Azure Data Factory

Lab: Self-Hosted Integration Runtime.

Pre-requisites:

- Azure Pass subscription
- Azure Data Lake Storage Gen2 storage account
- Azure Data Factory
- Windows 10 Virtual Machine

Lab Objective:

After completing this lab, you will be able to:

- Create Integration Runtime (Azure & Self Hosted)

Exercise: Create Self-Hosted Integration Runtime and copy the data from virtual machine to storage account.

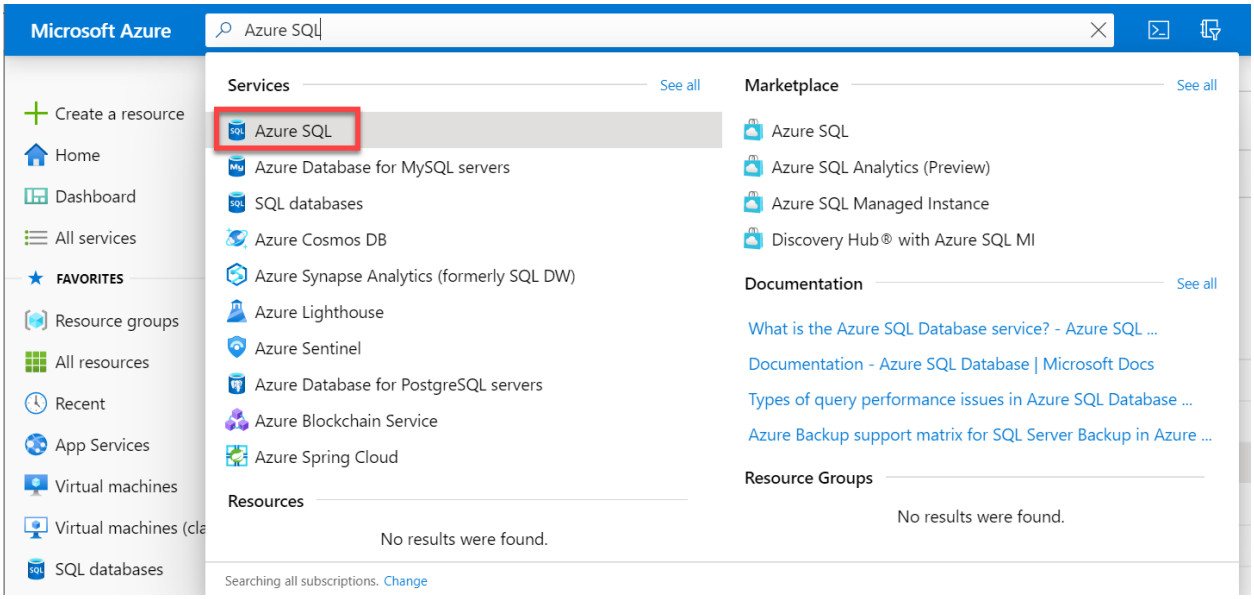
In this exercise you will be completing following task:

1. Provision a SQL Server on an Azure Virtual Machine
2. Connect to SQL Server on an Azure Virtual Machine
3. Create Self Hosted Integration Runtime in ADF
4. Copy data from SQL Server Database to Data Lake Storage

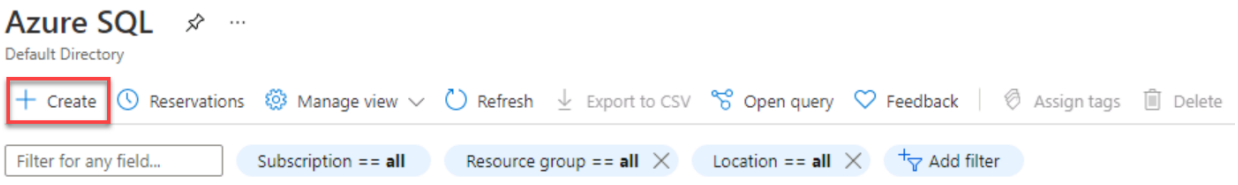
Task 1: Provision a SQL Server on an Azure Virtual Machine

Deploy a SQL Server on an Azure Virtual Machine

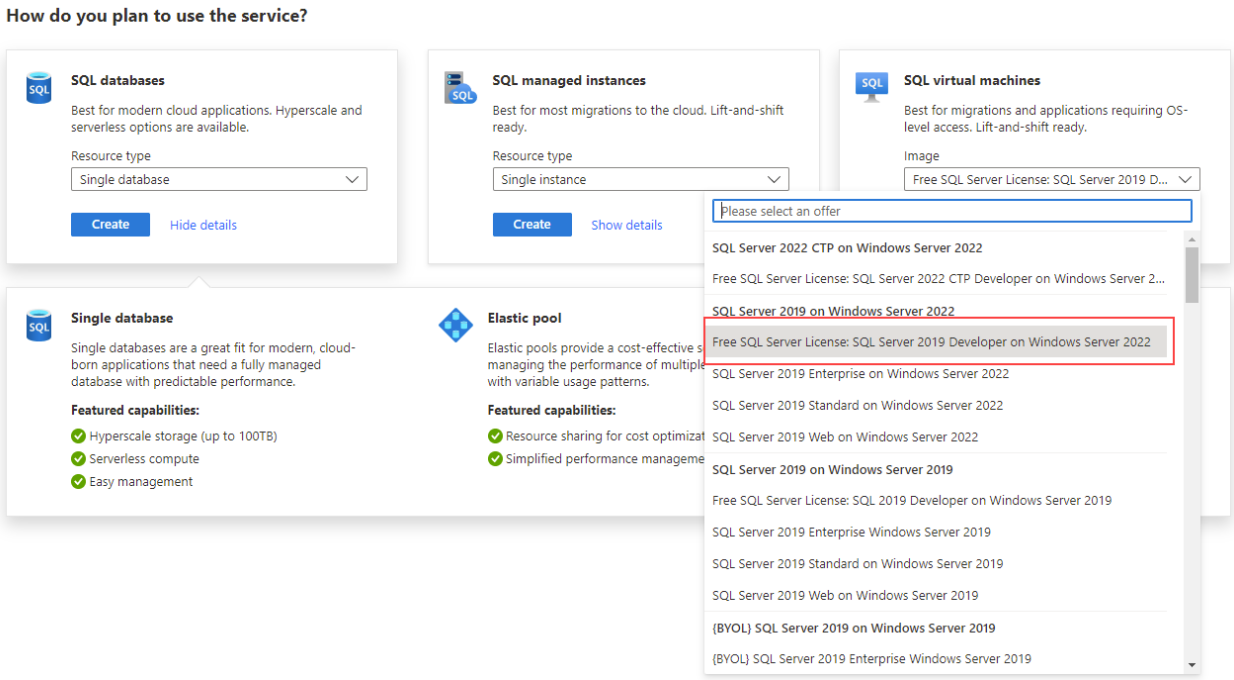
1. Locate the search bar at the top of the page. Search for **Azure SQL**. Select the search result for **Azure SQL** that appears in the results under **Services**.



2. On the **Azure SQL** blade, select **Create**.

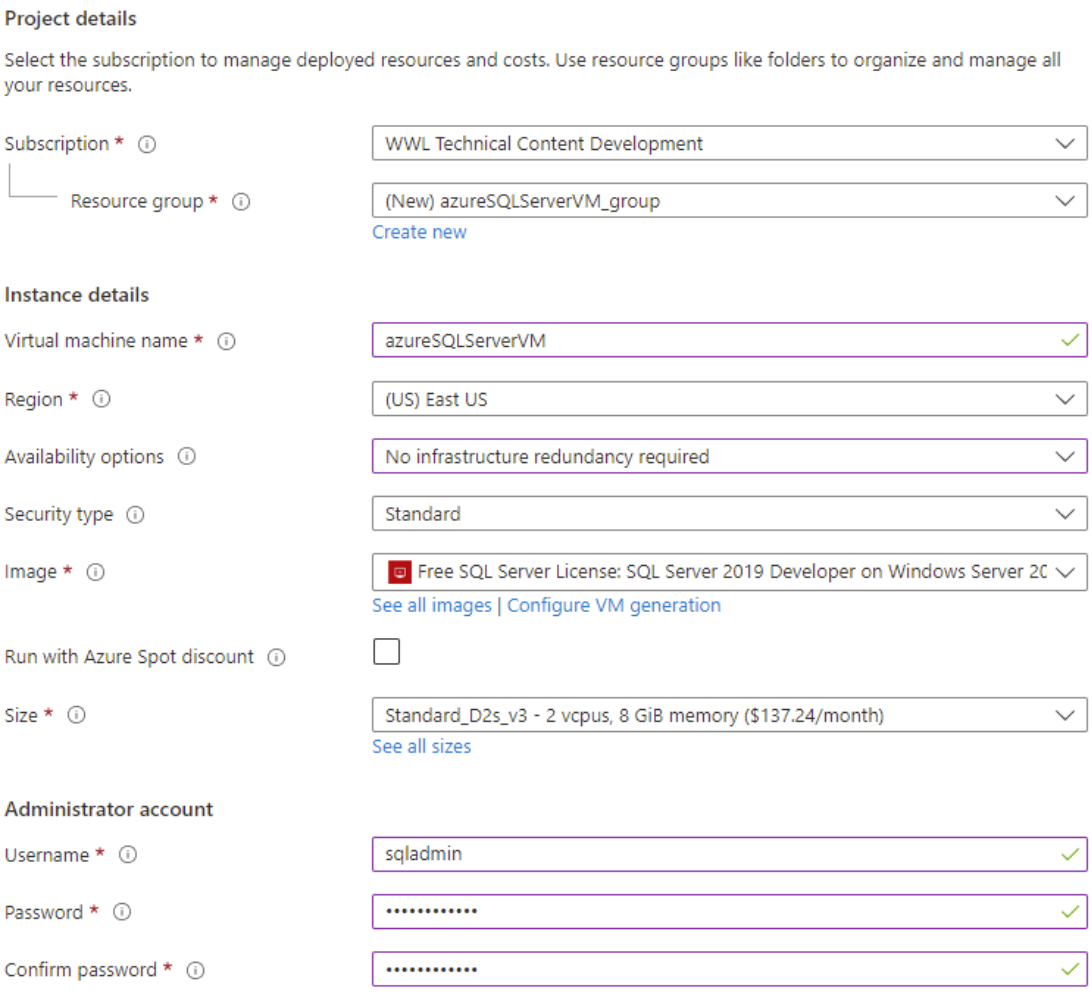


3. On the **Select SQL deployment option** blade, click on the drop-down box under **SQL virtual machines**. Select the option labeled **Free SQL Server License: SQL 2019 Developer on Windows Server 2022**. Then select **Create**.



4. On the **Create a virtual machine** page, enter the following information:
    - **Subscription:** <Your subscription>
    - **Resource group:** SQL-VM-RG
    - **Virtual machine name:** sqlvm
    - **Region:** <your local region, same as the selected region for your resource group>
    - **Availability Options:** No infrastructure redundancy required
    - **Image:** Free SQL Server License: SQL 2019 Developer on Windows Server 2022 - Gen1
    - **Azure spot instance:** No (unchecked)
    - **Size:** Standard *D2s_v3* (2 vCPUs, 8 GiB memory). You may need to select the "See all sizes" link to see this option)
    - **Administrator account username:** sqladmin
    - **Administrator account password:** pwd!DP300lab01 (or your own password that meets the criteria)
    - **Select inbound ports:** RDP (3389)
    - **Would you like to use an existing Windows Server license?:** No (unchecked)

Make note of the username and password for later use.



5. Navigate to the **Management** tab and review the configuration.

   Verify that **Enable auto_shutdown** is unchecked.

6. Navigate to the **SQL Server settings** tab and review the configuration.

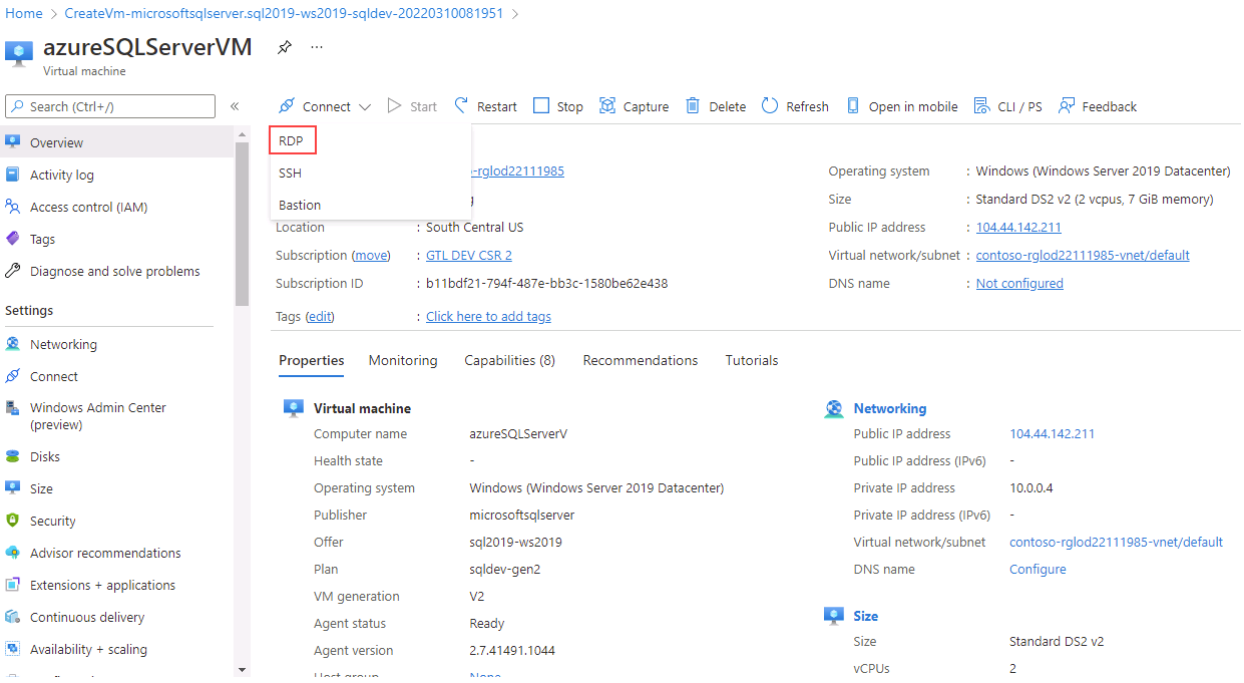
   Enable SQL Authentication

7. Select the **Review + create** button. Then select **Create**.
8. On the deployment blade, wait until the deployment is complete. The VM will take approximate 5-10 minutes to deploy. After the deployment is complete, select **Go to resource**.

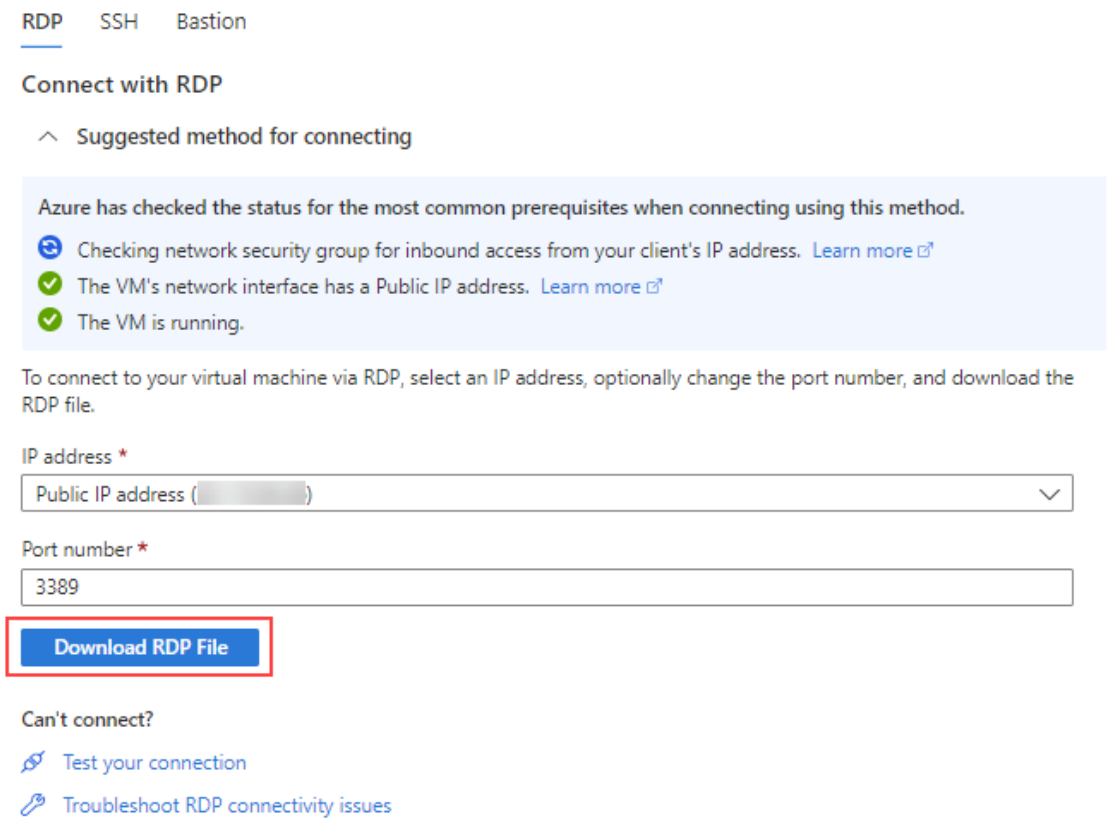
   Note: Your deployment may take several minutes to complete.

Task 2: Connect to SQL Server on an Azure Virtual Machine

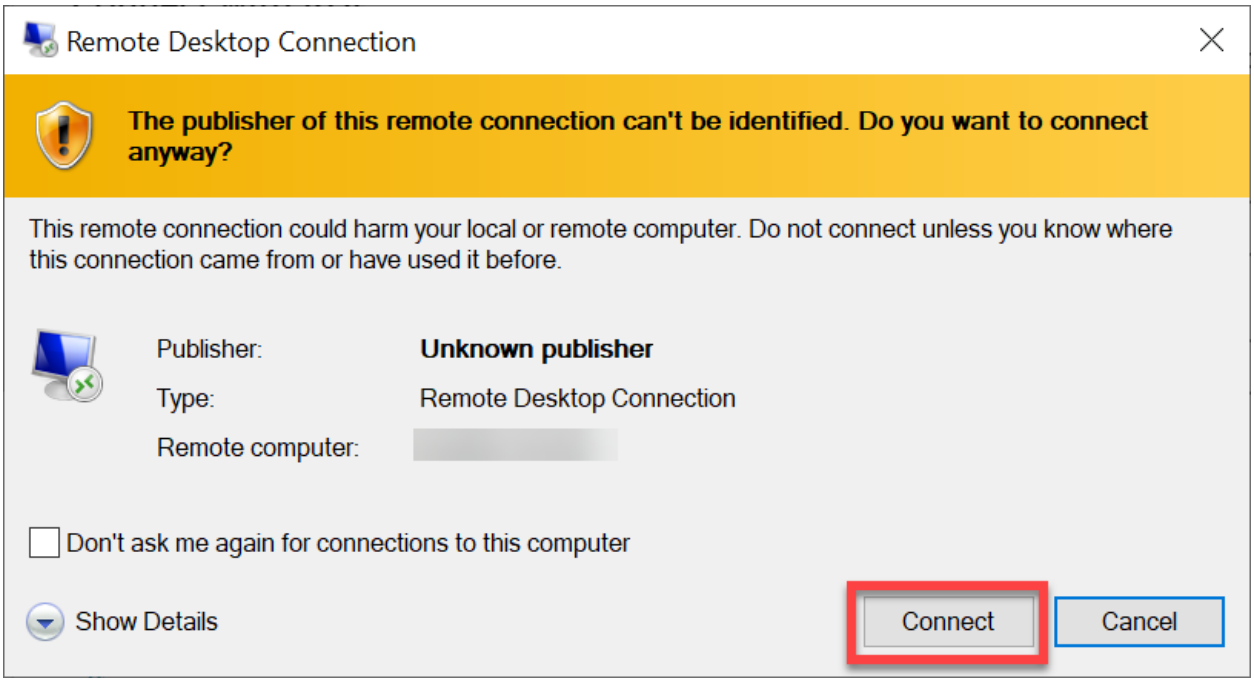
1. On the **Overview** page for the virtual machine, select the **Connect** button and choose RDP.



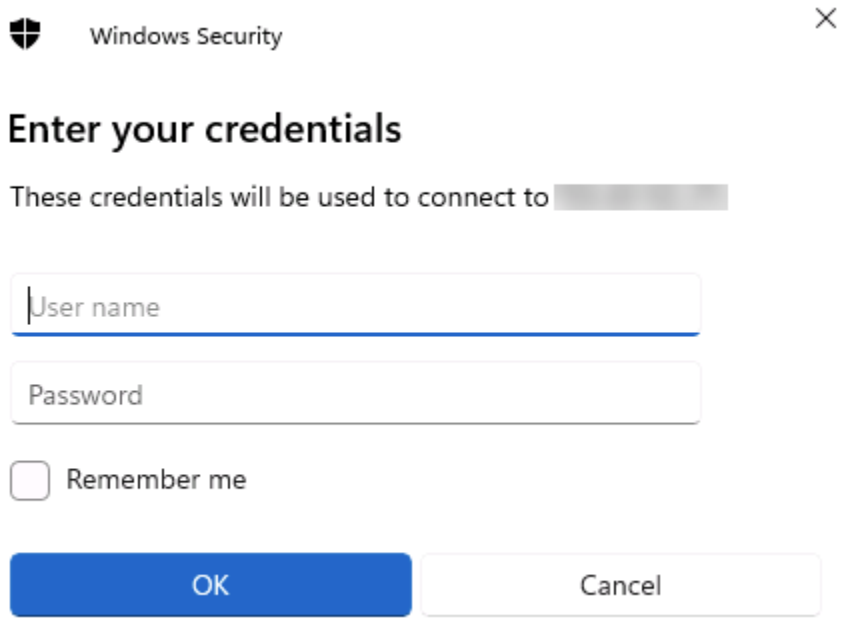
2. On the RDP tab, select the **Download RDP File** button.



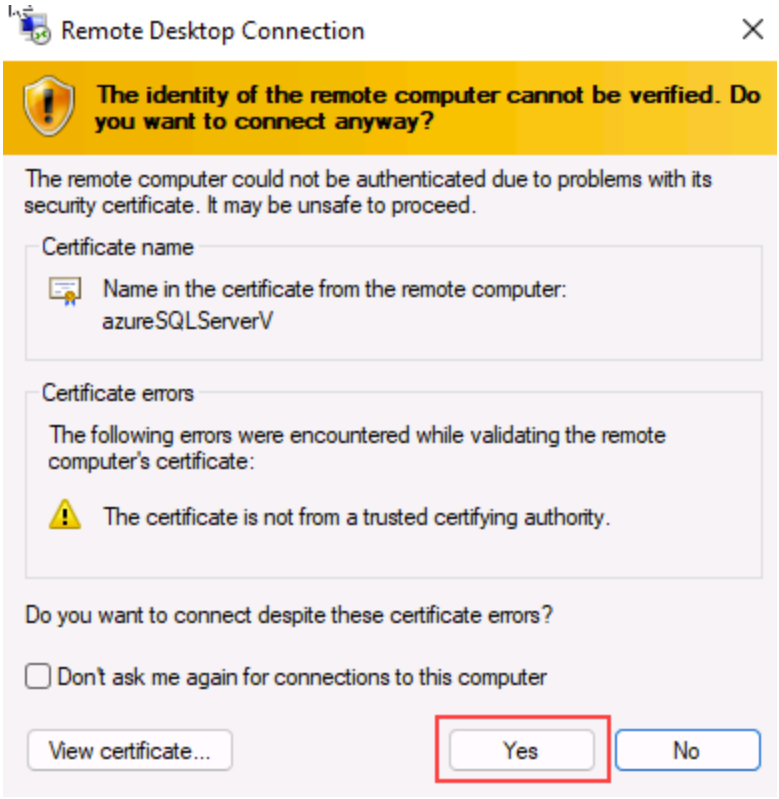
3. Open the RDP file that was just downloaded. When a dialog appears asking if you want to connect, select **Connect**.



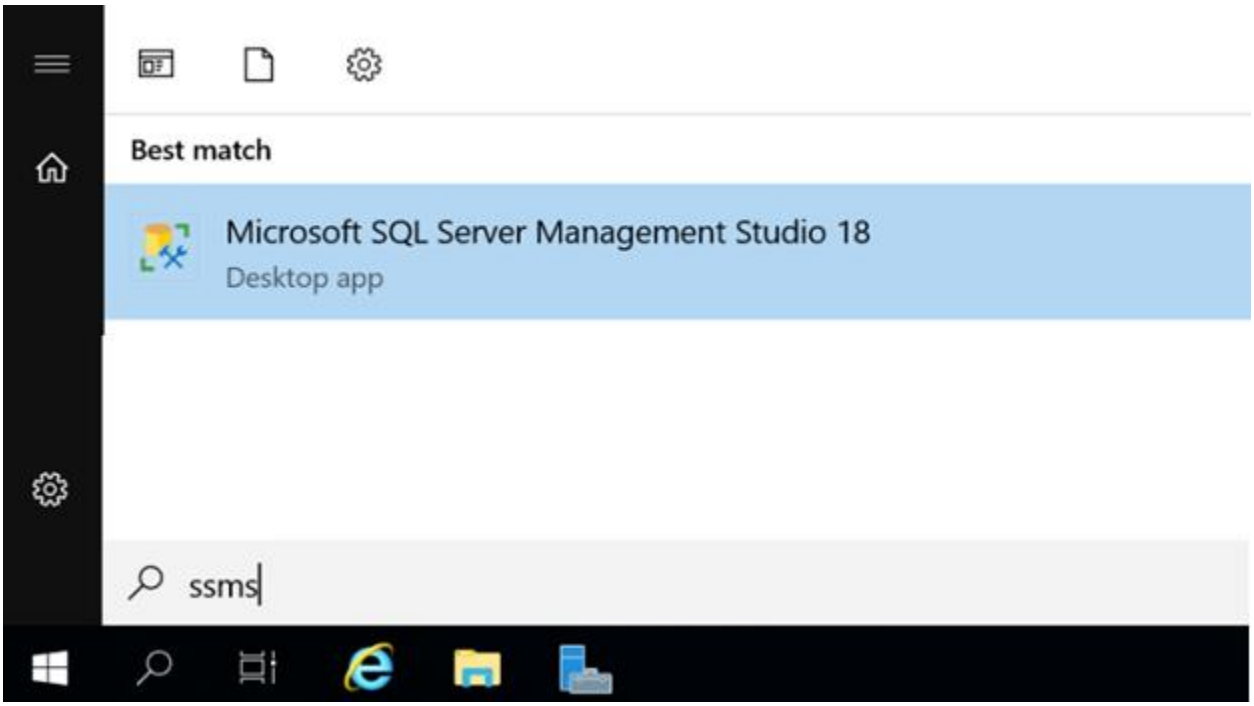
4. Enter the username and password selected during the virtual machine provisioning process. Then select **OK**.



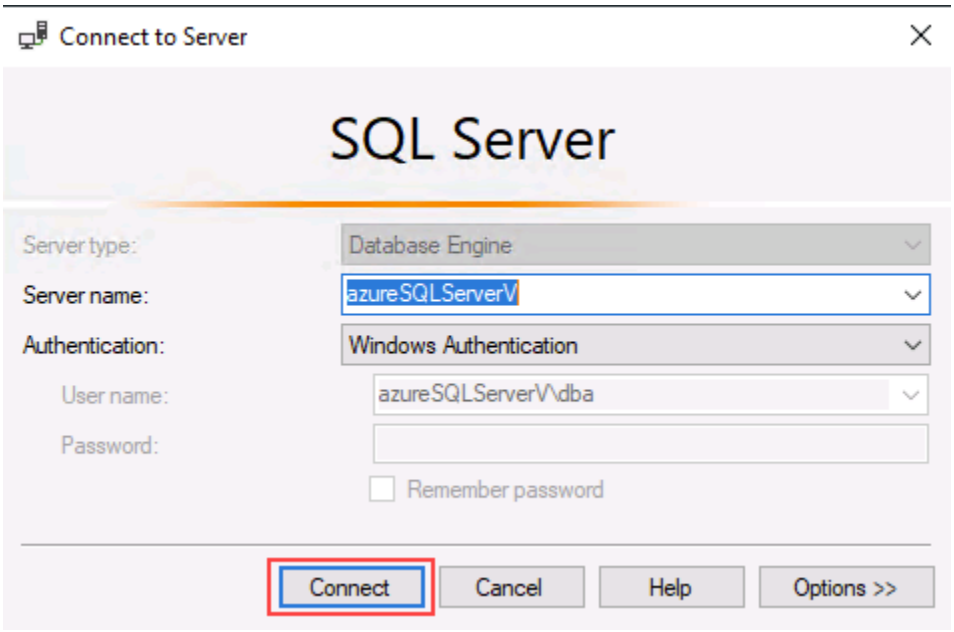
5. When the **Remote Desktop Connection** dialog appears asking if you want to connect, select **Yes**.



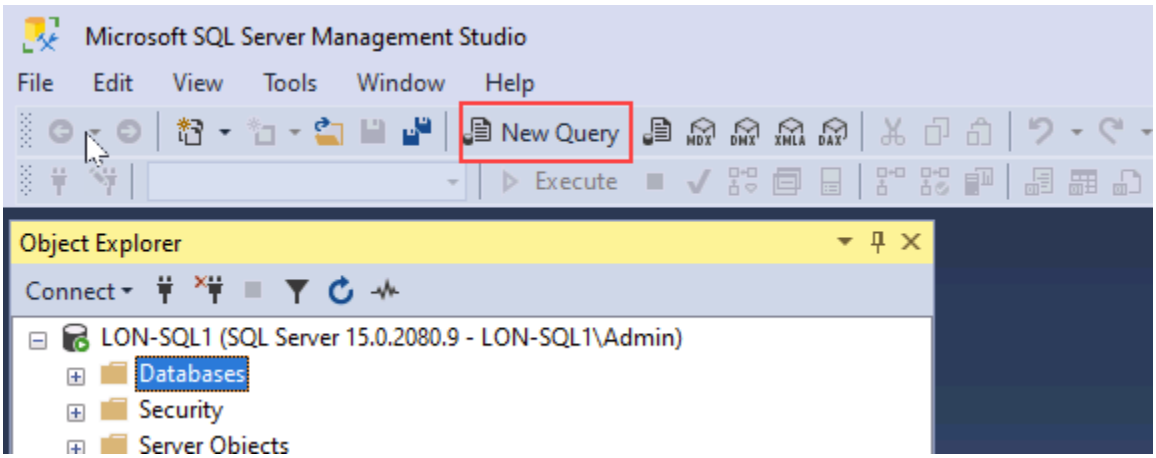
6. Select the Windows Start button and type SSMS. Select **Microsoft SQL Server Management Studio 18** from the list.



7. When SSMS opens, notice that the **Connect to Server** dialog will be pre-populated with the default instance name. Select **Connect**.



8. Select the **Databases** folder, and then **New Query**.



9. Copy the AdventureWorks2017 file from the URL:

   https://github.com/MicrosoftLearning/dp-300-database-administrator/blob/master/Instructions/Templates/AdventureWorks2017.bak

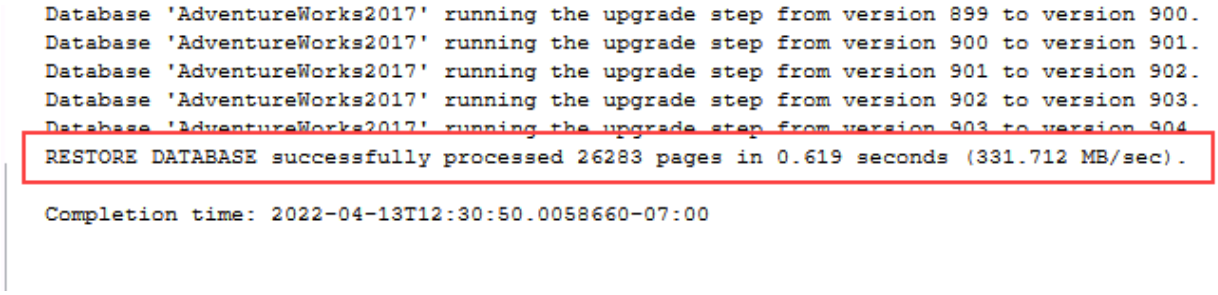
   and store it at C:\LabFiles Location. (Create Directory LabFiles in c: drive)

10. In the new query window, copy and paste the below T-SQL into it. Execute the query to restore the database.

```
RESTORE DATABASE AdventureWorks2017
FROM DISK = 'C:\LabFiles\AdventureWorks2017.bak'
WITH RECOVERY,
      MOVE 'AdventureWorks2017'
        TO 'C:\LabFiles\AdventureWorks2017.mdf',
      MOVE 'AdventureWorks2017_log'
        TO 'C:\LabFiles\AdventureWorks2017_log.ldf';
```

Note: The database backup file name and path should match with what you've downloaded on step 1, otherwise the command will fail.

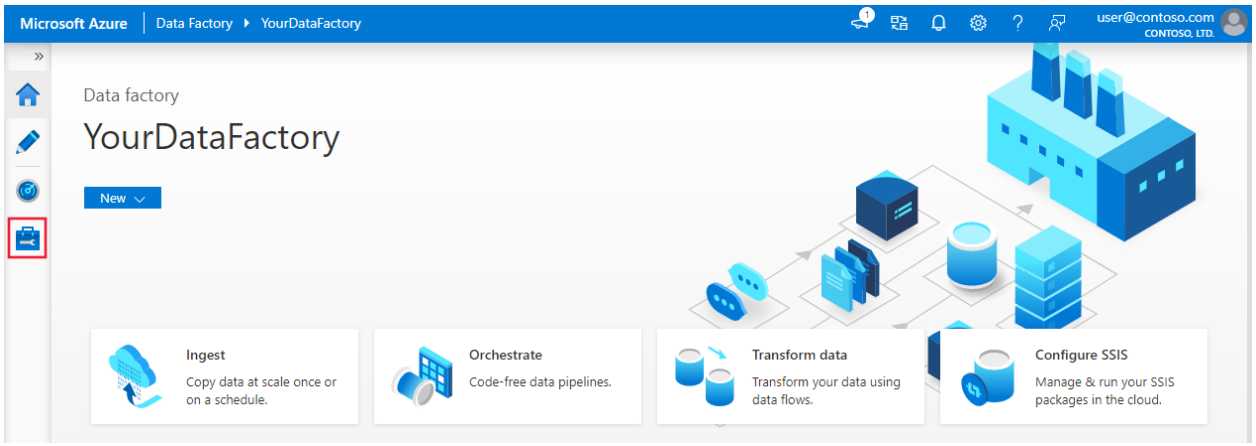
11. You should see a successful message after the restore is complete.



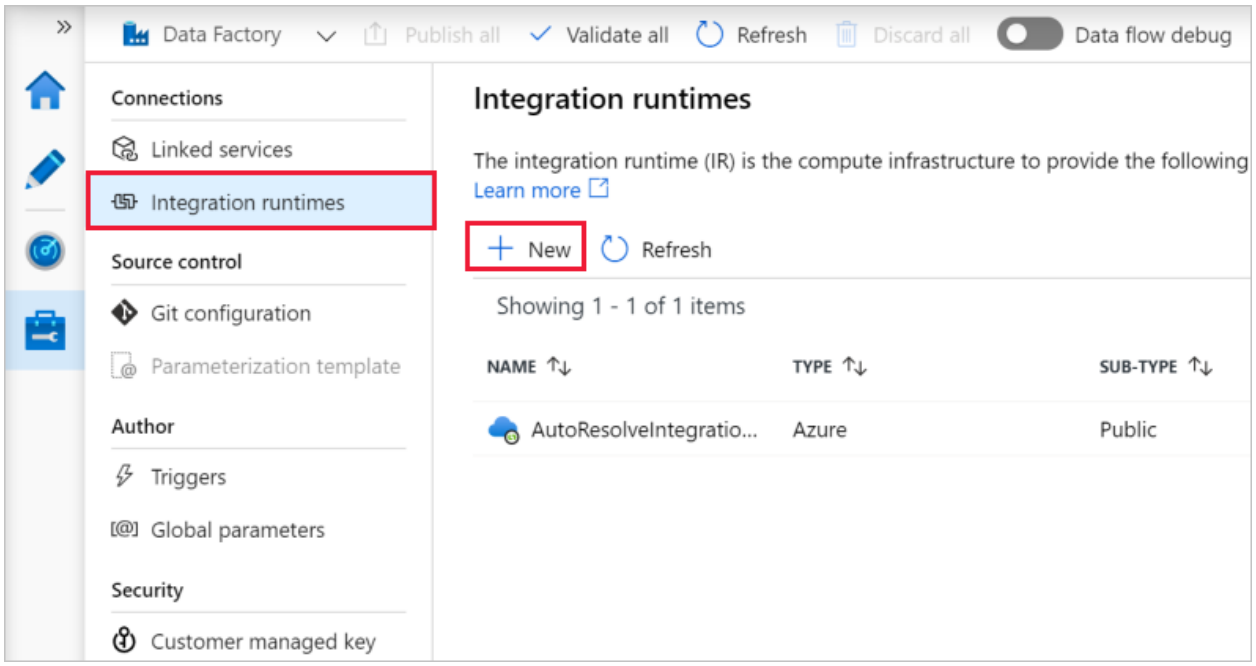
Task 3: Create Self Hosted Integration Runtime in ADF

Use the following steps to create a self-hosted IR using the Azure Data Factory UI.

1. On the home page of the Azure Data Factory UI, select the Manage tab from the leftmost pane.



2. Select **Integration runtimes** on the left pane, and then select **+New**.



3. On the **Integration runtime setup** page, select **Azure, Self-Hosted**, and then select **Continue**.
4. On the following page, select **Self-Hosted** to create a Self-Hosted IR, and then Select **Continue.**

Integration runtime setup

Network environment:

Choose the network environment of the data source / destination or external compute to which the integration runtime will connect to for data flows, data movement or dispatch activities:

Azure

Use this for running data flows, data movement, external and pipeline activities in a fully managed, serverless compute in Azure.

Self-Hosted

Use this for running activities in an on-premise / private network

View more

External Resources:

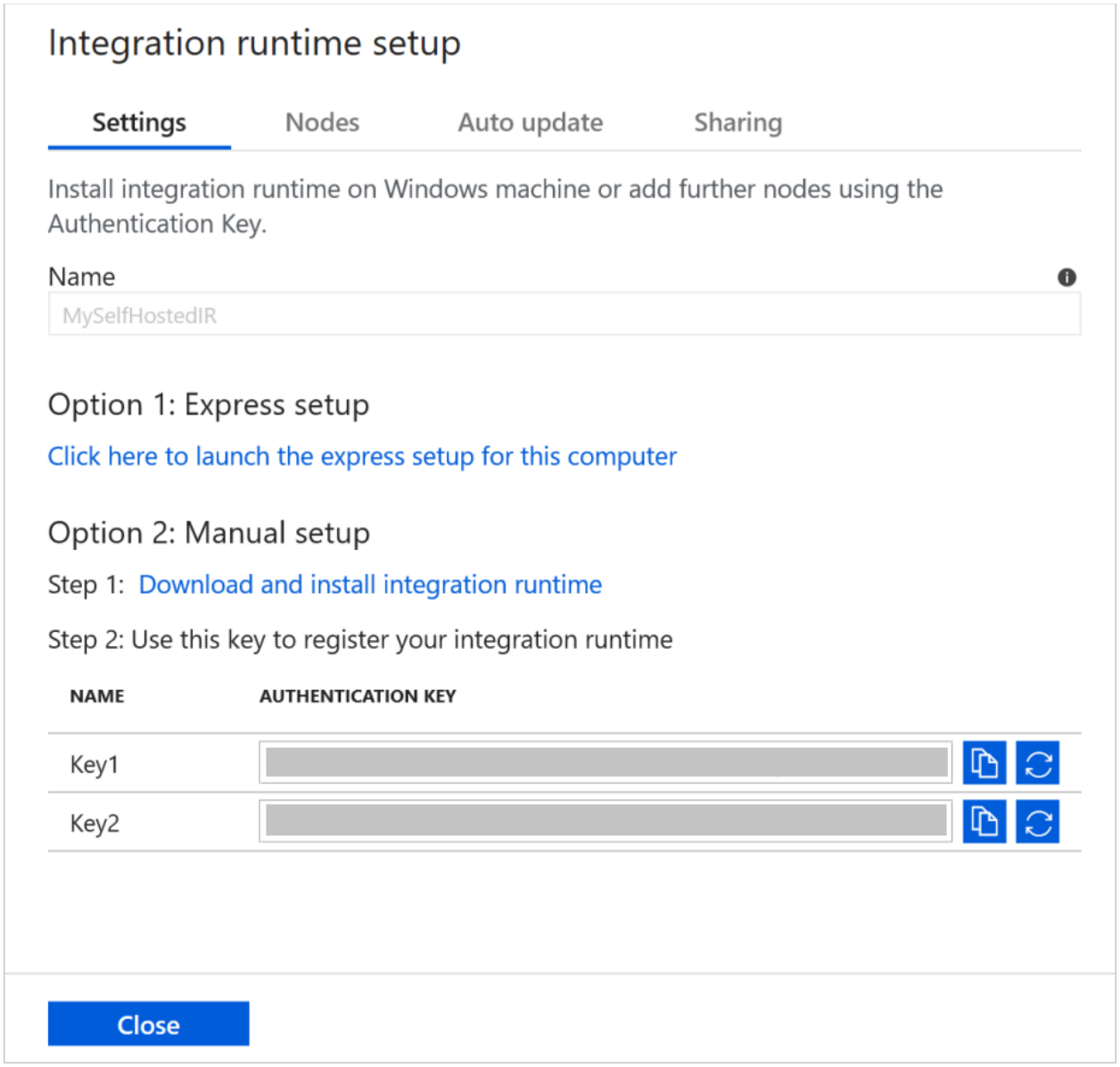
You can use an existing self-hosted integration runtime that exists in another resource. This way you can reuse your existing infrastructure where self-hosted integration runtime is setup.

Linked Self-Hosted

Learn more

Continue | Back | Cancel

5. Enter a name for your IR (SH-IR), and select **Create**.
6. On the **Integration runtime setup** page, select **Option 2** to set up manually.
7. The following instructions are based on manual setup:

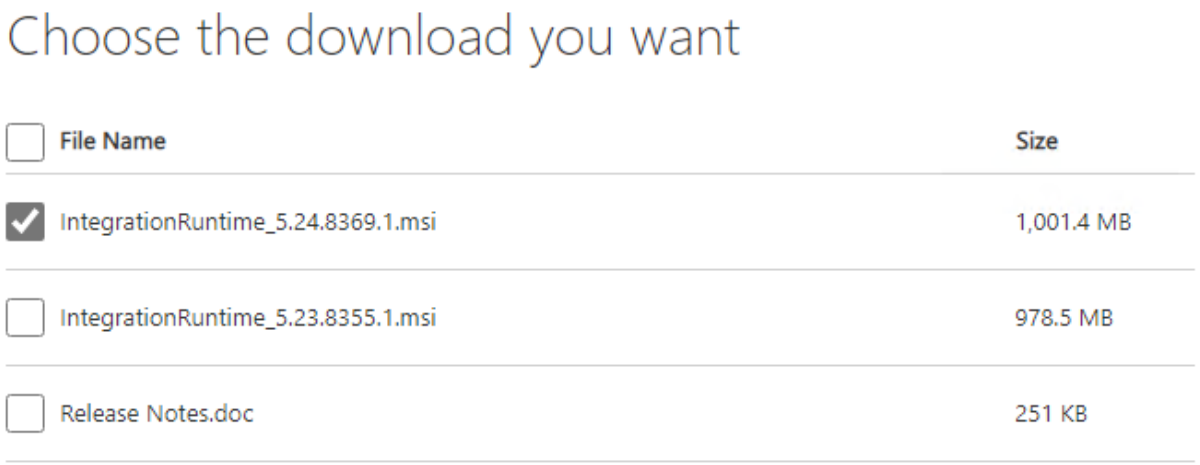


a. Copy and paste the authentication key. Select **Download and install integration runtime**.

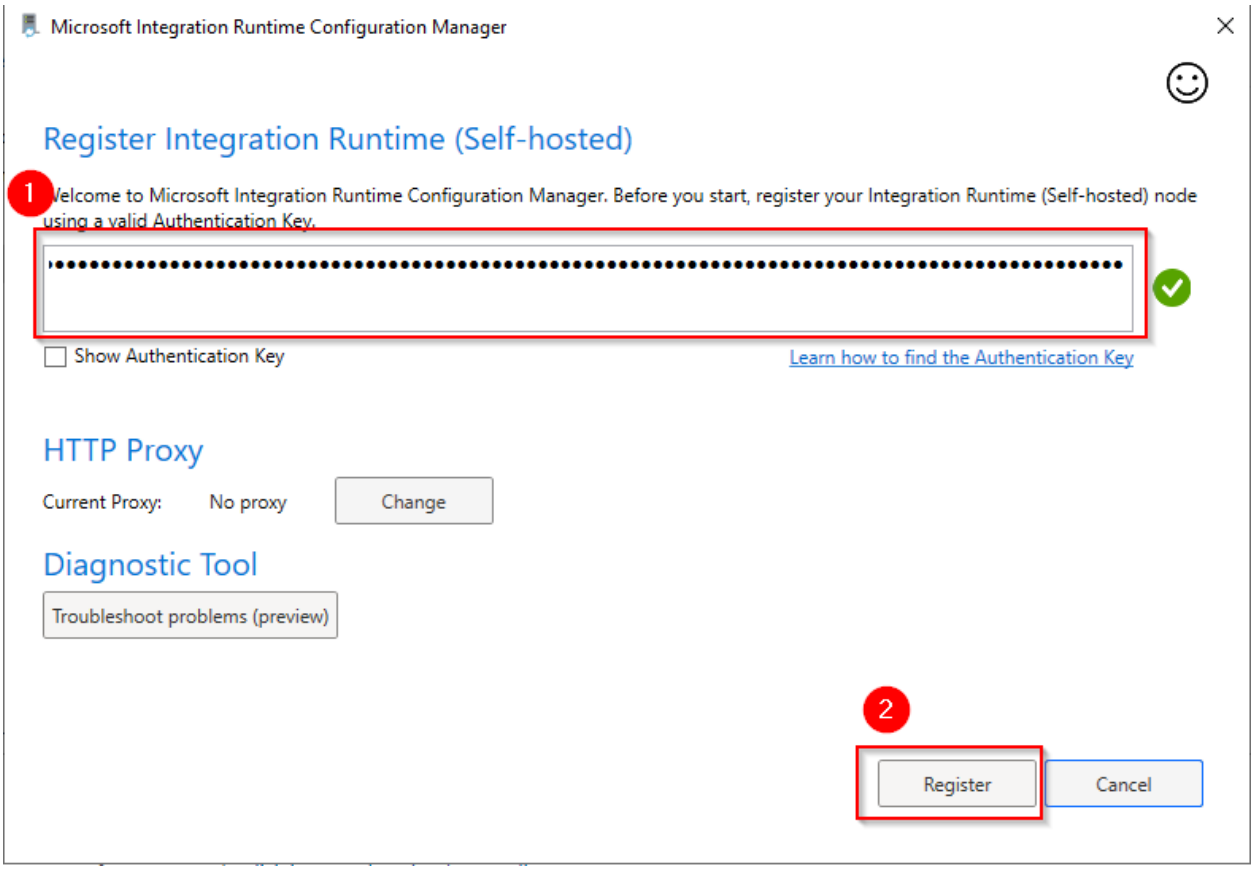
b. Download the self-hosted integration runtime on a **SQL Server Virtual Machine**. Run the installer.

   Or Copy this URL and download in the SQL Server VM:

   https://www.microsoft.com/en-us/download/details.aspx?id=39717

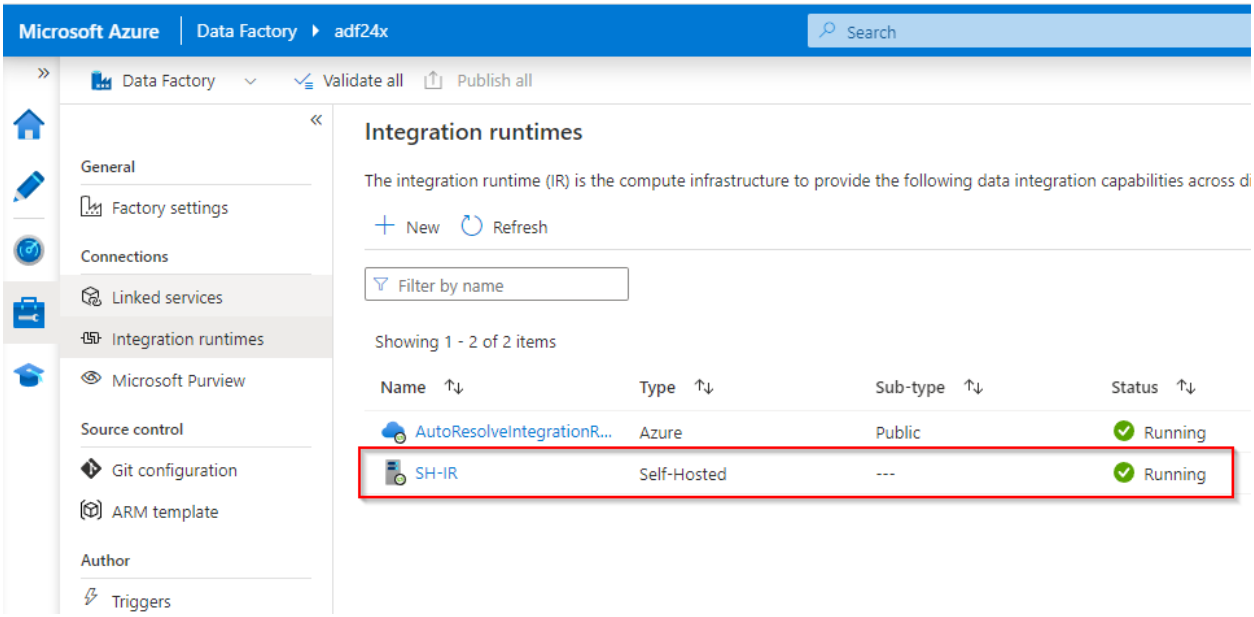


c. On the **Register Integration Runtime (Self-hosted)** page, paste the key you saved earlier, and select **Register**.



d. On the **New Integration Runtime (Self-hosted) Node** page, select **Finish**.

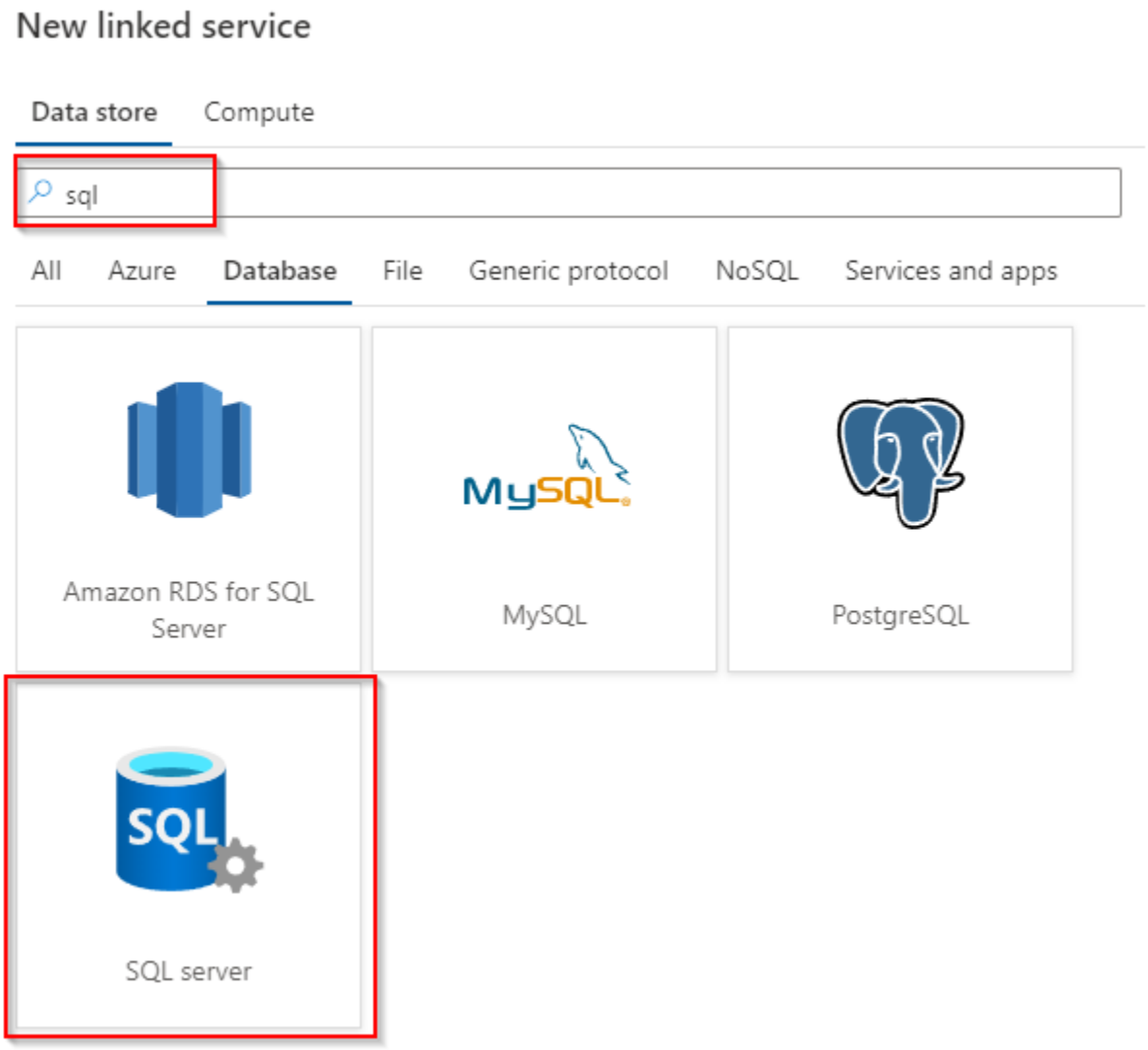
8. After the self-hosted integration runtime is registered successfully, you see the following window:


Microsoft Azure
Data Factory
adf24x
Search
Data Factory
Validate all
Publish all
General
Factory settings
Connections
Linked services
Integration runtimes
Microsoft Purview
Source control
Git configuration
ARM template
Author
Triggers
Integration runtimes
The integration runtime (IR) is the compute infrastructure to provide the following data integration capabilities across di
New
Refresh
Filter by name
Showing 1 - 2 of 2 items
Name
Type
Sub-type
Status
AutoResolveIntegrationR...
Azure
Public
Running
SH-IR
Self-Hosted

Running

Task 4: Copy data from SQL Server Database Table to Data Lake Storage

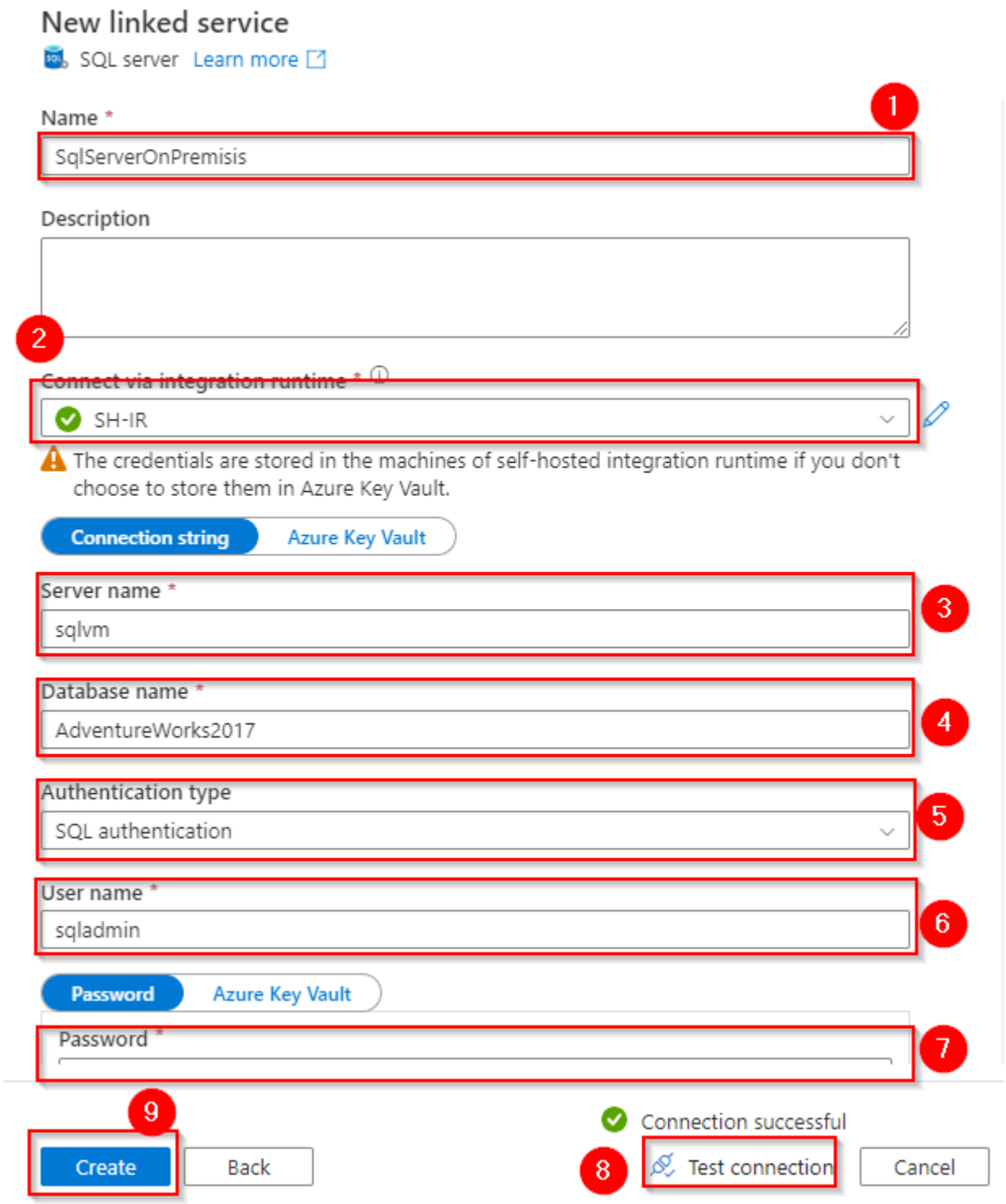
1. On the home page of the Azure Data Factory UI, select the Manage tab from the leftmost pane and select **Linked Service.**
2. Click + New to create a linked services and search sql to select the SQL Server and click continue.



3. Fill the following details

    a. Name: SqlServerOnPremises
    b. Connect via integration runtime: SH-IR
    c. Server name: sqlvm
    d. Database name: c

e. Authentication Type: SQL Authentication
f. Username: sqladmin
g. Password: pwd!DP300lab01 (or your own password that you setup at the time of VM creation)

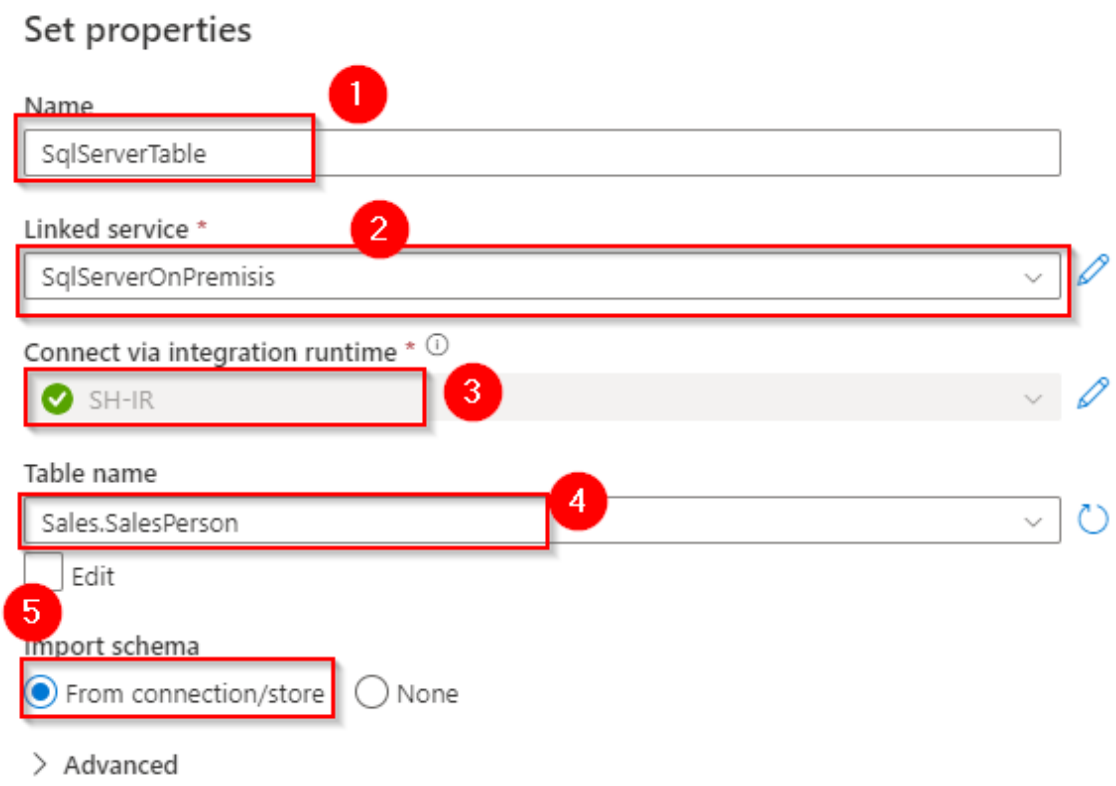


4. Create a new pipeline, Author Tab -> + -> Pipeline

5. Drag copy activity from Activities canvas under move and transform section to pipeline surface area.

6. In source tab of copy activity create new dataset as shown.

   Name: SqlServerTable
   Linked service: SqlServerOnPremises
   Table Name: Sales.SalesPerson

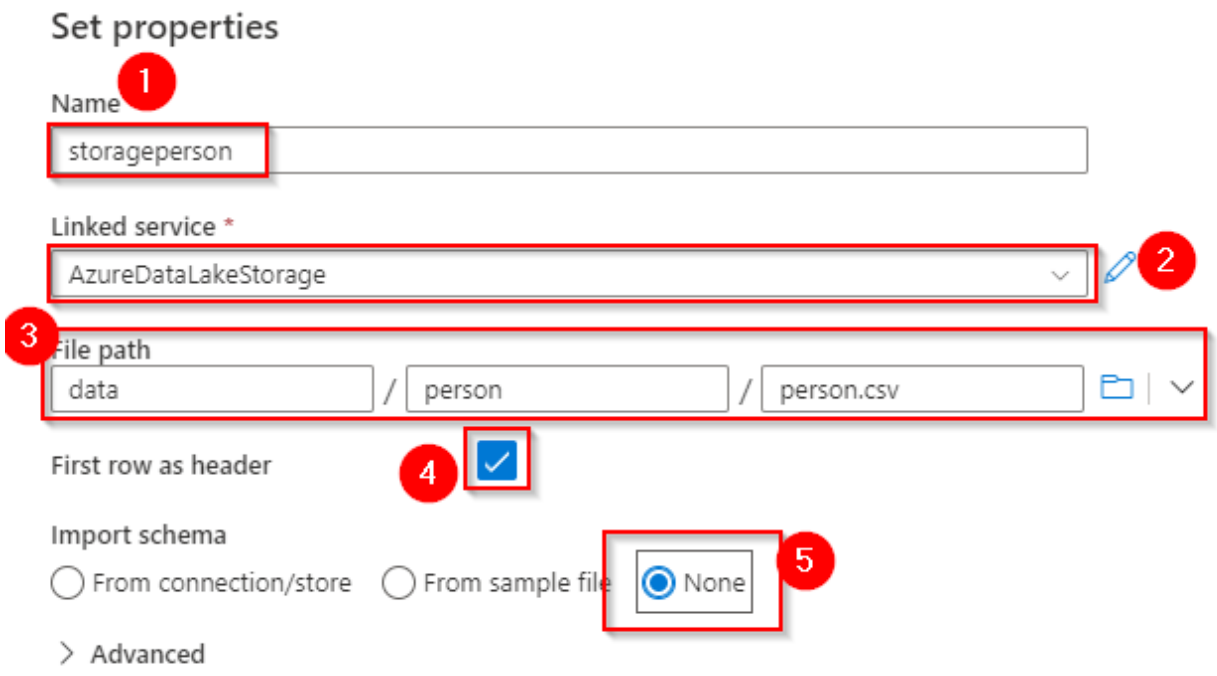


7. Switch to Sink tab and create a new dataset for data lake gen 2 storage account and follow the below steps:

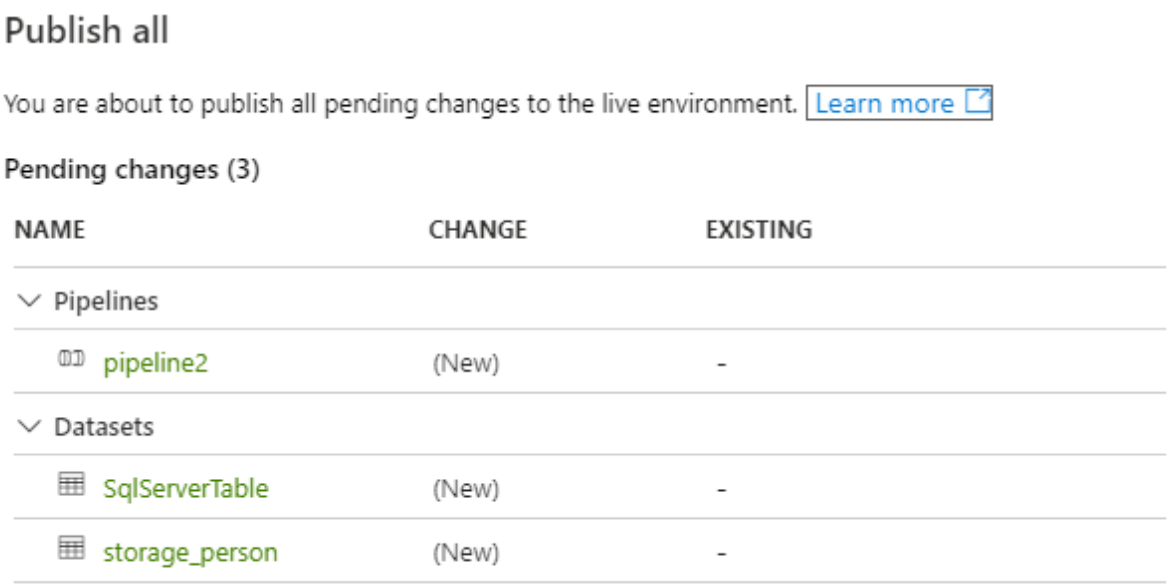
   a. Select format: DelimitedText

   b. Set Properties

      Name: storage_person
      Linked Service: AzureDataLakeStorage
      File Path: data/person/person.csv
      First row as header: checked
      Import Schema: None



8. Click Publish all.



9. Run the pipeline. After execution finish verify the file in the storage account.