**Process to load SharePoint files to Azure SQL DB**

**Overview**

Currently, there’s no built-in connector available in ADF to load files stored in SharePoint folders.

The only option available is to use Power Automate Premium which costs $15/month.

So, following procedure helps us to get files stored in SharePoint Library/Folder from a SharePoint site.

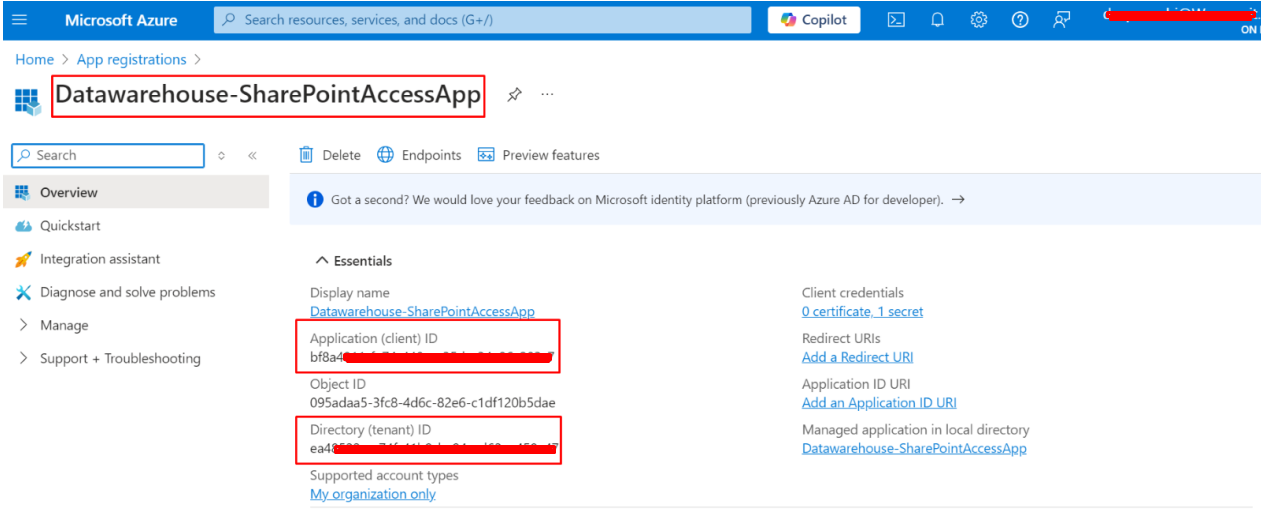
This method can be used to load files of type CSV, EXCEL, JSON

**Prerequisites:**

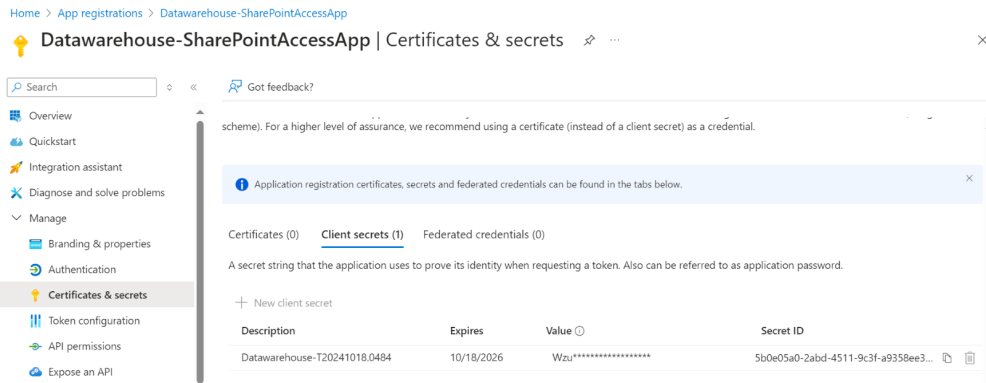
1. Azure Subscription
2. Ability to Register an App in Azure
3. Data Factory
4. Azure Blob Account
5. Access to SharePoint Site
6. Ability to Grant Read Permissions to SharePoint Site

**Steps**

1. Register an App in Azure



1. Create a secret for this app (we need the secret value in the later steps)



1. Create your actual URL from following template URL

Template URL: https://<your-site-url>/\_layouts/15/appinv.aspx

Actual URL: https://mydomain.sharepoint.com/sites/mysharepointsite/\_layouts/15/appinv.aspx

1. Go to your actual URL to give permissions to the registered app to read SharePoint site

We need to fill proper info in the pop up box

1. Application (Client ID) of registered app (highlighted in step1 screenshot) into App Id
2. Add actual name of registered app under Title e.g. Datawarehouse-SharePointAccessApp
3. Redirect URL remains as shows in image
4. Standard Script for Permissions as follows:

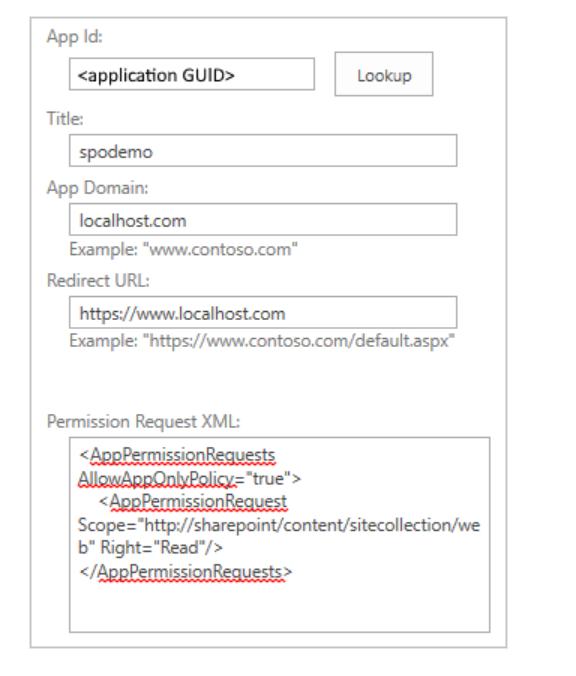
<AppPermissionRequests AllowAppOnlyPolicy="true">

<AppPermissionRequest Scope="http://SharePoint/content/sitecollection/web" Right="Read"/>

</AppPermissionRequests>

1. Create and Trust it

*\*If you do not have permission to perform step 4 then you need to ask your IT department to perform this step*



1. Head to Data Factory to create a pipeline which will copy files for SharePoint into Azure SQL DB and Create Web activity to get access token from SharePoint site
2. Template URL for Web activity: <https://accounts.accesscontrol.windows.net/[Tenant-ID]/tokens/OAuth/2>

Actual URL : https://accounts.accesscontrol.windows.net/ea48522a-c75f-47b9-ba94-ed62ee458a47/tokens/OAuth/2

*Get your Azure tenant ID (highlighted in step 1 screenshot)*

1. Method: POST
2. Create your own body using following template body:

*Client-ID - application id of registered app (step 1 screenshot)*

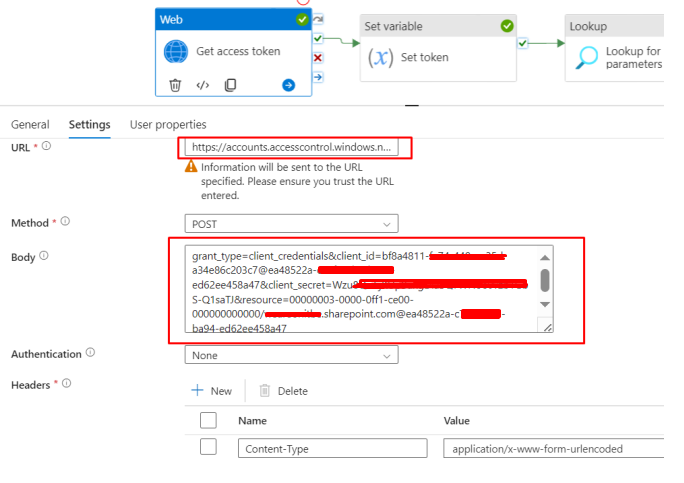
*Tenant-ID - Get your Azure tenant ID (highlighted in step 1 screenshot)*

*Client-Secret - The value of secret created for registered app in Azure*

*Tenant-Name- Your Domain name which appears under Azure portal*

Template Body: grant\_type=client\_credentials&client\_id=[Client-ID]@[Tenant-ID]&client\_secret=[Client-Secret]&resource=00000003-0000-0ff1-ce00-000000000000/[Tenant-Name].sharepoint.com@[Tenant-ID]

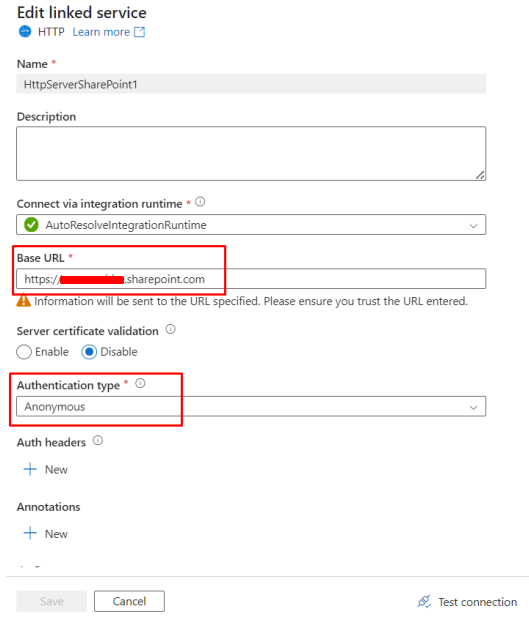
1. Put Headers like following:



1. Running this web activity will provide access token which can be used later to fetch files from SharePoint site
2. Create a linked service to connect to SharePoint site with base URL

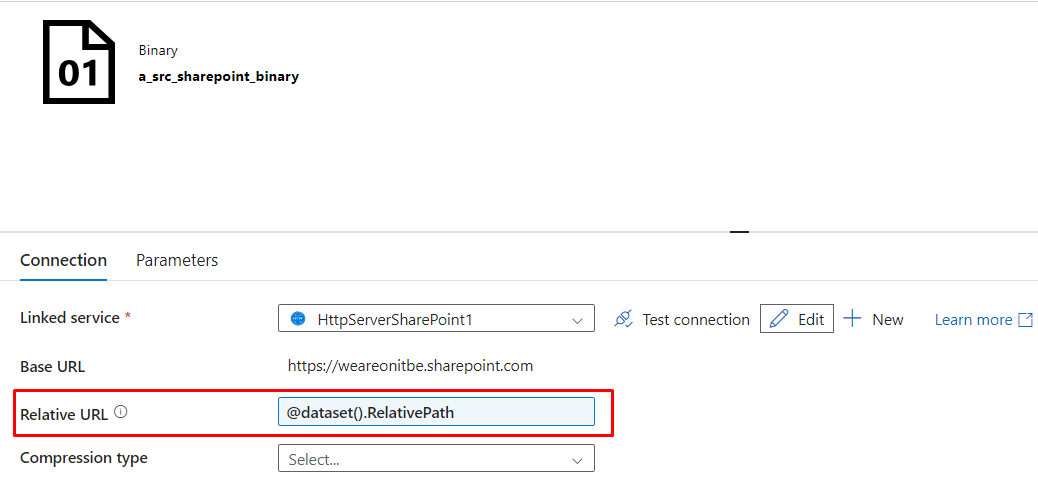
Select HTTP as a source

Base URL is URL of your SharePoint



1. Create a HTTP Dataset for a binary file type for SharePoint source

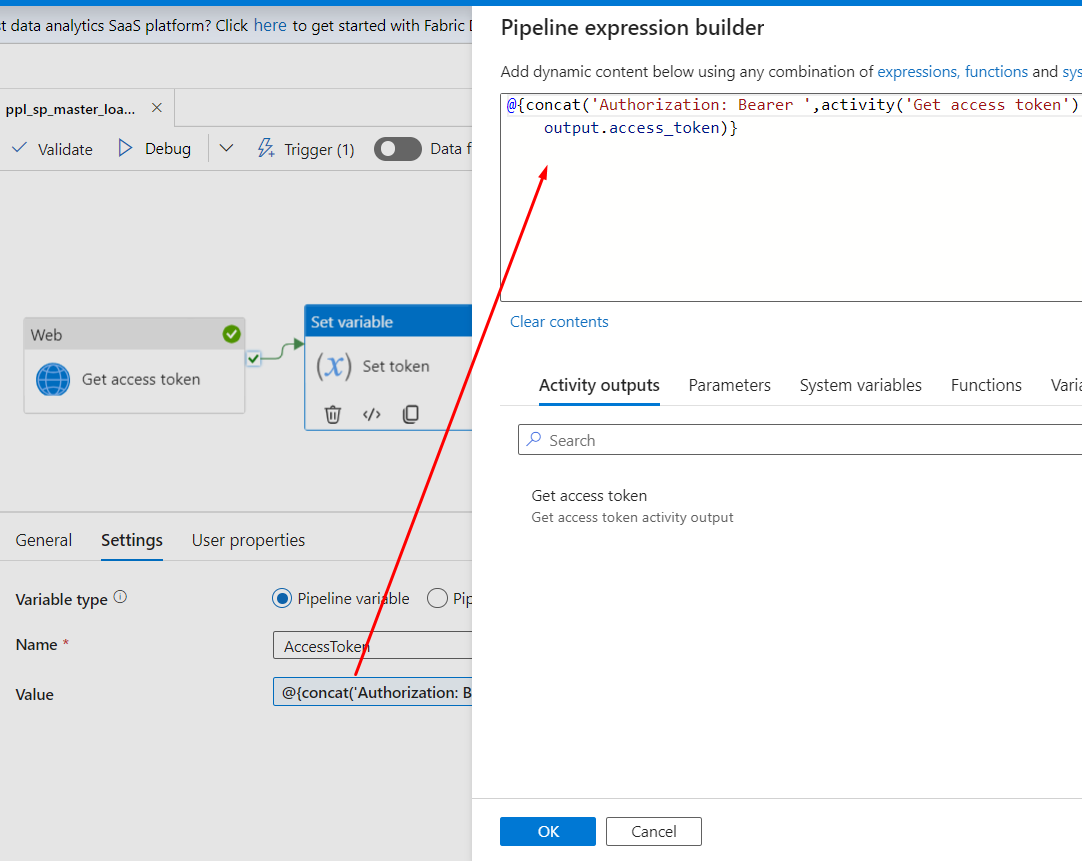
Create a parameter as “Relative path” for a dataset so that we can fetch any file stored in the SharePoint site



1. Create a variable called “AccessToken” to store access token obtained from Web activity
2. Create Set Variable activity and select newly created variable in Name section.

For Value, click “Add dynamic content” and write following expression.

*Make sure the name of Web activity must be same in the expression for variable*



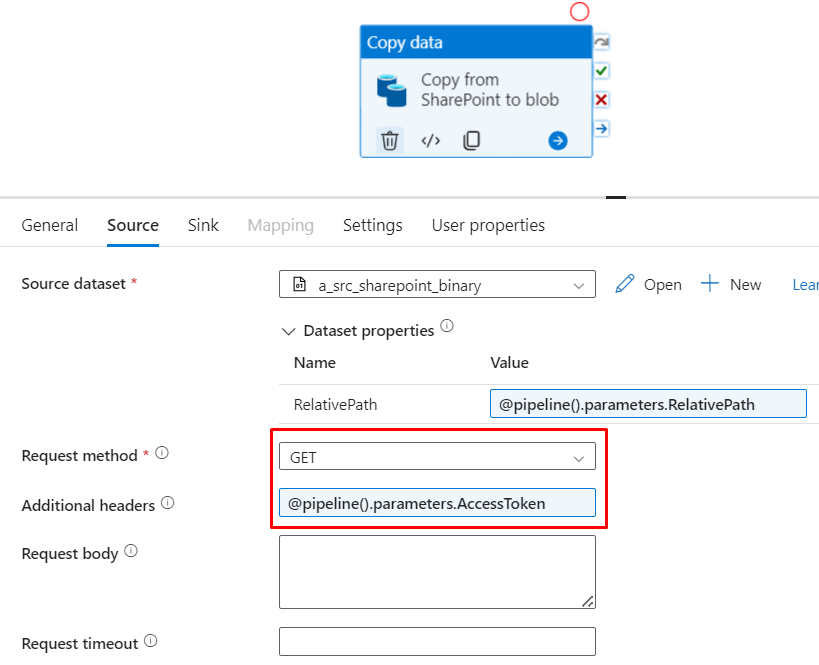
1. Create a copy activity in ADF to import file from SharePoint and store to Azure Blob

i) Select HTTP dataset created above as a Source

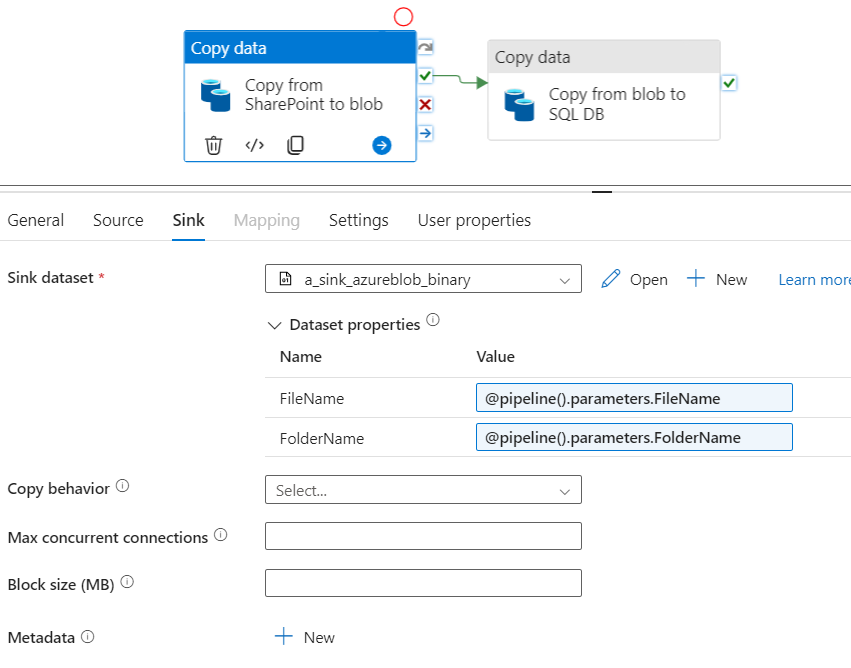
ii) Create a parameter for this pipeline and name it “RelativePath” and pass this parameter in the dataset

parameter “RelativePath” as shown in in image below.

iii) Pass the access token which is stored in a variable as additional header like following



1. Now create a sink dataset for Azure Blob
2. Then from Azure Blob we can use another copy activity to load excel into the SQL



1. Most important step - when running this pipeline, you need to mention a value for a parameter “RelativePath”. Create your own relative path for a file you want to copy to blob using template below:

Template relative path: /sites/[site-name]/\_api/web/GetFileByServerRelativeUrl('[relative-path-to-file]')/$value

Relative-path-to-file is a different thing. It is an actual path to your file. It can be found by right clicking on a required file, go to details and copy path.

Example of actual relative path of a file:

/sites/MySharePointSite/\_api/web/GetFileByServerRelativeUrl('/sites/MySharePointSite/Generaldocuments/General/Source files/Sales.xlsx')/$value

1. Final pipeline will look like following

