

<< MCT > blog > arm\_template

Search arm\_template

Name	Date modified	Type
factory	06-03-2022 11:44	File folder
linkedTemplates	06-03-2022 11:44	File folder
arm_template.json	06-03-2022 11:44	JSON File
arm_template_parameters.json	06-03-2022 11:44	JSON File

## Restore Azure data factory

Before starting with the restore process we have to create a new data factory from azure portal. I have named the new data factory as *adfdemo100222*. Once created open the same in data factory studio and click on 'manage' tab from the left side pane, followed by the options highlighted in the image below to import ARM template. Ultimately we are going to load the 'arm\_template.json' file from the folder which we have extracted from the zip previously.

Microsoft Azure | restoreadfinstance

Data Factory | Validate all | Publish all

Home | Author | Monitor | **Manage**

Connections

- Linked services
- Integration runtimes
- Azure Purview

Source control

- Git configuration
- ARM template**

Author

- Triggers
- Global parameters

Security

- Customer managed key
- Credentials
- Managed private endpoints

### ARM template

The Azure Resource Manager (ARM) template is a JavaScript Object Notation (JSON) file that defines . [Learn more](#)

Azure Data Factory can be exported and updated as an ARM template artifact. [Learn more](#)

**Import ARM template**

Update your Azure Data Factory environment by importing an ARM template.

[Import on Azure portal](#)

**Export ARM template**

Export your Azure Data Factory as an ARM template.

[Export](#)

### ARM parameter configuration

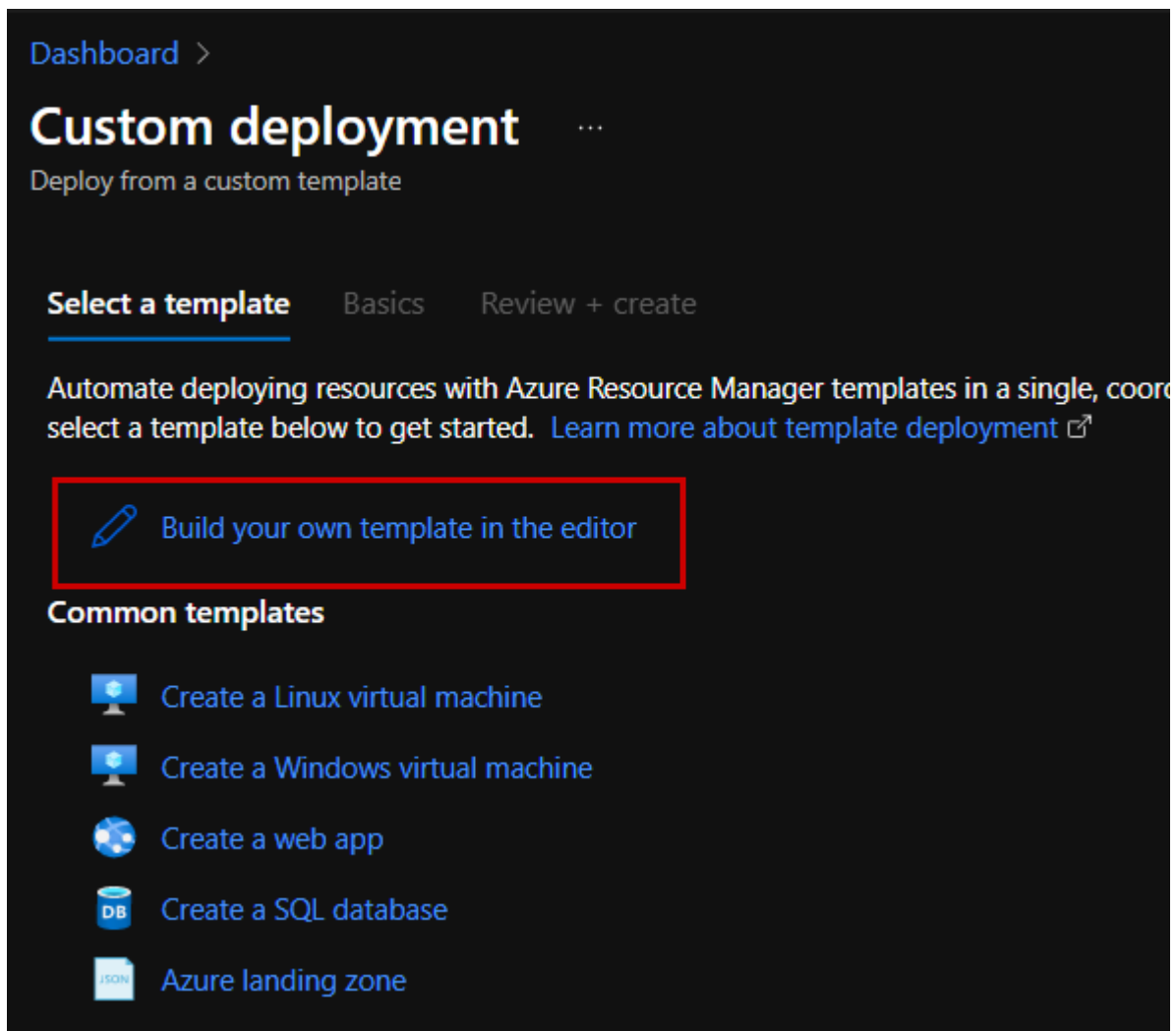
This configuration determines which properties are parameterized when generating the Azure Resource Mani

**Edit parameter configuration**

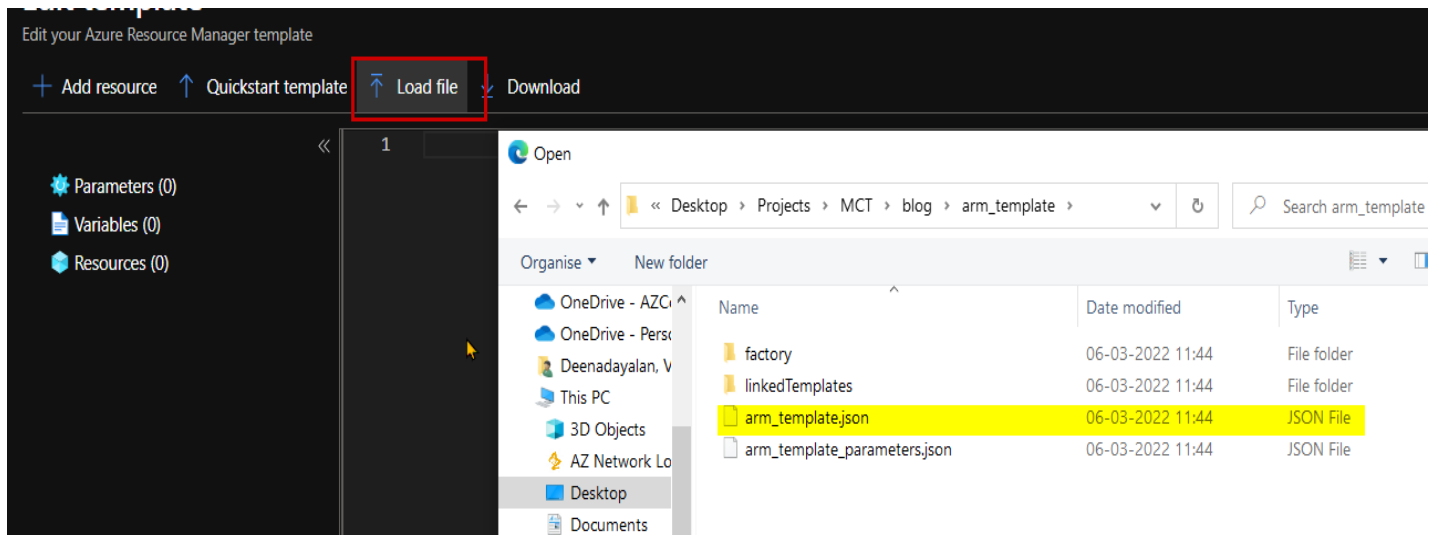
Determine which properties are parameterized when generating the ARM Template of this Data Factory.

[Edit](#)

Deploy resources with ARM templates, and you can edit the template in the editor.



In the next screen, you will see the edit template with a few lines of code, you can go ahead and delete that and click on 'Load file' option above to load the *arm\_template.json* file that we have extracted from the zip file.



I have modified the name of the file which contained the old name to '*adfdemo100222*' for identification.

```
1 {
2   "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
3   "contentVersion": "1.0.0.0",
4   "parameters": {
5     "factoryName": {
6       "type": "string",
7       "metadata": "Data Factory name",
8       "defaultValue": "adfdemo100222"
9     },
10    "AzureBlobStorage1_connectionString": {
11      "type": "secureString",
12      "metadata": "Secure string for 'connectionString' of 'AzureBlobStorage1'"
13    },
14    "AzureSqlDatabase1_connectionString": {
15      "type": "secureString",
16      "metadata": "Secure string for 'connectionString' of 'AzureSqlDatabase1'"
17    }
18  },
19  "variables": {
20    "factoryId": "[concat('Microsoft.DataFactory/factories/', parameters('factoryName'))]"
21  },
22  "resources": [
23    {
24      "name": "[concat(parameters('factoryName'), '/CopydatatoSQL')]",
25      "type": "Microsoft.DataFactory/factories/pipelines",
26      "apiVersion": "2018-06-01"
```

After modifying the name now save the file where will be required to enter the connection string details with the resource groups and region.



1. Data factory name
2. Connection string of blob storage where the sample files has been stored
3. Connection string of SQL database

## Custom deployment

Deploy from a custom template

**Basics** Review + create

**Template**

 Custom template  5 resources

[Edit template](#) [Edit parameters](#) [Visualize](#)

**Project details**

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ Visual Studio Enterprise Subscription

Resource group \* ⓘ SynResGroup100222

[Create new](#)

**Instance details**

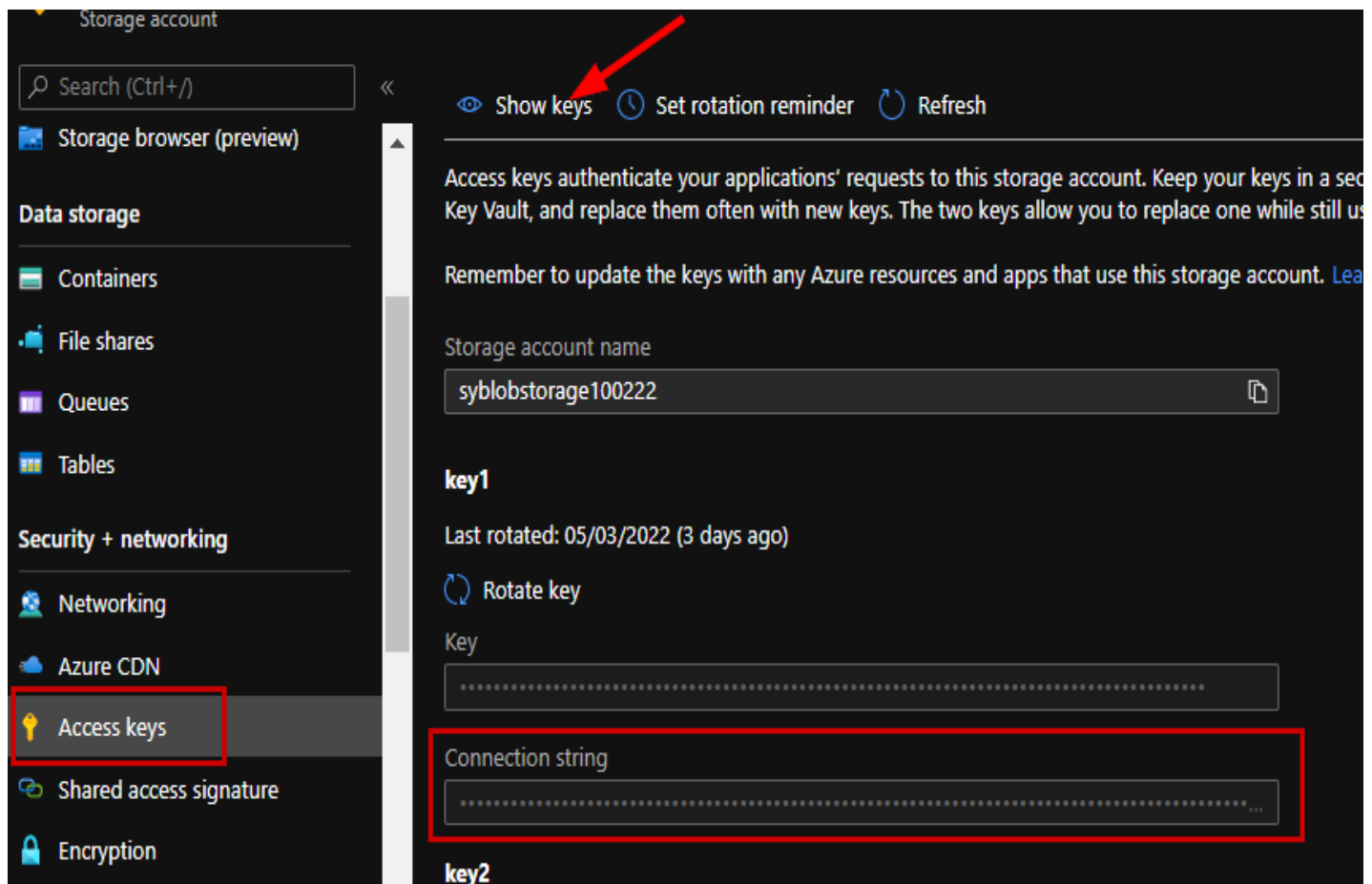
Region \* ⓘ (US) East US

Factory Name 1 adfdemo100222

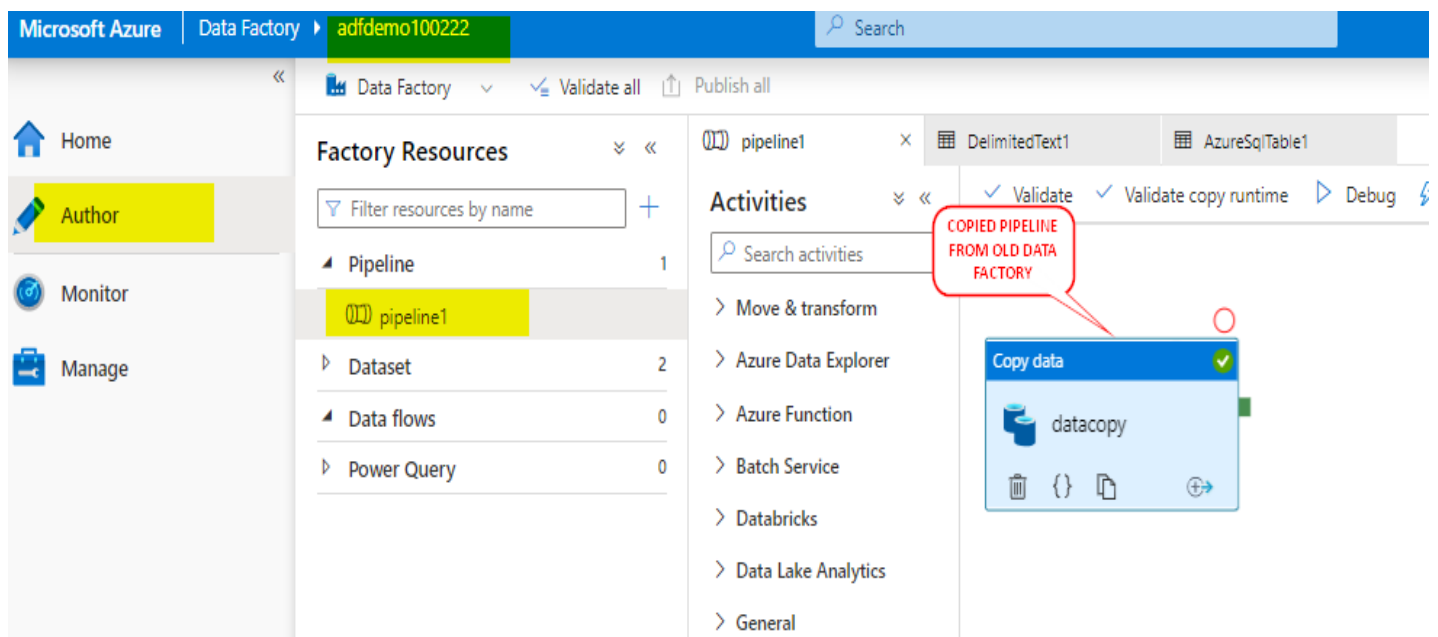
Azure Blob Storage1\_connection String 2 ..... ✓

Azure Sql Database1\_connection String 3 ..... ✓

For SQL database connection string details go to the database connection string tab in the left side pane. You can copy the string available in the ADO.NET tab and make sure you replace the password before pasting into the create deployment page.



Once done, go ahead and make sure it is successfully deployed.



We could see the copy of our restored pipeline from the old data factory instance. So you can go ahead to see how the restored pipeline is working.