

Configuration settings to initialize the Data Transformation Job

1. Steps to retrieve the configuration parameters –
 - a. Copy the **Get-ConfigurationParams.ps1** in **C:\DataTransformation** folder.
 - b. Run the Get-ConfigurationParams.ps1 script with the following command in Windows PowerShell

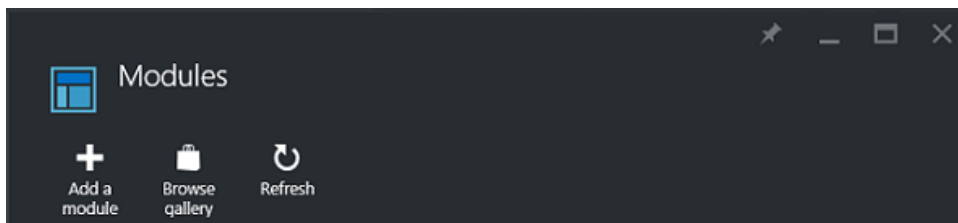
```
C:\DataTransformation\Get-ConfigurationParams.ps1 -SubscriptionName  
"AzureSubscriptionName" -ActiveDirectoryKey "anyrandompassword" -AppName  
"ApplicationName"
```

[You can pass in any values for the ActiveDirectoryKey and the AppName]

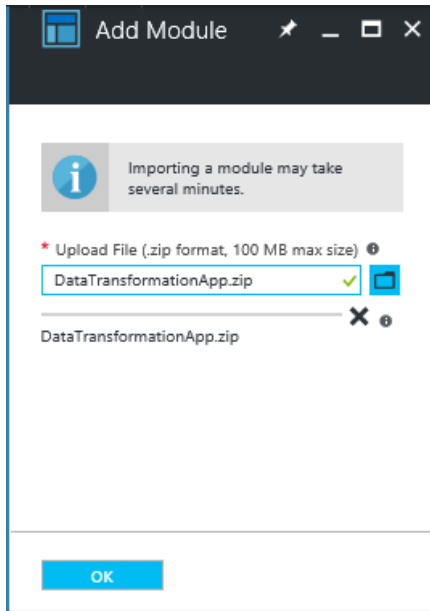
3. This script will output the following values that should be used while triggering the automation runbook, please note down these –
 - a. Client Id
 - b. Tenant Id
 - c. ActiveDirectoryKey (same as the one entered above)
 - d. Subscription Id

To import a module from local machine

1. Logon to the Azure [site](#)
2. In the Azure Portal, open your Automation account.
3. Click on the **Assets** tile to open the list of assets.
4. Click on the **Modules** tile to open the list of modules.
5. Click on the **Add a module** button and the Add module blade is launched.



6. After you have selected the module from your local computer. Click **OK** to import the module. When Azure Automation imports a module to your account, it extracts metadata about the module.



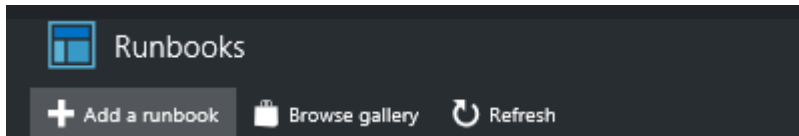
This may take a couple of minutes since each activity needs to be extracted.

9. You will receive a notification that the module is being deployed and a notification when it has completed. Also you can check the status in **Modules** tile.

NAME	LAST MODIFIED	STATUS
Azure	12/16/2016 11:58 PM	Available
Azure.Storage	12/17/2016 12:05 AM	Available
AzureRM.Automation	12/17/2016 12:03 AM	Available
AzureRM.Compute	12/17/2016 12:03 AM	Available
AzureRM.Profile	12/17/2016 12:02 AM	Available
AzureRM.Resources	12/17/2016 12:04 AM	Available
AzureRM.Sql	12/17/2016 12:04 AM	Available
AzureRM.Storage	12/17/2016 12:05 AM	Available
DataTransformationApp	1/6/2017 5:24 PM	Available
Microsoft.PowerShell.Core	12/16/2016 11:59 PM	Available
Microsoft.PowerShell.Diagnost...	12/16/2016 11:59 PM	Available
Microsoft.PowerShell.Manage...	12/17/2016 12:00 AM	Available
Microsoft.PowerShell.Security	12/17/2016 12:01 AM	Available
Microsoft.PowerShell.Utility	12/17/2016 12:01 AM	Available
Microsoft.WSMan.Management	12/17/2016 12:02 AM	Available
Orchestrator.AssetManageme...	12/17/2016 12:06 AM	Available

To import a runbook from a file with the Azure portal

1. In the Azure portal, open your Automation account.
2. Click on the **Runbooks** tile to open the list of runbooks.
3. Click on the **Add a runbook** button and then **Import and existing runbook (Import)**.



4. Click **Runbook file** to select the file to import (Trigger-DataTransformation-Job)
5. The **name** will be automatically filled, which is non-editable.
6. The **runbook type** will be automatically selected, but you can change the type after taking the applicable restrictions into account.
7. Click **Create** to import the runbook.
8. The new runbook will appear in the list of runbooks for the Automation Account.
9. Click the **Trigger-DataTransformation-Job** runbook.
10. Click the **Edit** button.
11. Click the **Publish** button and then **Yes** to the verification message.

To Start a runbook:

1. In the Azure portal, open your Automation account.
2. Click on the Runbooks tile to open the list of runbooks.
3. Click the **Trigger-DataTransformation-Job** runbook.
4. Click **Start** button that starts the runbook



5. Opens below blade, enter all params then Click OK then it submits the Data transformation job.

Start Runbook
Trigger-DataTransformation-Job

Parameters

* SUBSCRIPTIONID ⓘ

Enter a value

Mandatory, String

* TENANTID ⓘ

Enter a value

Mandatory, String

* CLIENTID ⓘ

Enter a value

Mandatory, String

ACTIVEDIRECTORYKEY ⓘ

No value

Mandatory, String

RESOURCEGROUPNAME ⓘ

No value

Mandatory, String

DATAMANAGERNAME ⓘ

No value

Mandatory, String

JOBDEFINITIONNAME ⓘ

No value

Mandatory, String

Run Settings

Run on Azure ⓘ

OK