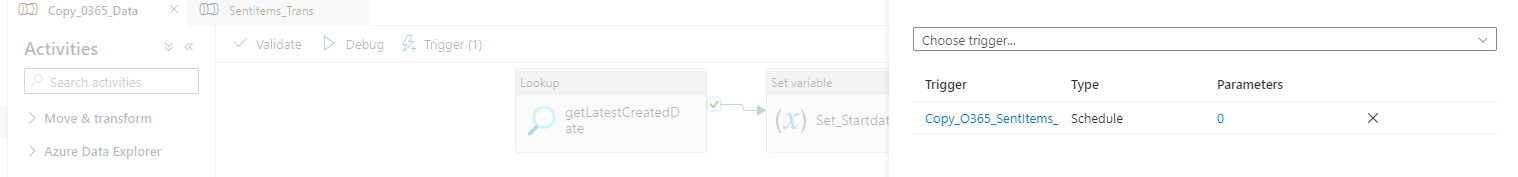
Now connect the Azure SQL database to our Power Bi report (Use Azure credentials) and implement the insights as per requirement.

**Trigger Events (for Copy\_0365\_Data) :**

To create a trigger event for **Copy\_0365\_Data** to run and copy the data from M365 to Azure blob storage.

1. go to the Azure data factories> pipelines> select Copy\_0365\_Data.
2. In the canvas select on the transform and choose the Trigger option as New/Edit.
3. Create a new trigger event and name it as a Copy\_O365\_Sentitems.
4. Enter the information of Start date, time zone, Recurrence and status as started.

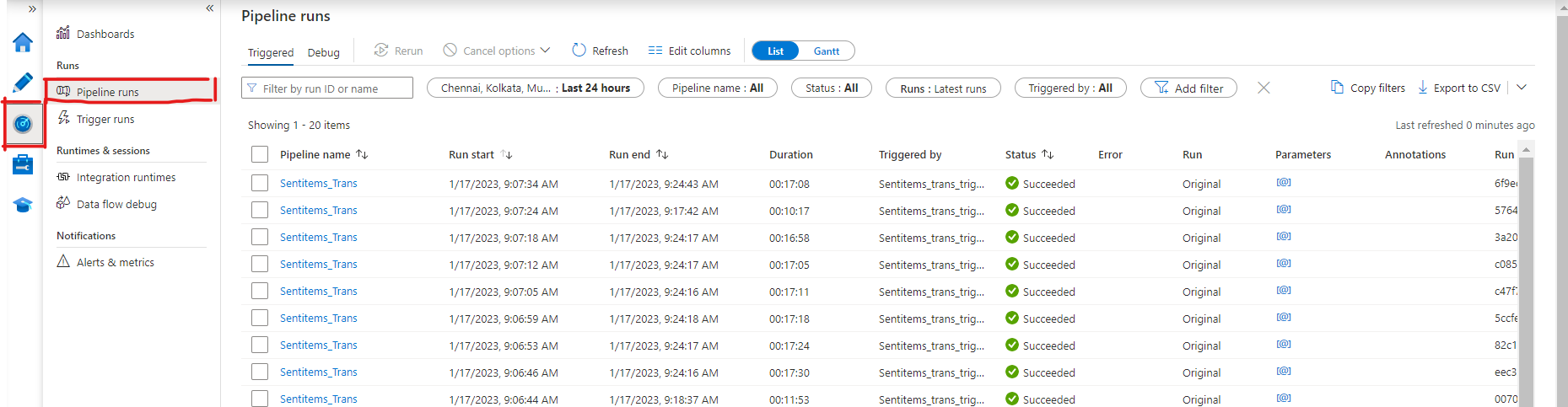




Graphical user interface, application

Description automatically generated

1. Once the trigger is set up. It will start working as per schedule.
2. You can view them in the monitor tab. Go to monitor tab & click on pipeline runs.



**Trigger Event (For Sentitems\_Trans):**

To create a trigger event for **Copy\_0365\_Data** to run and copy the data from M365 to Azure blob storage.

1. go to the Azure data factories> pipelines> select Sentitems\_Trans.
2. In the canvas select on the transform and choose the Trigger option as New/Edit.
3. Create a new trigger event and name it as a Copy\_O365\_Sentitems.
4. Enter the information of Subscription, Storage account, Container name & event as Blob created.

Graphical user interface, application

Description automatically generated

Graphical user interface, application, Teams

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. Once the trigger is set up. It will trigger the run job whenever a file created into the given container.
2. You can view them in the monitor tab. Go to monitor tab & click on pipeline runs.

Graphical user interface, application, Excel

Description automatically generated

**Pipeline:**

**User-details:**

1. Create a pipeline to copy the User details from M365 Admin center to Azure blob storage.
2. create a pipeline by clicking on the New Pipeline of Pipeline side menu.

Graphical user interface, text, application

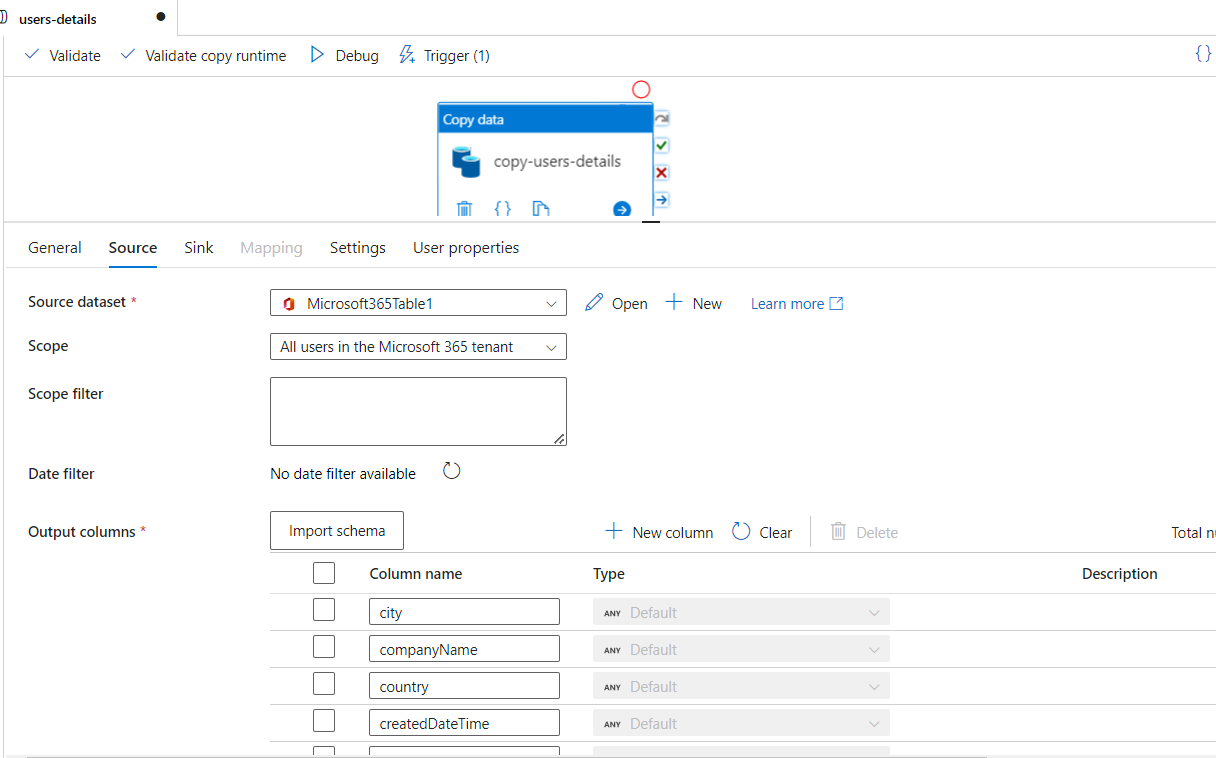
Description automatically generated

1. Provide a Name(user-details) and Description of the Pipeline.
2. Drag and drop a **copy data** activity into the canvas.
3. Click on the pipeline select **Source** tab select Source dataset as Microsoft365Table1 That has already created previously or else create new one according to the requirement. Open that Source dataset and choose BasicDtaaset\_v0.User\_v1 & go back to the **Source** tab.

Graphical user interface, text, email

Description automatically generated

1. In the Source tab click on import schema to import the schema from the selected dataset. You will see the column names remove sensitive information like the Personal details.

****

1. Go to **Sink** tab choose the **Sink dataset** create a new by clicking on + symbol. Select Azure Blob Storage and provide storage details of storage account. Note you should have create a folder in the storage container to create linked service and sink dataset. Once you create a linked service provide the folder path of the file location.

**Graphical user interface, text

Description automatically generated**

1. Go back to sink dataset check the details of the dataset. Once that is doe go to Use properties and click on Auto Generate Option to get the source and Destination (sink) connection details.

Graphical user interface, text, application, email

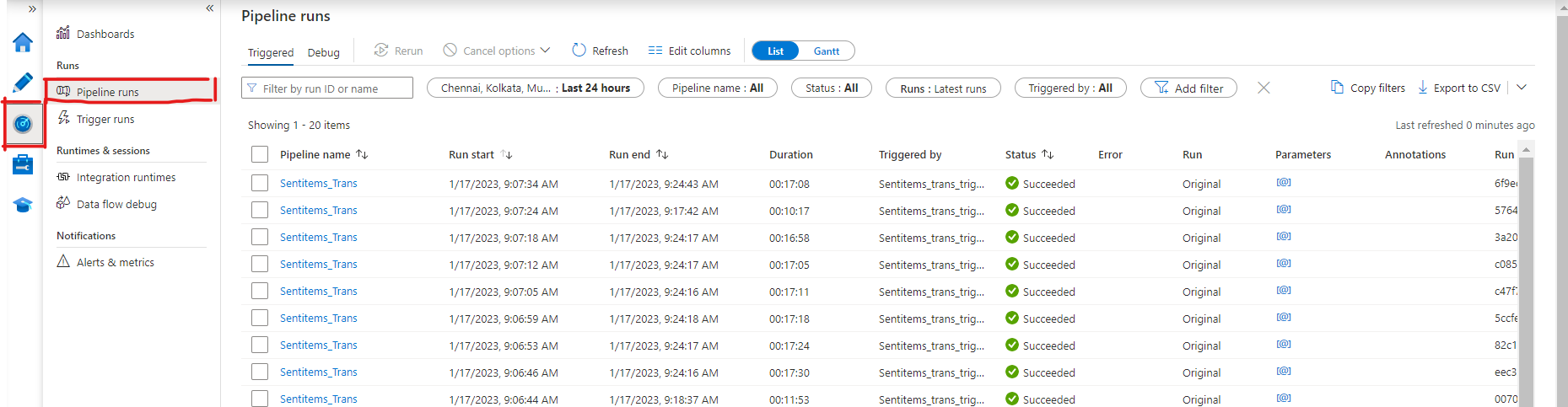
Description automatically generated

1. Now select the Copy data and click on trigger> Trigger now.

Graphical user interface, text, application

Description automatically generated

1. Now go to Monitor tab and select the Pipeline Runs. There you will see a pipeline run start.
2. This pipeline status will be under Process until and unless it gets access from M365 Admin Center. Ask Admin team to approve the access. This is onetime approve. Once they Approved the request the pipeline copy the data into the given container/folder/Blob storage.



**Pipeline:**

**Userdetails\_Transformation:**

1. Create a Pipeline and provide name (Userdetails\_Transformation) and Description (Optional)
2. Drag and drop a Data Flow from the copy activity into canvas. Click on the Data flow.
3. Click anywhere in open space of canvas & click on Parameters then create a New Parameter by clicking on **+New,** Choose Name: filename; Type: String; Default value: Value.

Graphical user interface, text, application, email

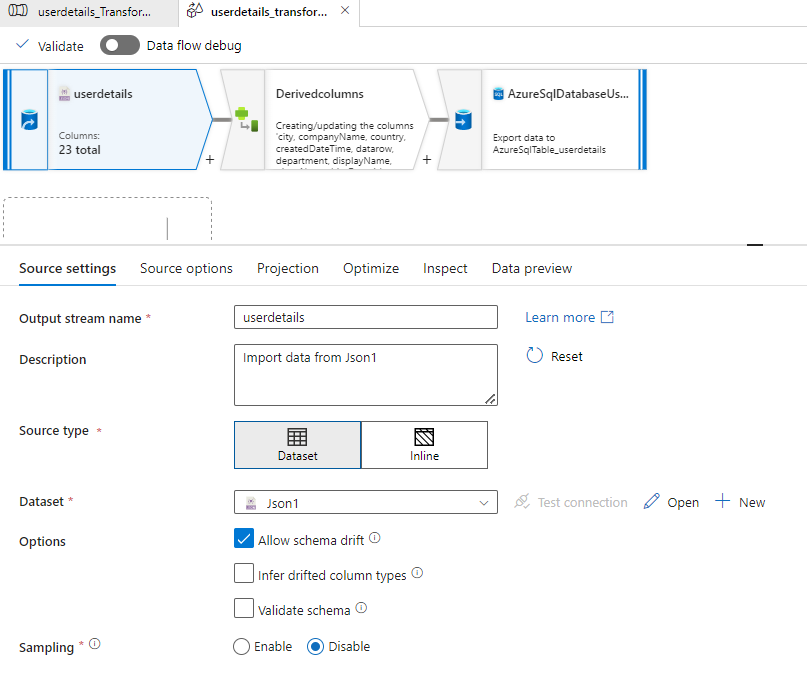
Description automatically generated

1. Click on the Data Flow and select **Settings** tab create a new Data Flow click on **+New** Symbol
2. Go to **Source Settings** tab & Give a name, description, **Source Type** as Dataset, Dataset as JSON, Options as Allow schema drift.

Graphical user interface, text, application, email

Description automatically generated

1. Go to source Options tab, **Column to store file Name** as filename, Under **JSON settings** > Document from as Document per line.



Graphical user interface, text, application, email

Description automatically generated

1. Go to Projection Tab click on import Projection or if you think the default format looks good proceed with that or else you can use overwrite schema to modify the datatype.

Graphical user interface, text, application, email

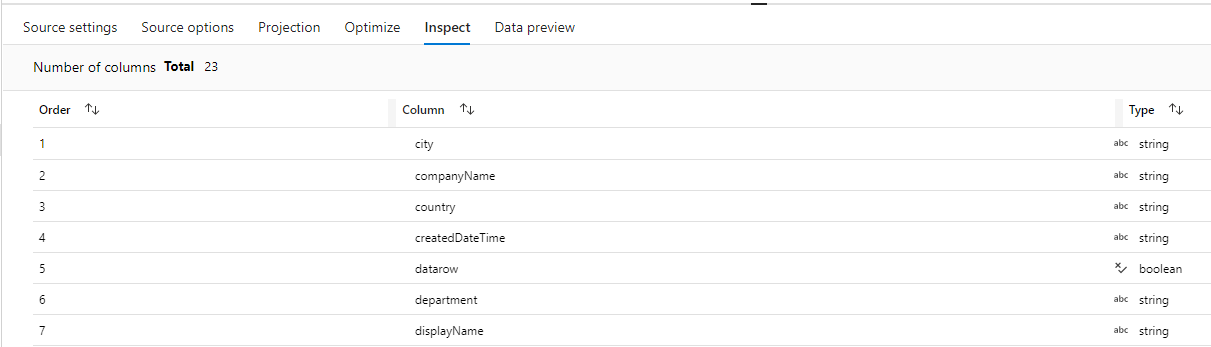
Description automatically generated

1. Go to Optimize tab and then **Use current partitioning.**

Graphical user interface, text, application, Word

Description automatically generated

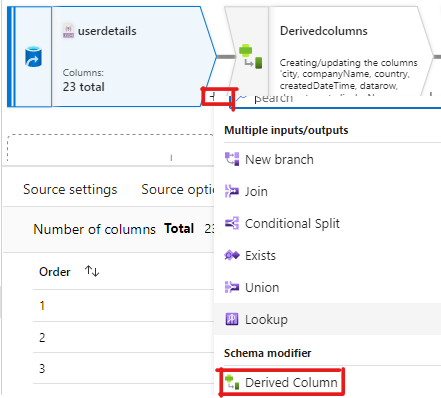
1. Go to Inspect and cross verify the columns and their data types.



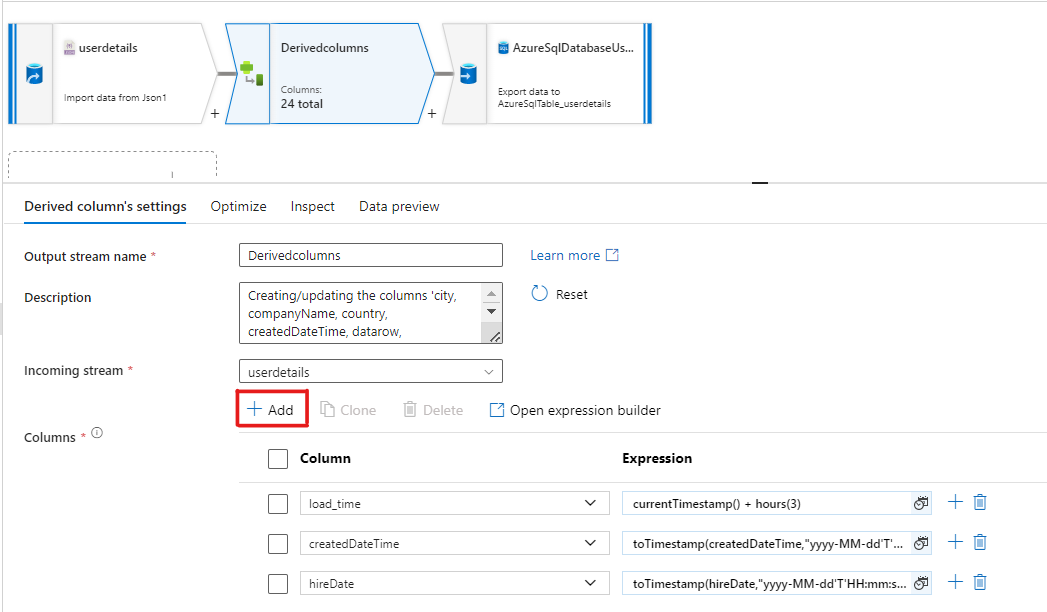
1. Check the data preview. If you are getting the Data it means the transformation connection is established.

**Derived Columns:**

1. Click on the + symbol and search for Derived columns option.



1. Click on the Derived columns and go to **derived Column’s settings** tab click on the +Add button to add a derived column.



1. Add createdDateTime: toTimestamp(createdDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")
2. Add hireDate: toTimestamp(createdDateTime,"yyyy-MM-dd'T'HH:mm:ss'Z'")
3. Add Load\_time: currentTimestamp() + hours(3)
4. Now Go to Inspect tab and check for Date time data types for respective columns.

Graphical user interface, application

Description automatically generated

1. Check the data preview for data.

**Azure SQL Database User details:**

1. Create a sink by clicking on + symbol followed by derived columns in the canvas.
2. Go to **sink** tab give name and description, choose **sink type** as Dataset & Options as Allow schema drift.
3. **Dataset** create a database by clicking **+ New** symbolchoose Azure SQL databases & provide name, linked service, table name and schema details. Save the settings.
4. Open the dataset that we have created and check the table and name.
5. Now go to settings tab allow insert for Update method, Table action: None, Use TempDB: check.
6. Once you create the table go to the ssms or any database editor app create a the following columns

**/\*\*\*\*\*\* Object: Table [Sentitems].[user\_sentitems] Script Date: 11/16/2022 10:59:29 PM \*\*\*\*\*\*/**

**SET ANSI\_NULLS ON**

**GO**

**SET QUOTED\_IDENTIFIER ON**

**GO**

**CREATE TABLE [Sentitems].[userdetails](**

**[Name] [nvarchar](max) NULL,**

**[PrimarySmtpAddress] [nvarchar](max) NULL,**

**[DisplayName] [nvarchar](max) NULL,**

**[Company] [nvarchar](max) NULL,**

**[Department] [nvarchar](max) NULL,**

**[Office] [nvarchar](max) NULL,**

**[Title] [nvarchar](max) NULL,**

**[UserPrincipalName] [nvarchar](max) NULL,**

**[puser] [nvarchar](max) NULL,**

**[ptenant] [nvarchar](max) NULL,**

**[load\_time] [datetime2](7) NULL,**

**) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]**

**GO**

1. Create one more table for datelookup

**select MAX(createdDateTime) as startdate into sentitems.userdetails\_datelookup from [Sentitems].[user\_sentitems]**

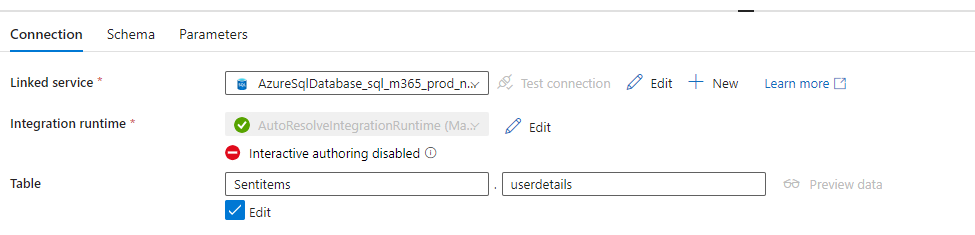
1. Go to Mapping and check the mapping output. Add mapping if you think the column mapping is not correct.
2. Go to **Data Preview** to check the final output.

**Userdetails\_Transfromation:**

1. No Coming back to Userdetials\_transformation pipeline. Till now we have set up a data flow.
2. Now trigger the pipeline to import the data from storage to our SQL database.
3. Go to trigger and trigger now. So that we can get the data into our database. Go to the databases and check for the data.
4. Now go back to our userdetails\_transformation pipeline in the settings tab we can see the dataflow details. Till this is a manual process to automate this process we need to apply trigger events. To do so add a parameter in the settings tab. We have already created a parameter in the canvas if not go and create a new one.
5. Click on the empty canvas and go to Parameters tab create a +New parameter Name: filename Type: String, Default value: value.
6. Now go to Data flow settings tab and add the parameter Name: fileName, Value: @pipeline().paramaters.pipeline.

Graphical user interface, text, application, email

Description automatically generated



Graphical user interface, text, application, email

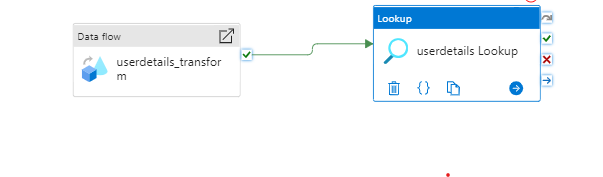
Description automatically generated

Graphical user interface, text, application, email

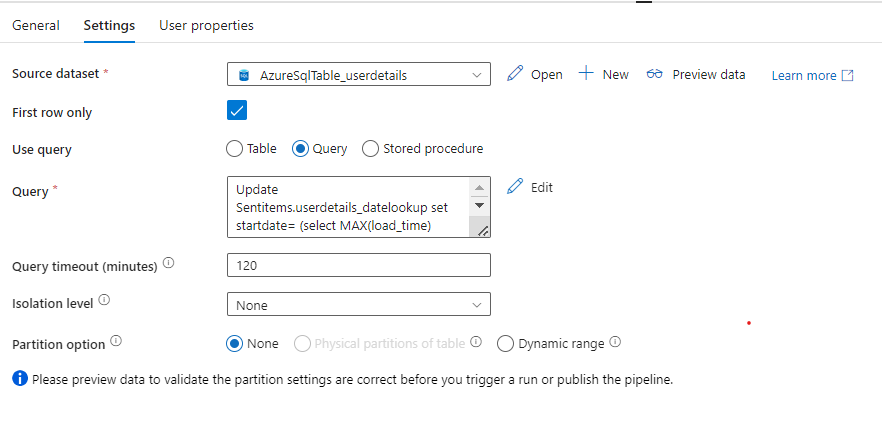
Description automatically generated

**User details Lookup:** To automate the triggering and to only access the newly imported data into storage account we are using look up activity.

1. Search a Lookup Activity from activities & drag into canvas.
2. Connect the Lookup activity from Dataflow activity in the canvas



1. Click on the look up activity, go to settings tab connect sink that we have already created in data flow.
2. Select User query as query & write the query in the given box below it. “Update Sentitems.userdetails\_datelookup set startdate= (select MAX(load\_time) from [Sentitems].[userdetails] ) select 0 as dummyvalue ;”
3. Query Time Out= 120 & Partition Option None.



Dataset:

|  |  |  |
| --- | --- | --- |
|  | **MESSAGES** | |
| **S.NO** | **Output-Parameter-Name** | **Description** |
| **1** | **bccRecipients** | The Bcc: recipients for the message.(Email-address) |
| **2** | **bodyPreview** | The first 255 characters of the message body. It is in text format. |
| **3** | **categories** | The categories associated with the message defined by the indvidual User |
| **4** | **ccRecipients** | The Cc: recipients for the message.(Email-address) |
| **5** | **conversationId** | The ID of the conversation the email belongs to. |
| **6** | **conversationIndex** | Indicates the position of the message within the conversation. |
| **7** | **createdDateTime** | The date and time the message was created. The date and time information uses ISO 8601 format and is always in UTC time. |
| **8** | **hasAttachments** | Indicates whether the message has attachments. This property doesn't include inline attachments,(Example : attachments in Body of email like Snippets,JPEG pictures…) |
| **9** | **id** | Unique identifier for the message |
| **10** | **importance** | The importance of the message. The possible values are: low, normal, and high. |
| **11** | **inferenceClassification** | The classification of the message for the user, based on inferred relevance or importance, or on an explicit override. The possible values are: focused or other. |
| **12** | **internetMessageHeaders** | A collection of message headers defined by RFC5322. The set includes message headers indicating the network path taken by a message from the sender to the recipient. I |
| **13** | **internetMessageId** | [The message ID in the format specified by RFC2822.](https://www.ietf.org/rfc/rfc2822.txt) |
| **14** | **isDeliveryReceiptRequested** | Indicates whether a read receipt is requested for the message. |
| **15** | **isDraft** | Indicates whether the message is a draft. |
| **16** | **isRead** | Indicates whether the message has been read |
| **17** | **isReadReceiptRequested** | Indicates whether a read receipt is requested for the message. |
| **18** | **lastModifiedDateTime** | The date and time the message was last changed. The date and time information uses ISO 8601 format and is always in UTC time |
| **19** | **parentFolderId** | The unique identifier for the message's parent mailFolder. |
| **20** | **receivedDateTime** | The date and time the message was last received. The date and time information uses ISO 8601 format and is always in UTC time |
| **21** | **replyTo** | The email addresses to use when replying. |
| **22** | **sentDateTime** | The date and time the message was last sent. The date and time information uses ISO 8601 format and is always in UTC time |
| **23** | **subject** | The subject of the message. |
| **24** | **toRecipients** | The To: recipients for the message.(Email-Address) |
| **25** | **webLink** | The URL to open the message in Outlook on the web. |
| **26** | **extensions** | The collection of open extensions defined for the message. Nullable. |
| **27** | **multiValueExtendedProperties** | The collection of multi-value extended properties defined for the message. Nullable. |
| **28** | **singleValueExtendedProperties** | The collection of single-value extended properties defined for the message. Nullable. |
| **29** | **body** | The body of the message. It can be in HTML or text format |
| **30** | **flag** | The flag value that indicates the status, start date, due date, or completion date for the message. |
| **31** | **from** | The owner of the mailbox from which the message is sent. In most cases, this value is the same as the Sender property |
| **32** | **sender** | The account that is actually used to generate the message. In most cases, this value is the same as the from property |
| **33** | **uniqueBody** | The part of the body of the message that is unique to the current message. |
| **34** | **contentType** | The MIME type.Standard Email format |
| **35** | **contentId** | ContentId can be used as a custom way to identify an attachment in order to reference it from within the body of the item t |
| **36** | **lastModifiedDateTime** | DateTimeOffset the message was modified |
| **37** | ***domain name*** | will extarct from Torecipents and create custom column in Report |
| **38** | **Attachment Fields** | **Below are the Attachment Fields that are available for Given attachment** |
| **39** | ***attachment-id*** | Attachment ID |
| **40** | ***attachment-isInline*** | true if the attachment is inline if not false (Example : attachments in Body of email like Snippets,JPEG pictures…) |
| **41** | ***attachment-name*** | The attachment's file name related to Email |
| **42** | ***attachment-size*** | The length of the attachment in bytes. |
| **43** | ***attachment-type*** | The attachment content type like spreadsheet , image JPEG |
|  |  |  |
| **S.NO** | **USER DETAILS** | |
| **1** | **country** | The country or region in which the user is located; for example, "US" or "UK". |
| **2** | **city** | The city in which the user is located. |
| **3** | **officeLocation** | The office location in the user's place of business. |
| **4** | **jobTitle** | The user’s job title. |
| **5** | **mail** | The SMTP address for the user. |
| **6** | **companyName** | The company name which the user is associated. |
| **7** | **department** | The name for the department in which the user works. |
| **8** | **displayName** | The name displayed in the address book for the user. This is usually the combination of the user's first name, middle initial and last name. |
| **9** | **givenName** | The given name (first name) of the user. |
| **10** | **Surname** | The Surname name of the user. |
| **11** | **id** | The unique identifier for the user. Inherited from directoryObject. |
| **12** | **preferredLanguage** | The preferred language for the user. Should follow ISO 639-1 Code; for example "en-US". |
| **13** | **userType** | A string value that can be used to classify user types in your directory, such as “Member” and “Guest”. |
| **14** | **onPremisesLastSyncDateTime** | Indicates the last time at which the object was synced with the on-premises directory. The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. |

**General Email Domains:**

This is another data source ([General Email Domains.xlsx](https://investcorp.sharepoint.com/:x:/s/tech/ETp94xGkc09OjDBf_QHVVpkBdxKVhL87nmRn338AdyOSQQ?e=sAAPZc)) we are using in our report. This source file contains more than hundred general email domains list. Used this data in many visuals. source file comes from share point & have access to some particular accounts like ([Sa\_powerbi@investcorp.com](mailto:Sa_powerbi@investcorp.com), [Nkarki@Investcorp.com](mailto:Nkarki@Investcorp.com) & [Ksunnapurallapalli@investcorp.com](mailto:Ksunnapurallapalli@investcorp.com) ) Those pages are.

1. **Ext. General Email Domains**
2. **LOB Details**
3. **General Email Domains -Top10**

**Power Query:**

**Sentmaildata:**

1. Import the data Get Data> Azure SQL Databases> use **server name:** sql-m365-prod-ne-01.database.windows.net and **Database Name:** db-m365-logs-prod-ne-01,
2. Choose the **VW\_EXTERNAL\_EMAIL\_REPORTING** view as a table source.
3. Once the data is loaded into the table add a conditional column ‘to\_Address’ by keeping the condition if [Recipient Type] = "TO" then [Recipient] else null this will separate the whole to address from the recipient.
4. In the same way, add two more conditional columns "cc\_address", with the following condition if [Recipient Type] = "CC" then [Recipient] else null. For "bcc\_address", put the condition like if [Recipient Type] = "BCC" then [Recipient] else null. This will separate the cc\_address and bcc\_address from the recipient type.
5. Make a copy of to\_address column and split the column into two parts using the ‘@’ delimiter. This will generate two new columns "to\_address - Copy.1"and "to\_address - Copy.2". Name "to\_address - Copy.2" as "To\_Domain\_Name".
6. Make a copy of "To\_Domain\_Name" column you will get "To\_Domain\_Name - Copy", and split the "To\_Domain\_Name - Copy" column using ‘.’(dot) delimiter then we will get "To\_Domain\_Name - Copy.1" and "To\_Domain\_Name - Copy.2" columns.
7. Rename "To\_Domain\_Name - Copy.2" column to "TO\_Domain\_Type" and remove "To\_Domain\_Name - Copy.1" column.
8. Rename the column "TO\_Domain\_Type" to "To\_Domain\_Type".
9. Duplicate the column "cc\_address" we will get "cc\_address - Copy". Split the column” cc\_address – Copy” using “@” delimiter then we will get "cc\_address - Copy.1", "cc\_address - Copy.2" columns.
10. Rename the column "cc\_address - Copy.2" to "CC\_Domain\_Name".
11. Remove the column "cc\_address - Copy.1".
12. Duplicate the column "CC\_Domain\_Name" we will get "CC\_Domain\_Name - Copy".
13. Split the "CC\_Domain\_Name - Copy" column using “.”(Dot) delimiter we will get two new columns "CC\_Domain\_Name - Copy.1", "CC\_Domain\_Name - Copy.2".
14. Rename the column "CC\_Domain\_Name - Copy.2" to "CC\_Domain\_Type".
15. Remove the column "CC\_Domain\_Name - Copy.1".
16. Duplicate the column "bcc\_address" then we will get "bcc\_address - Copy".
17. Split the column "bcc\_address - Copy" using the “@” then we will get "bcc\_address - Copy.1", "bcc\_address - Copy.2".
18. Rename the column "bcc\_address - Copy.2" to "BCC\_Domain\_Name".
19. Split the column "bcc\_address - Copy.1" using the delimiter “. ”(Dot) then we will get new 2 columns "bcc\_address - Copy.1.1" and "bcc\_address - Copy.1.2"
20. Duplicate the column "BCC\_Domain\_Name" and we will get "BCC\_Domain\_Name - Copy" column.
21. Split the column "BCC\_Domain\_Name - Copy" using the delimiter “.” Then we will get two new columns "BCC\_Domain\_Name - Copy.1" and "BCC\_Domain\_Name - Copy.2"
22. Rename the column "BCC\_Domain\_Name - Copy.2" to "BCC\_Domain\_Type".
23. Remove the columns "BCC\_Domain\_Name - Copy.1", "bcc\_address - Copy.1.1" and "bcc\_address - Copy.1.2".
24. Rename the following columns "To\_Domain\_Type" to "To\_Domain\_Form", "CC\_Domain\_Type" to "CC\_Domain\_Form", "BCC\_Domain\_Type" to "BCC\_Domain\_Form".
25. Add a conditional column "To\_Domain\_Type", with following condition : if [To\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else if [To\_Domain\_Name] = null then "NULL" else if [To\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else if [To\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else "EXTERNAL"
26. Add a conditional column "CC\_Domain\_Type" with the following condition: if [CC\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else if [CC\_Domain\_Name] = null then "NULL" else if [CC\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else if [CC\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else "EXTERNAL"
27. Add a conditional column "BCC\_Domain\_Type" with the following condition: if [BCC\_Domain\_Name] = "Investcorp.com" then "INTERNAL" else if [BCC\_Domain\_Name] = null then "NULL" else "EXTERNAL"
28. Add a column "Domain\_Name” from the example use the following logic that get the text after the delimiter “@” from the column [to\_address].
29. Add a conditional column "Domain\_Type", with the following condition: if [Domain\_Name] = "Investcorp.com" then "Internal" else if [Domain\_Name] = "Investcorp.com" then "Internal" else if [Domain\_Name] = "Investcorp.com" then "Internal" else if [Domain\_Name] = "" then "Null" else "External".
30. Remove the following columns "Domain\_Name" and "Domain\_Type".
31. Add a custom column "Attachment Type" to extract the extension of the file from the column [attachment\_contentType]
32. Add a merged column "EmailDomains", by combining the columns[To\_Domain\_Name], [CC\_Domain\_Name], [BCC\_Domain\_Name]
33. Convert the text to lower case in the column "sender\_name"
34. Extract the month name from [sentDateTime] and name it as "Month\_Name" the month name should be “MMM” format. Ex: JAN, FEB, MAR etc.,
35. Create two parameters “RangeStart” and “RangeEnd” with date/time type and the respective values are 01/01/2022 12:00:00 AM and 07/01/2023 (current date).
36. Apply the text filter on [sentDateTime] column with the following condition [sentDateTime] is after and equal to (>=) RangeStart and [sentDateTime] is below (<)RangeEnd.
37. The above date filter is for incremental refresh.

A screenshot of a computer

Description automatically generated

**Sentmail\_userdetails:**

1. Create an azure SQL Databases connection using get data option **Server name:** "sql-m365-prod-ne-01.database.windows.net", **Database Name:** "db-m365-logs-prod-ne-01" **table Name:** Sentitems.userdetails.
2. Once the data is loaded remove the duplicates from the column "mail".
3. Rename the column "department" to "department old".
4. Add an index column to the data using index column option.
5. Replace the values using conditional column as follows if [userPrincipalName] = "ClientServices-BBS@Investcorp.com" or [userPrincipalName] = "ClientServices-Ops@Investcorp.com" then "Corporate and Investment Accounting" else if [userPrincipalName] = "Investcorp-IRMCA-Compliance@Investcorp.com" then "Investcorp Private Wealth" else if [userPrincipalName] = "Reception\_Mumbai@Investcorp.com" then "Corporate General Services-Gulf & Asia" else if [userPrincipalName] ="LondonReception@Investcorp.com" then "Corporate General Services-London" else if [userPrincipalName]="indiafunds@Investcorp.com" then "Investment Admin" else if [userPrincipalName] = "ICMInvestorRelations@investcorp.com" then "Investcorp Credit Management" else if [userPrincipalName] = "investcorp-superreturn2023@Investcorp.com" then "Institutional Capital Raising - Europe" else if [userPrincipalName]= "cma@investcorp.com" then "Private Equity - Technology" else if [userPrincipalName] = "wealth@Investcorp.com" then "Investcorp Private Wealth" else if [userPrincipalName]= "IR-NA@investcorp.com" then "Institutional Capital Raising - North America" else if [userPrincipalName] ="Reception-Bahrain@Investcorp.com" then "Corporate General Services-Gulf & Asia" else if [userPrincipalName] ="RECEPTIONNY@Investcorp.com" then "Corporate General Services-New York" else "",Replacer.ReplaceValue,{"department"})

**General Email Domains:**

1. Create a table using sharepoint folder option from get data site URL link: <https://investcorp.sharepoint.com/sites/tech/> and the file name is **General Email Domains.xlsx**
2. Use first row as header option to make the first row as a header if the power BI unable to header.

**Date Last Refreshed:**

1. let
2. Source = #table(type table[Date Last Refreshed=datetime], {{DateTime.LocalNow()}})
3. in
4. Source

**Prod-PowerBI-Gateway:**

This is a On-premises data gateway which is connected to our report in power bi services. Since our source is Azure Data factory/Azure SQL databases we cannot access the dataset without gateways for security reasons. [Sa\_Powerbi@investcorp.com](mailto:Sa_Powerbi@investcorp.com) account has the permissions to access that gateway & Gaurav Pandey(gpandey@Investcorp.com) is the admin. Dashboard refresh date and timings are based out of Bahrain time zone. Triggers are also works under the Bahrain time Zone(UTC+3:00 Kuwait, Riyadh). Schedule Refresh is one time per day. This time is based on the data importing time in the Azure Triggers & SQL database.

A screenshot of a computer

Description automatically generated

