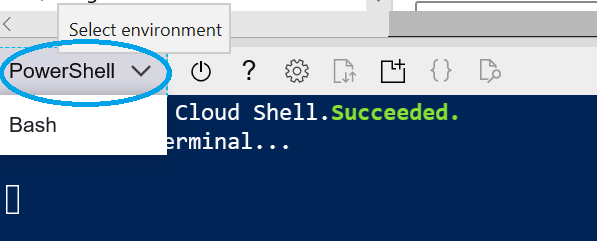
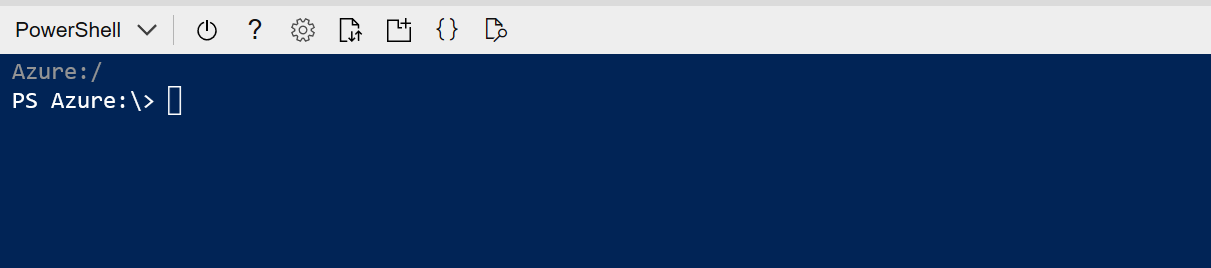
**ARM Template**

1. Create a folder in C:/ drive and name is ‘Azure’
2. Create a folder ‘ADFUsageDetails’ in C:/Azure folder
3. Download Usagedetails.zip and extract the files to ‘C:\Azure\ADFUsageDetails’ folder
4. Go to portal.azure.com
5. Log into Azure portal
6. Open cloud shell and select power shell environment.

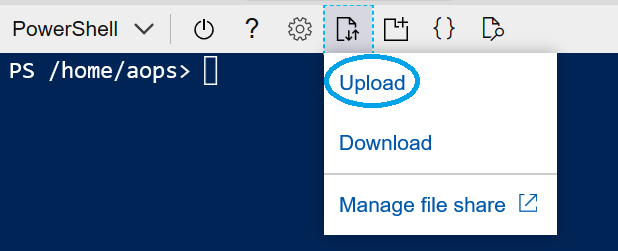




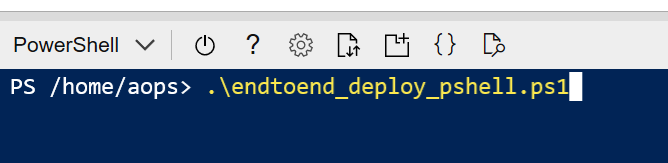
1. Set subscription to the subscription you want this solution to be deployed by running the following command
   1. set-azcontext -subscription <*subscriptuonName*>



1. Navigate to home directory : cd $home
2. Upload the following two files to power shell from ‘C:\Azure\ADFUsageDetails\FilesToUploadToPowershell’ folder
   1. endtoend\_consumption\_template\_sql-v1-1.json
   2. endtoend\_deploy\_pshell.ps1

**

1. run the power shell script



Enter a name for the resource group to contain all objects: <name for the new Resource group>

Specify an Azure Region to deploy the resource group: <region for the resource group. E.g. ‘Central US’>

Provide a name for the ARM deployment: <specify name for this deployment>

Provide the path to the ARM template file: <home directory path/ endtoend\_consumption\_template\_sql-v1-3.json >

/home/aops/endtoend\_consumption\_template\_sql-v1-3.json

Enter a globally unique name for the Azure Storage Account resource (3-24 characters, lowercase, ex: contoso-usage-store):

Enter a globally unique name for the Azure Key Vault resource (3-24 characters, lowercase, no special characters, ex: contosokv):

Enter a globally unique name for the Azure Data Factory resource (3-24 characters, lowercase, ex: contosoadf, contoso-datafactory):

Enter a globally unique name for the Azure SQL Server resource (ex: contoso-az-usage):

Enter an administrator name for SQL Server (cannot use admin, administrator):

Enter a password for SQL Server admin: <password for SQL server admin account>

Enter a name for SQL database: <name for SQL server db . e.g. usagedetails-db>

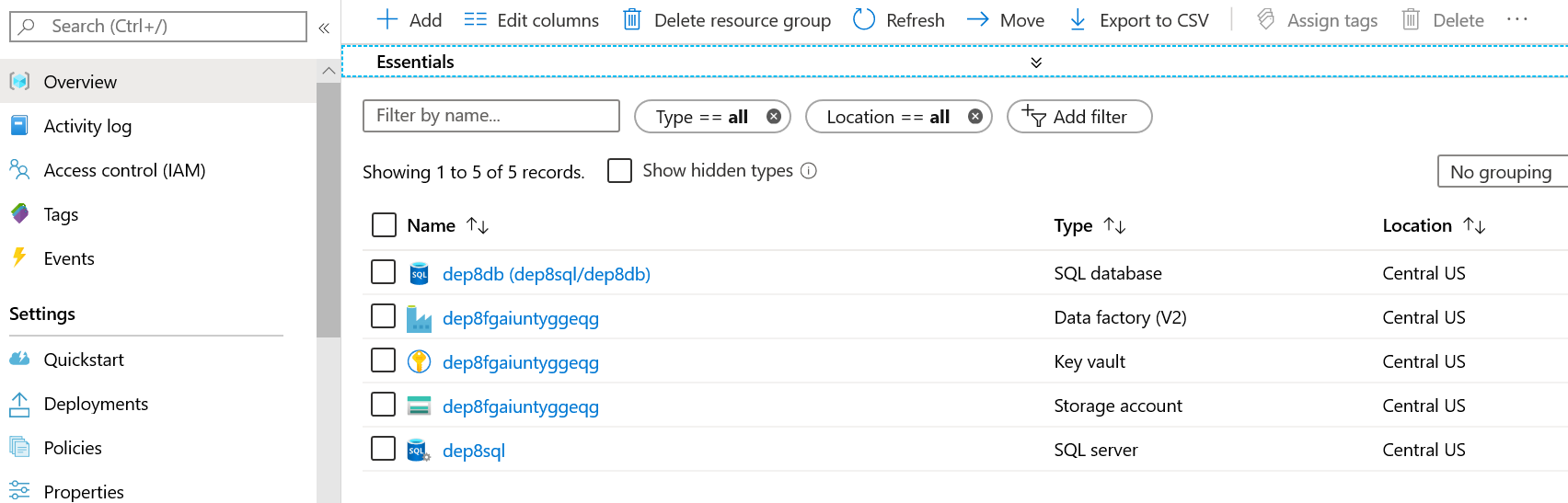
Confirm

Provided resource group already exists. Are you sure you want to update it?

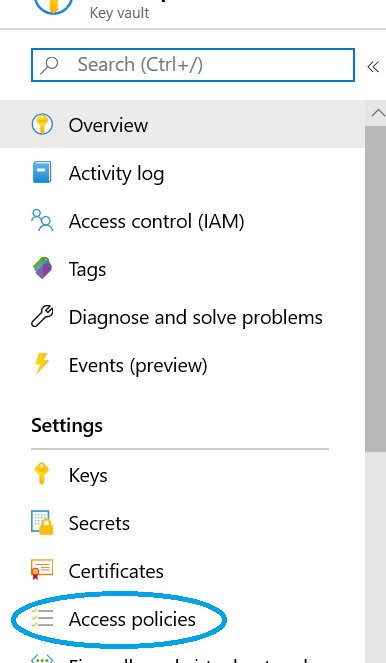
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): <type Y if you want to update Resource group **if it already exists**). Type ‘N’ if you don’t want to use existing Resource group.

This might take some time (please be patient).

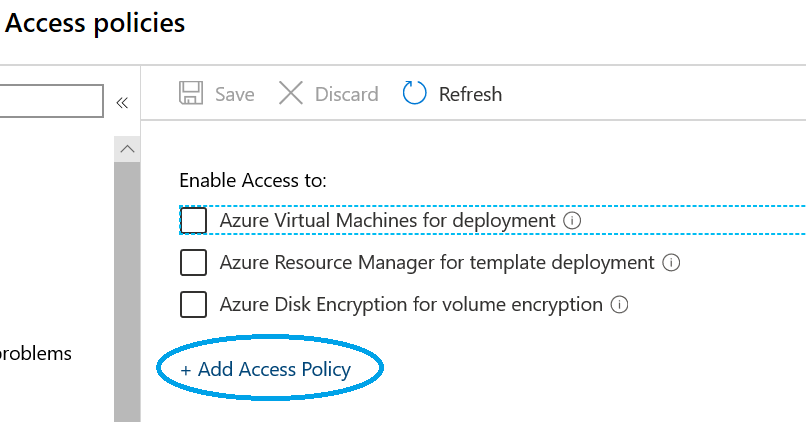
1. Make sure all the objects got created by clicking on the Resource group you just created.

**

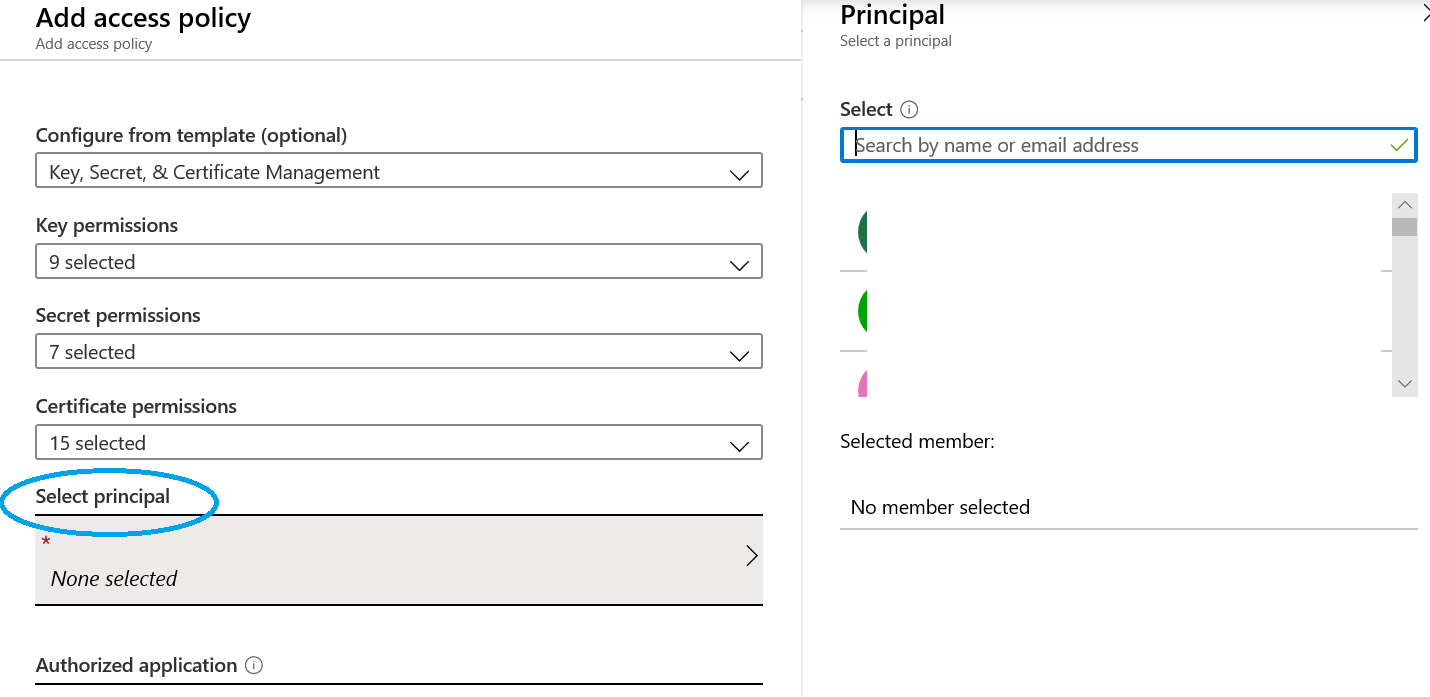
1. Click on the Key vault created above to update permissions.
   1. Select Access Policies



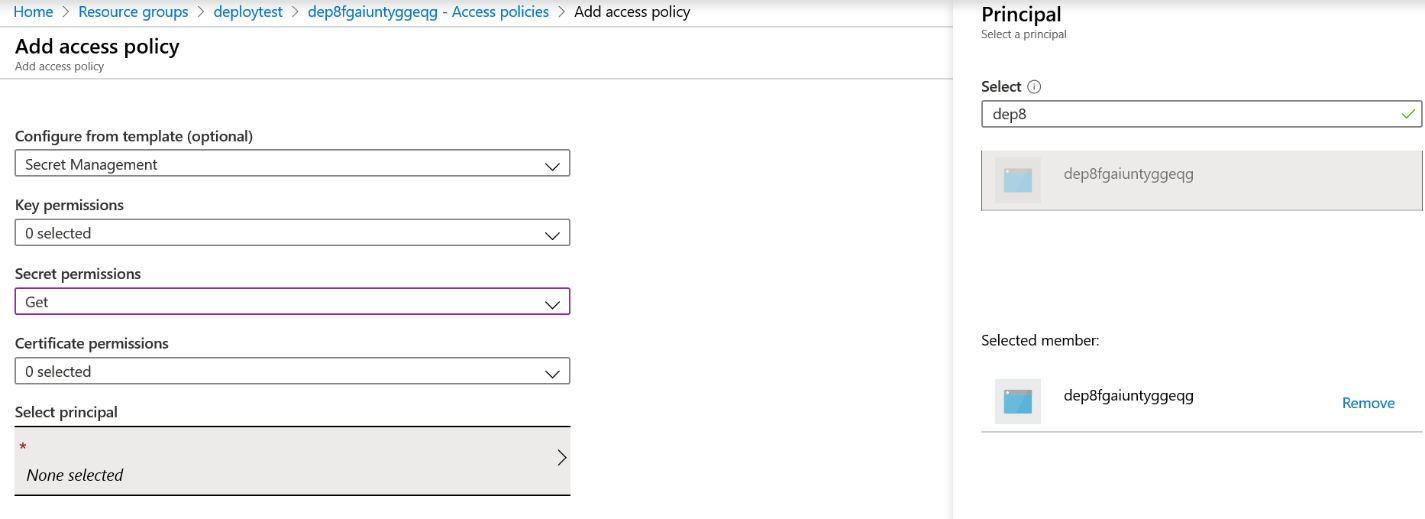
* 1. Click on ‘Add Access Policy’



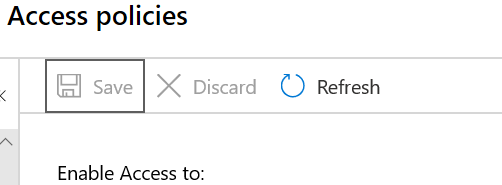
* 1. Add the following policies
     1. *Configure from template (optional) – Key, Secret, & Certificate Management* –
        1. Under ‘Select principal’ add desired user or group to grant permission.
        2. Click on ‘Select’ and then on ‘Add’



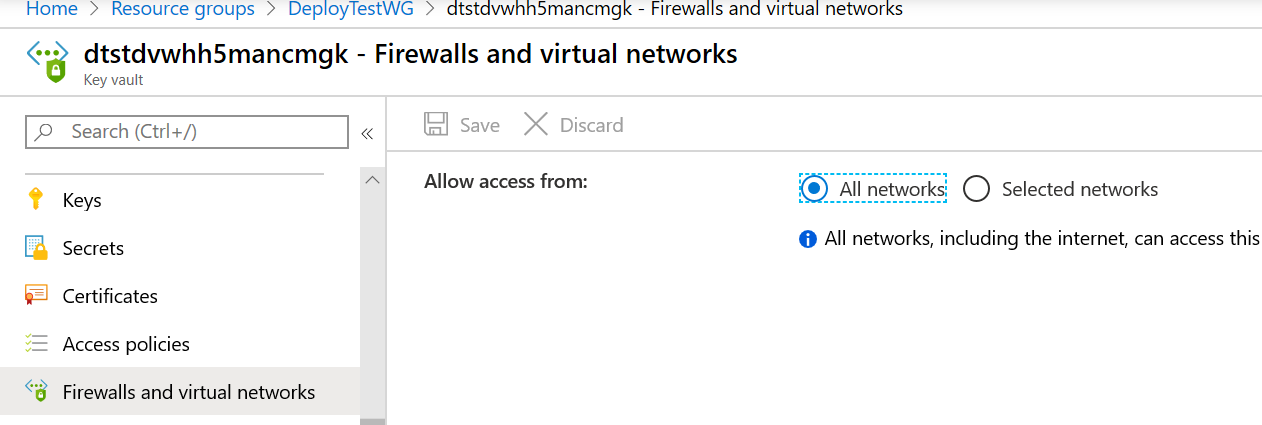
* + 1. Click on ‘Add Access Policy’ to add the second policy
    2. *Secret permissions – Secret Management Operations*
       1. *select Get* -under select principal search for the name of the data factory resource
       2. *click on ‘Select’ and then on ‘Add’*



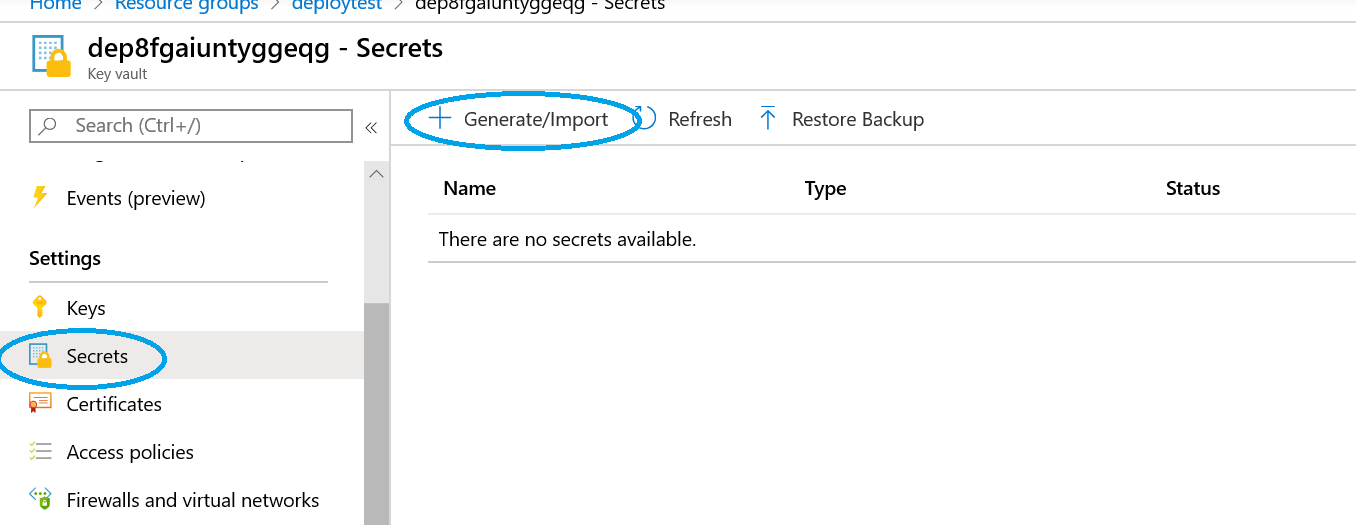
* 1. Save the changes



1. Add ‘All Networks’ in ‘Firewalls and virtual networks’ section.

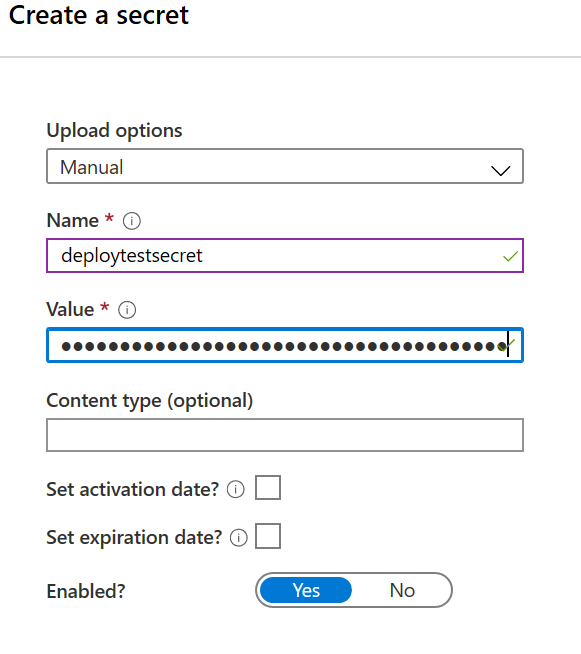


1. Click on Secrets on the left pane and then click on Generate import

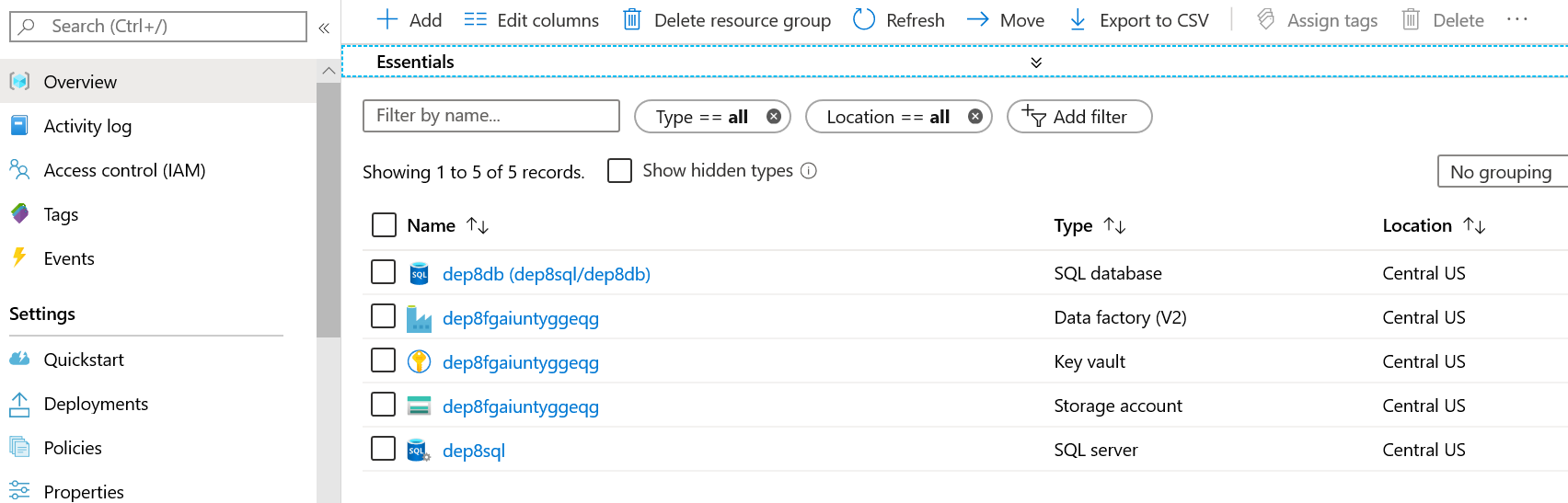


1. Create a new key vault Secret and provide the value of your API key. Add *Bearer* to the start of the key value

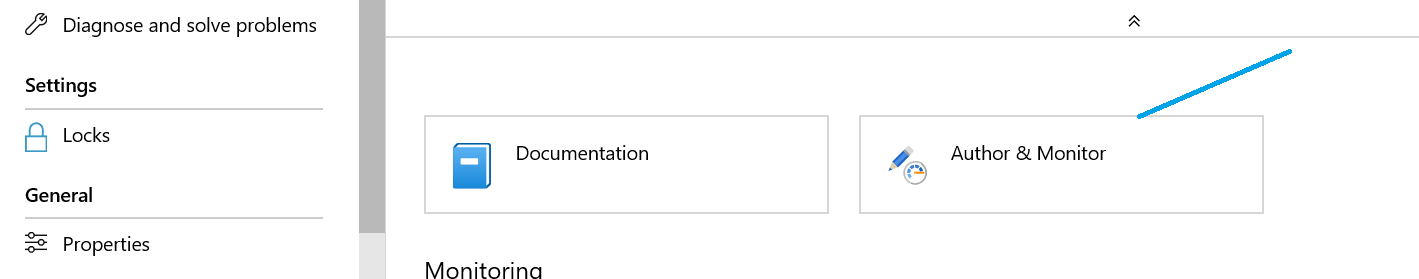
e.g. Bearer *<key>*



1. Go back to the resource group and click on the data factory.

**

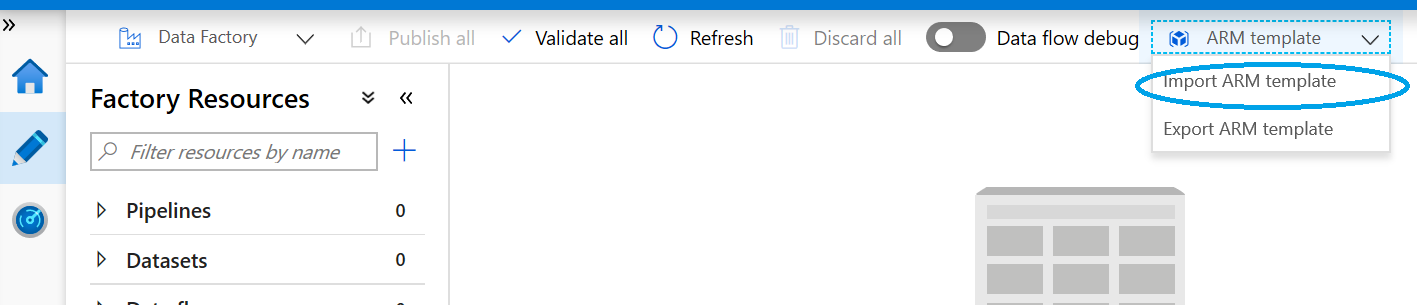
1. Click on Author and Monitor



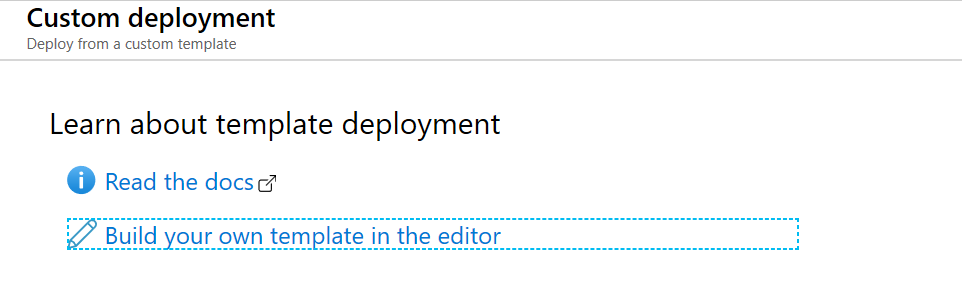
1. Click in Author icon



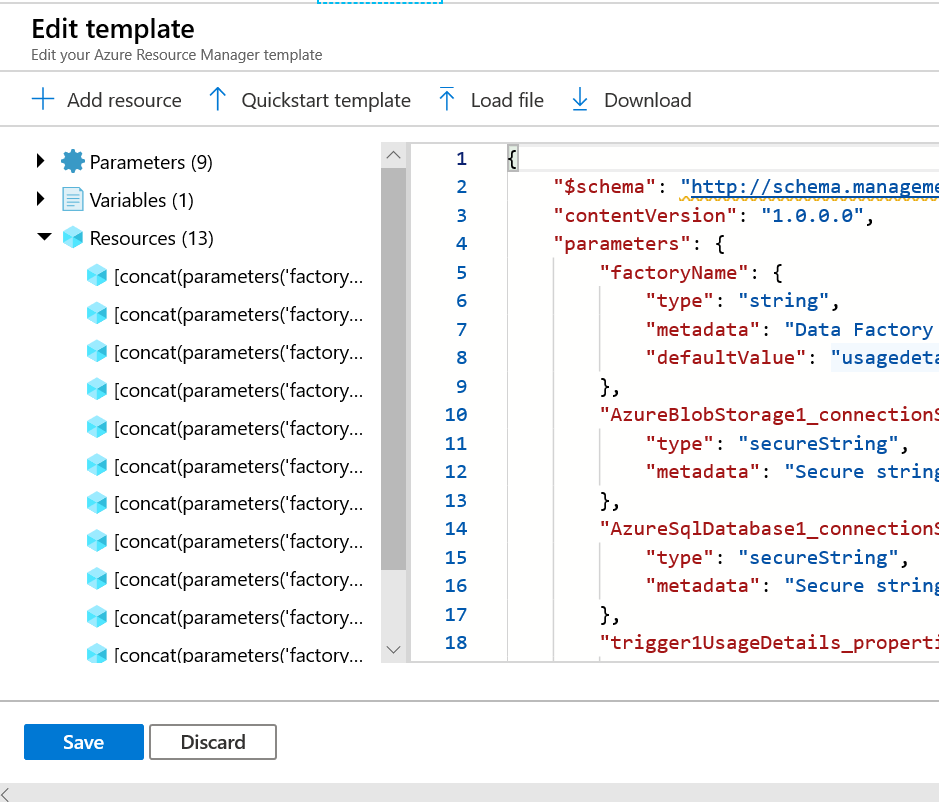
1. Select Import ARM template from ARM template drop down menu.



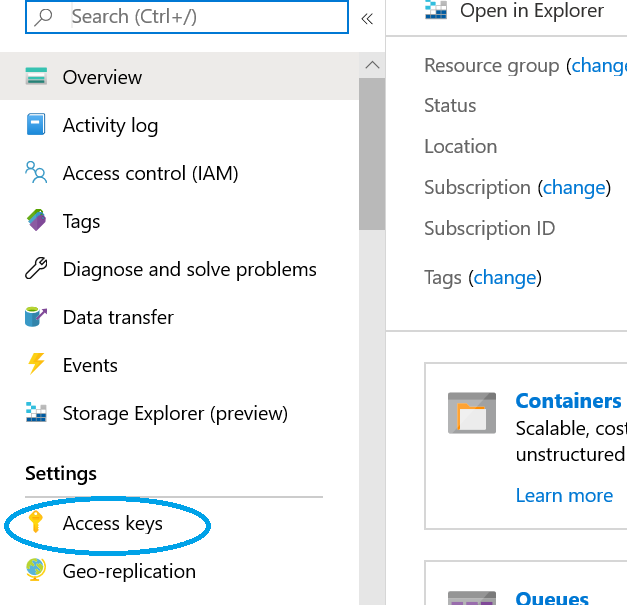
1. Select Build your own template in the editor



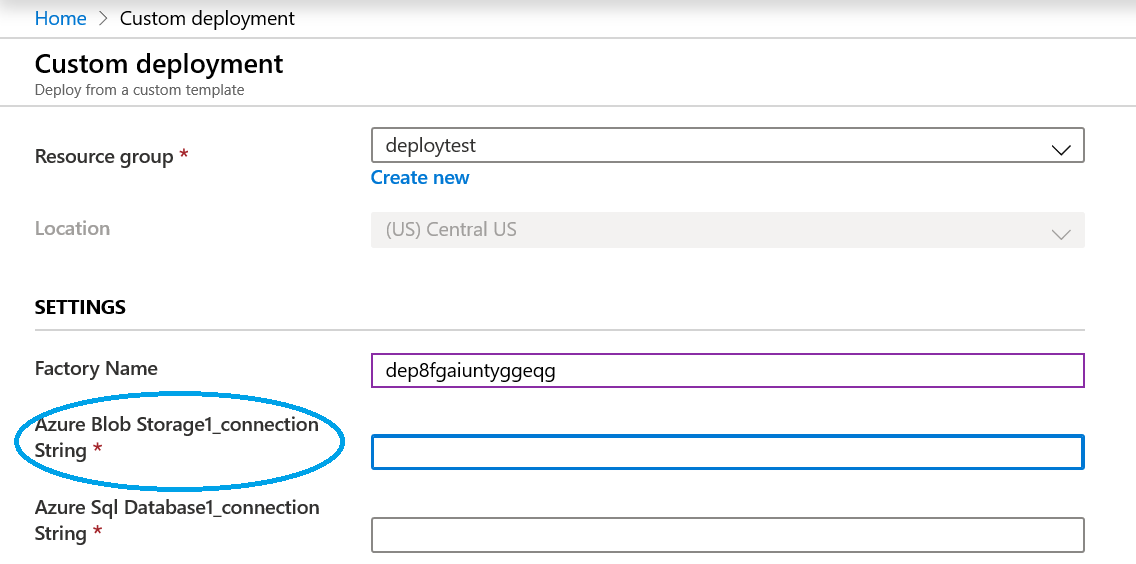
1. Click on ‘Load file’ and select arm\_template\_parameters.json file extracted in step 1 (from C:\Azure\ADFUsageDetails\ARMTemplate).
2. Click on ‘Load file’ again and select arm\_template.json file extracted in step 1(from C:\Azure\ADFUsageDetails\ARMTemplate).
3. Click on Save



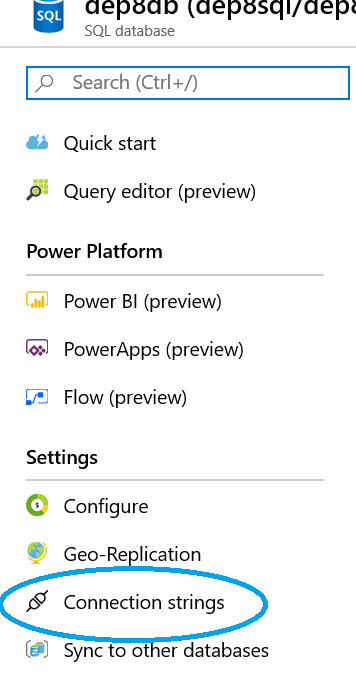
1. Select the subscription and Resource group (you created for this solutioncls) from the drop down
2. Make sure that the factory name is set to new Factory Name you created
3. In another tab open another Azure portal and navigate to the blob storage you created and click on Access Keys



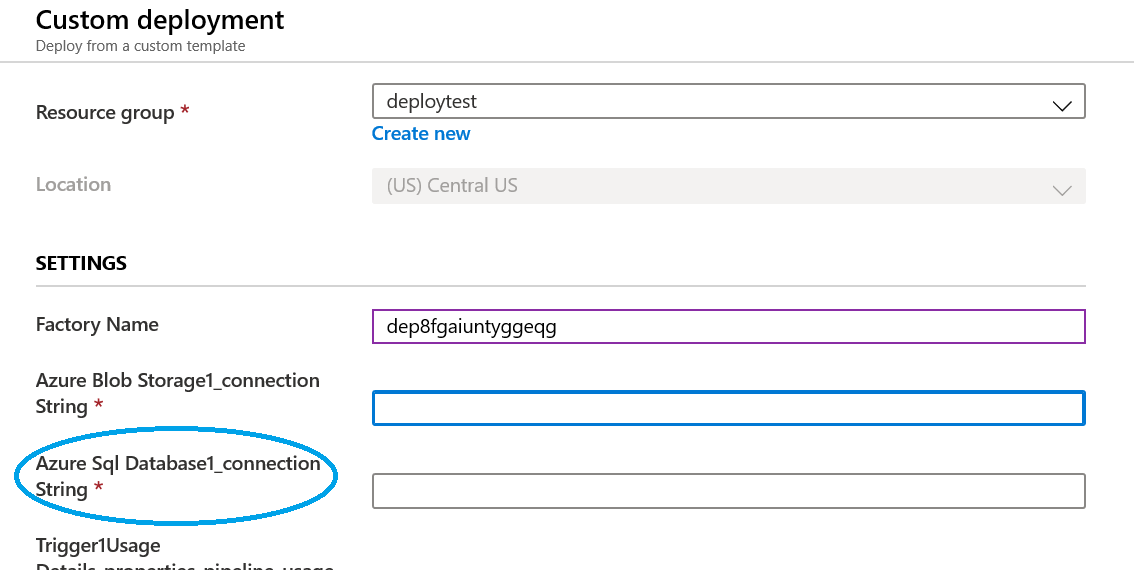
1. Copy connection String and paste in ‘Azure Blob Storage1\_connection String’ field.



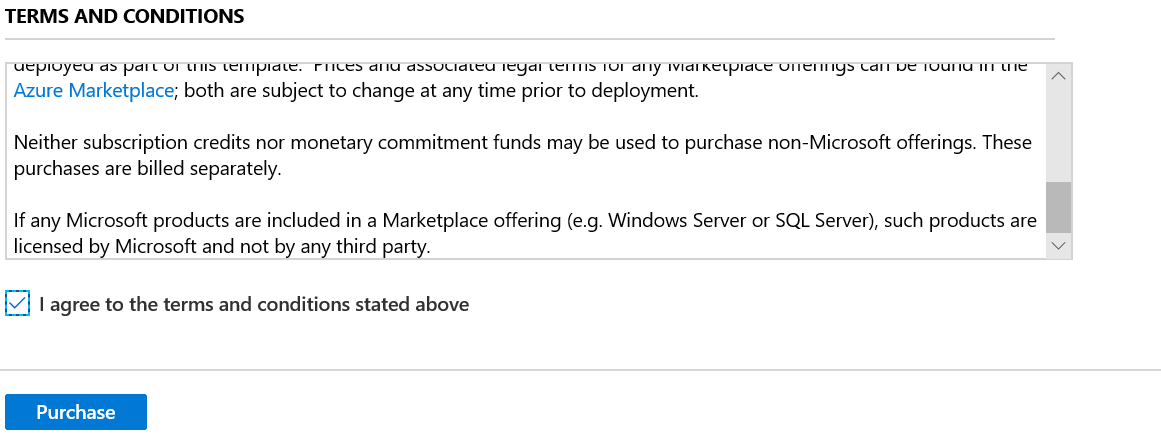
1. Navigate to SQL Database you created and click on connection strings



1. Copy the connection string (ADO.NET) and paste it into ‘Azure Sql Database1\_connection String’ field



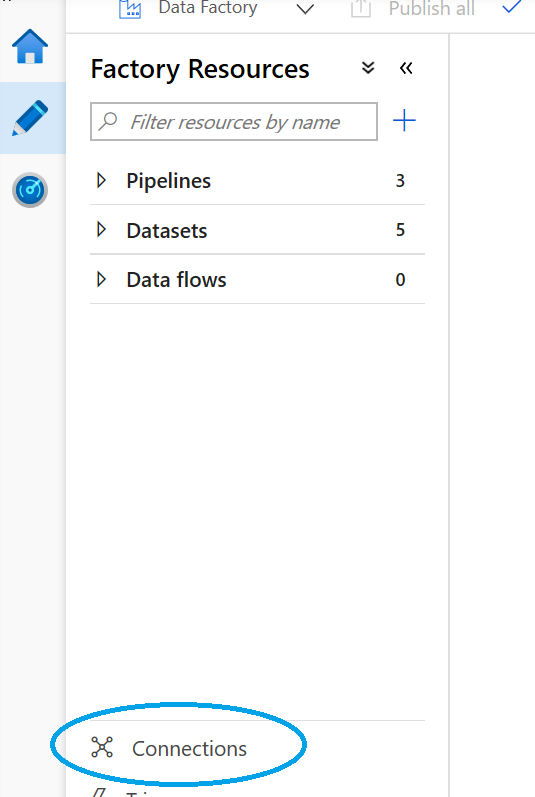
1. Scroll all the way down and select the checkbox and click on purchase



1. Navigate to the data factory and click on Author and monitor and then click on ‘Author’ icon.

Notice that 3 pipelines and 5 data sets are created.

1. Click on ‘Connections’

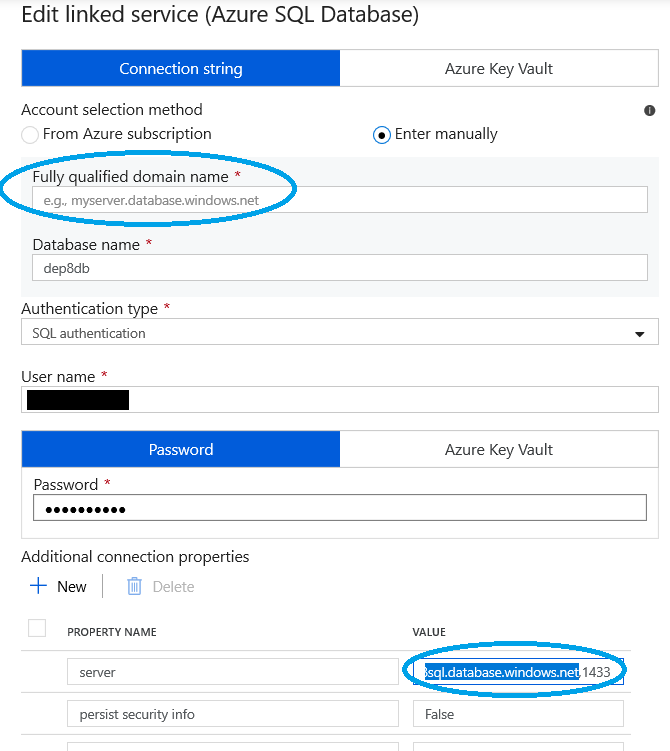


1. Click on ‘AzureSQLDatabase1’
2. Copy text from property ‘Server’ without tcp and port number and paste it into ‘Fully qualified domain name’ field.

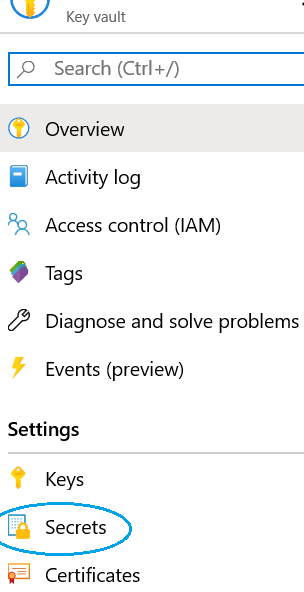
e.g. server: tcp:dep8sql.database.windows.net,1433

paste ‘dep8sql.database.windows.net’

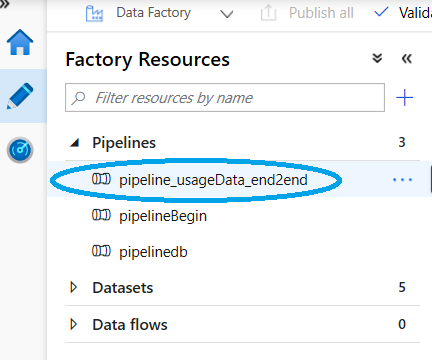
1. Re-enter password and click ‘Apply.



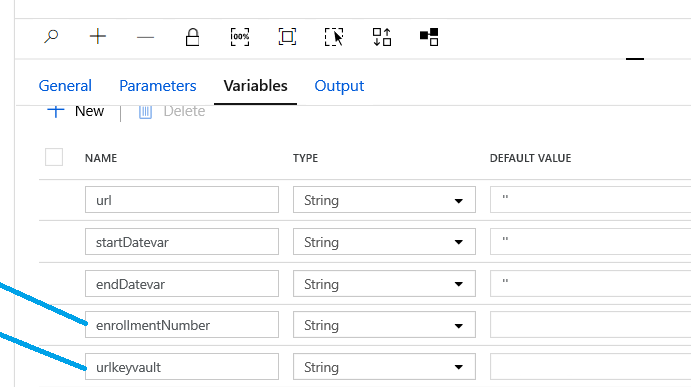
1. In the second tab: Go to Key Vault and click on Secrets



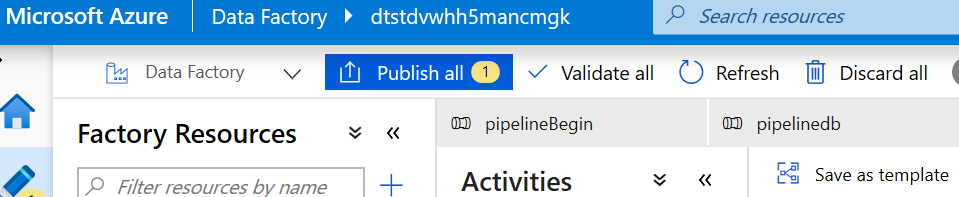
1. Copy the Secret identifier and append the following to the secret identifier
   1. ?api-version=7.0 and copy the entire identifier (with this text).
2. Now navigate to Data Factory tab
3. Expand ‘Pipelines’ and click on ‘pipeline\_usageData\_end2end’



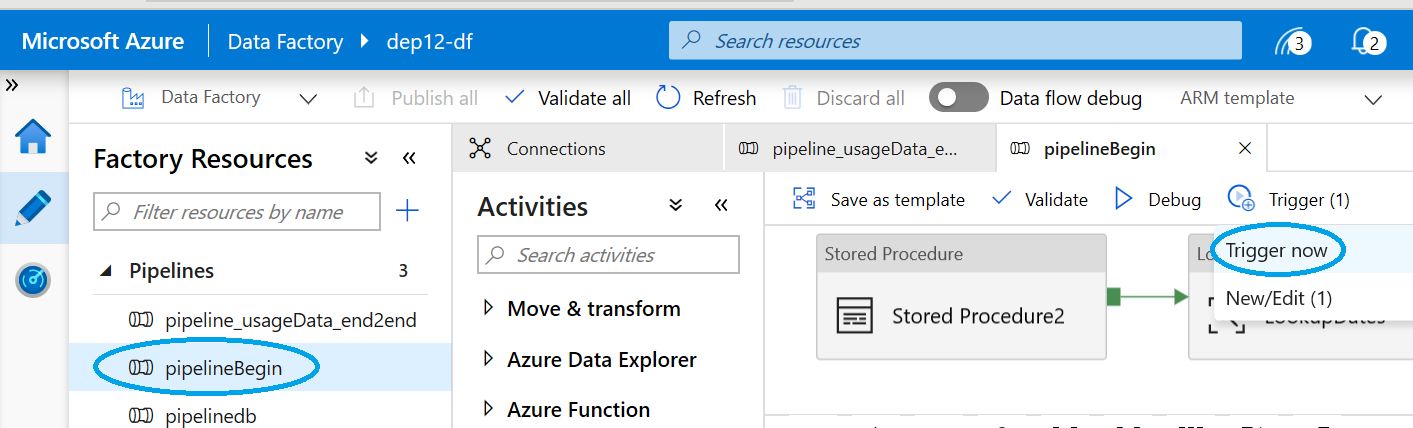
1. Select Variables and paste the copied identifier for variable ‘urlkeyvault’
2. Add your enrollment number for variable ‘enrollmentNumber’



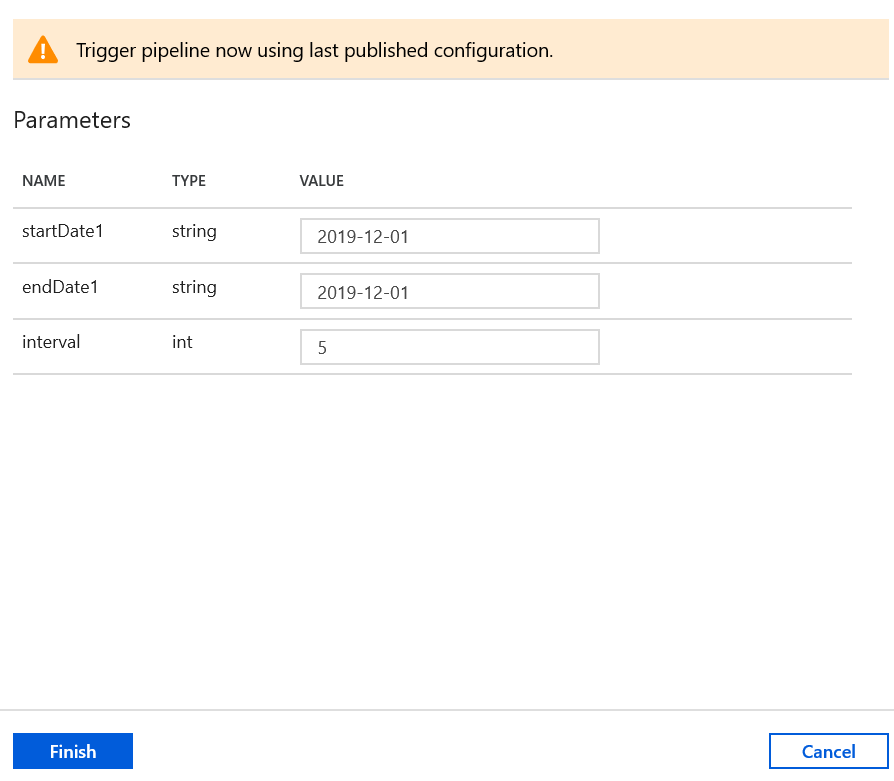
1. Click on Publish all



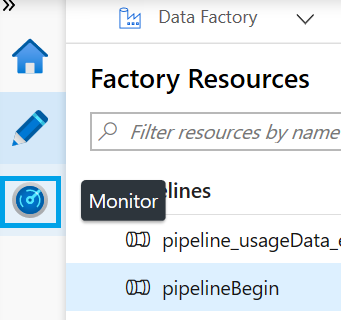
1. Manual steps in SQL db:
2. Create the following tables in Azure SQL db by running usageDetailsTables.sql
   1. usageDetailsFinal
   2. lookupDates
3. Create SQL Stored Procedures
   1. SP\_loadUsageDataToFinalTbl.sql
   2. SP\_lookupDates.sq



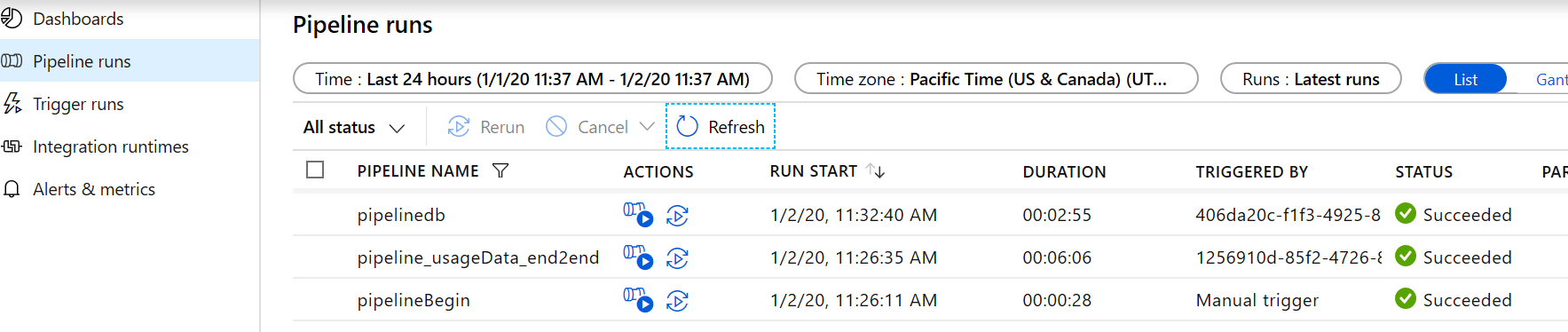
1. Test the deployment by running the pipeline for one day.
2. Click on Trigger Now



1. Update dates and click on ‘Finish’
2. Update dates and click on ‘Monitor’



1. Monitor the run
2. Once all 3 pipelines are completed, the status us shown as ‘Succeeded’



1. Validate the data by querying usageDetailsFinal table in SQL server.