Lab 1 – Create Azure resources

Welcome to Lab 1! To complete this lab series, you're going to need a few things.

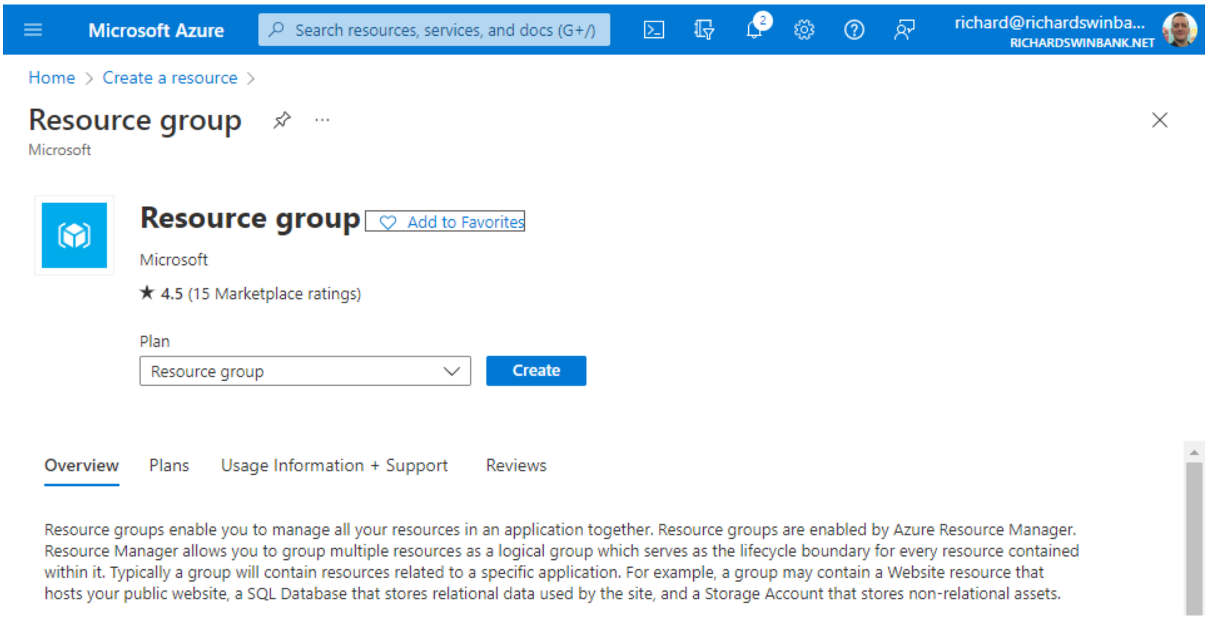
- A web browser compatible with Azure Data Factory (ADF) – use Microsoft Edge or Google Chrome.
- An Azure subscription – if you don't have one, you can sign up for a free trial at https://azure.microsoft.com/en-gb/free/. Whichever subscription you use, you'll need enough access to create resources in it.
- A GitHub repository where you can store lab code – if you don't already have a GitHub account, sign up for one at https://github.com/signup. (We use GitHub for these labs, but Azure Data Factory also supports Azure DevOps Repos).

Lab 1.1 – Create a resource group

Resource groups are logical containers for resources in Azure. In this lab you will create a resource group to contain all the resources you create in later labs. This will make cleaning up easier – when you've finished all the labs, you can just delete the resource group.

1. In the Azure Portal (https://portal.azure.com/) click "Create a resource" and use the "Search services and marketplace" textbox to search for "Resource group".

2. On the "Resource group" overview, click "Create".



Resource groups enable you to manage all your resources in an application together. Resource groups are enabled by Azure Resource Manager. Resource Manager allows you to group multiple resources as a logical group which serves as the lifecycle boundary for every resource contained within it. Typically a group will contain resources related to a specific application. For example, a group may contain a Website resource that hosts your public website, a SQL Database that stores relational data used by the site, and a Storage Account that stores non-relational assets.

3. Give the resource group a name, and choose the Region geographically closest to you.

4. Click "Review + create", then "Create".

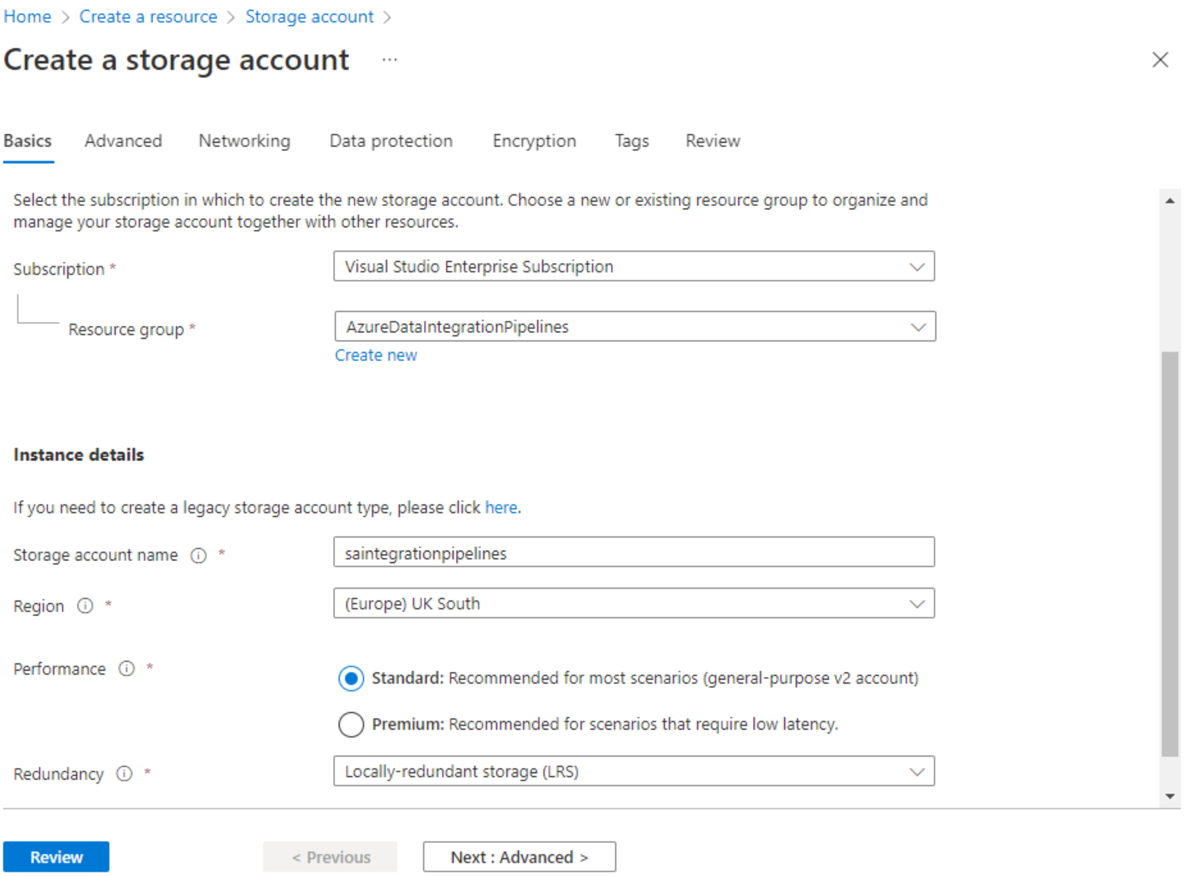
Lab 1.2 – Create data lake storage

Data lake storage is blob storage in an Azure storage account, with one particularly important feature: hierarchical namespaces are **enabled**. This makes certain file operations – renaming file folders, for example – much more efficient.

1. In the portal, click “Create a resource” and search for “Storage account”. Click “Create” on the overview screen.

2. Complete the “Basics” tab like this:

   - Choose your subscription and the resource group you created in Lab 1.1.
   - Enter a storage account name – this must be globally unique (across the entire Azure platform).
   - Choose the same location you specified for your resource group. Having storage located close to you makes data movement faster and cheaper.
   - Choose redundancy option “Locally-redundant storage”. This is nice and cheap for lab work, but you’ll want something more resilient in a production environment!



Create a storage account

Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.

Subscription *: Visual Studio Enterprise Subscription

Resource group *: AzureDataIntegrationPipelines

Instance details

If you need to create a legacy storage account type, please click here.

Storage account name *: saintegrationpipelines

Region *: (Europe) UK South

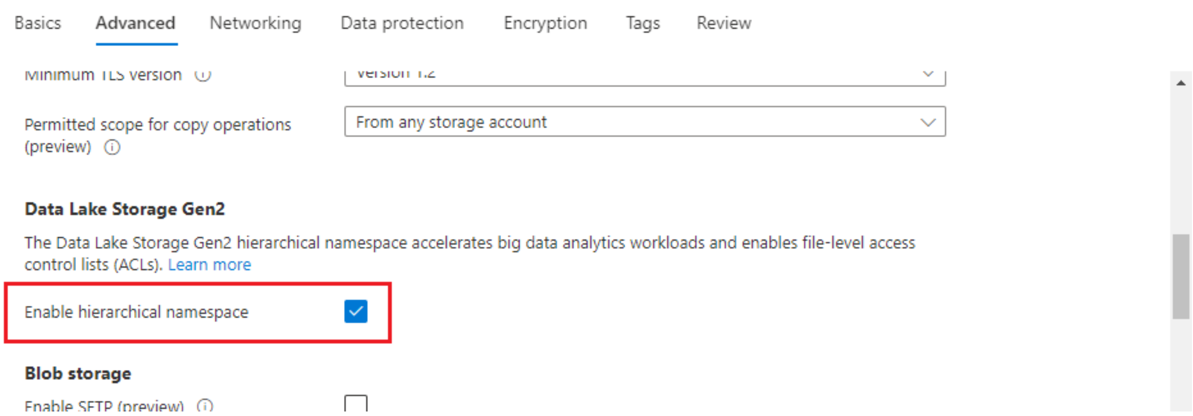
Performance *:

● **Standard:** Recommended for most scenarios (general-purpose v2 account)

○ **Premium:** Recommended for scenarios that require low latency.

Redundancy *: Locally-redundant storage (LRS)

3. On the “Advanced” tab, tick the “Enable hierarchical namespace” checkbox. This step is **essential** to make the storage account a data lake.



Basics | Advanced | Networking | Data protection | Encryption | Tags | Review

Minimum TLS version ⓘ — Version 1.2

Permitted scope for copy operations (preview) ⓘ — From any storage account

Data Lake Storage Gen2

The Data Lake Storage Gen2 hierarchical namespace accelerates big data analytics workloads and enables file-level access control lists (ACLs). Learn more

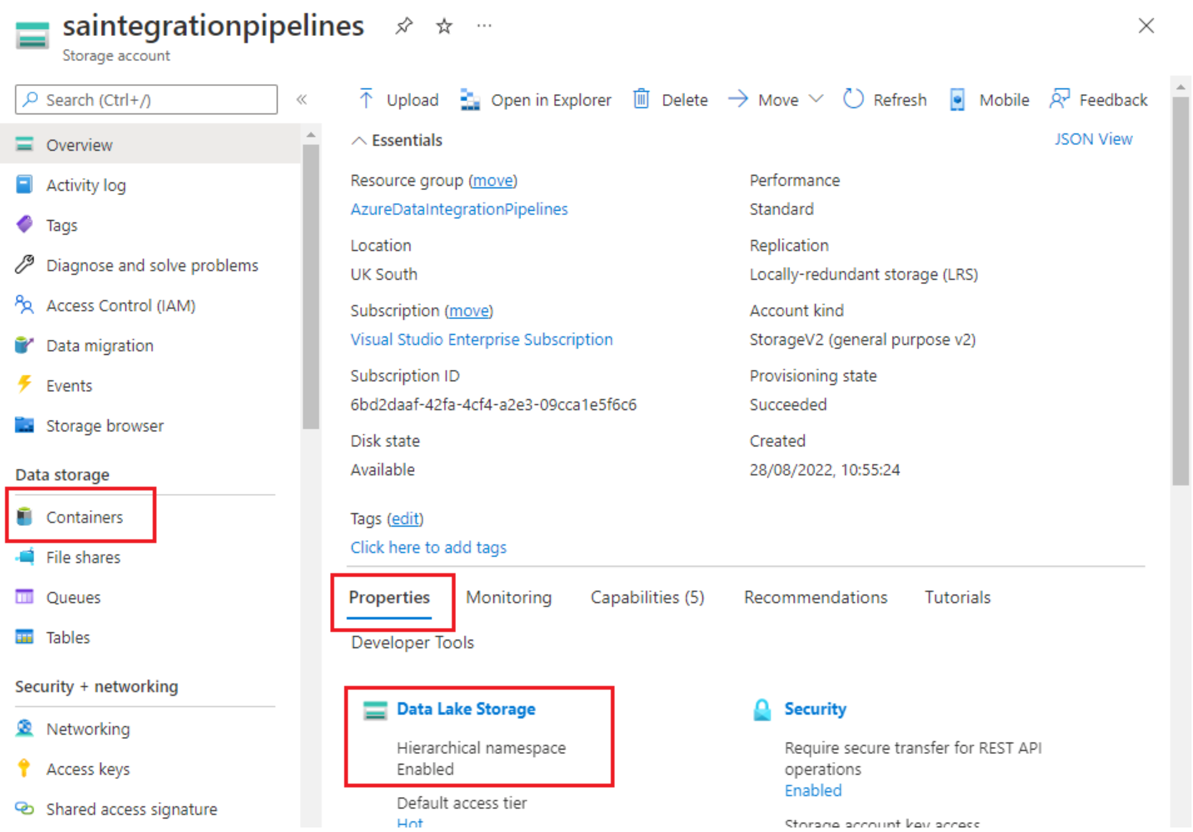
Enable hierarchical namespace ☑

Blob storage

Enable SFTP (preview) ⓘ ☐

4. Click “Review” to accept defaults on the remaining tabs and skip to the “Review” tab. Click “Create”. When the data lake finishes deploying (this may take a couple of minutes), click on “Go to resource”.

   The “Storage account” blade contains an “Essentials” section above a tabbed pane. The entry in the top left of the “Properties” tab should read “Data Lake Storage”:



saintegrationpipelines

Storage account

Search (Ctrl+/)

Upload | Open in Explorer | Delete | Move | Refresh | Mobile | Feedback

Overview
Activity log
Tags
Diagnose and solve problems
Access Control (IAM)
Data migration
Events
Storage browser

Data storage
Containers
File shares
Queues
Tables

Security + networking
Networking
Access keys
Shared access signature

Essentials — JSON View

Resource group (move) AzureDataIntegrationPipelines	Performance Standard
Location UK South	Replication Locally-redundant storage (LRS)
Subscription (move) Visual Studio Enterprise Subscription	Account kind StorageV2 (general purpose v2)
Subscription ID 6bd2daaf-42fa-4cf4-a2e3-09cca1e5f6c6	Provisioning state Succeeded
Disk state Available	Created 28/08/2022, 10:55:24

Tags (edit)
Click here to add tags

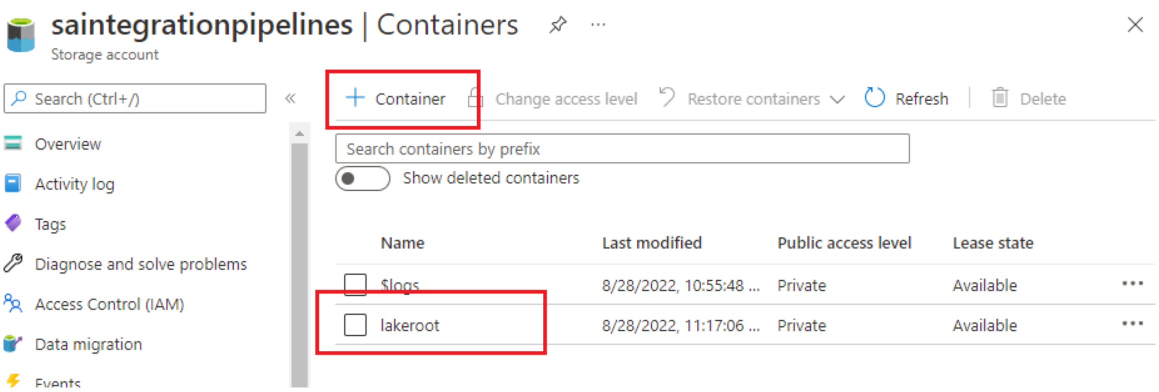
Properties | Monitoring | Capabilities (5) | Recommendations | Tutorials | Developer Tools

Data Lake Storage
Hierarchical namespace
Enabled
Default access tier
Hot

Security
Require secure transfer for REST API operations
Enabled
Storage account key access

   If “Blob service” appears instead, you forgot to enable hierarchical namespaces and have created an ordinary blob storage account. In this case, the tab will also indicate that the “Hierarchical namespace” property is “Disabled” – click on the “Disabled” link to upgrade the blob storage account to data lake storage.

5. Click on the “Containers” link in the Storage account blade sidebar (also indicated on the left of the screenshot above). Use the “+ Container” button to create a container with the name “lakeroot”. After creation, the container appears in the list.

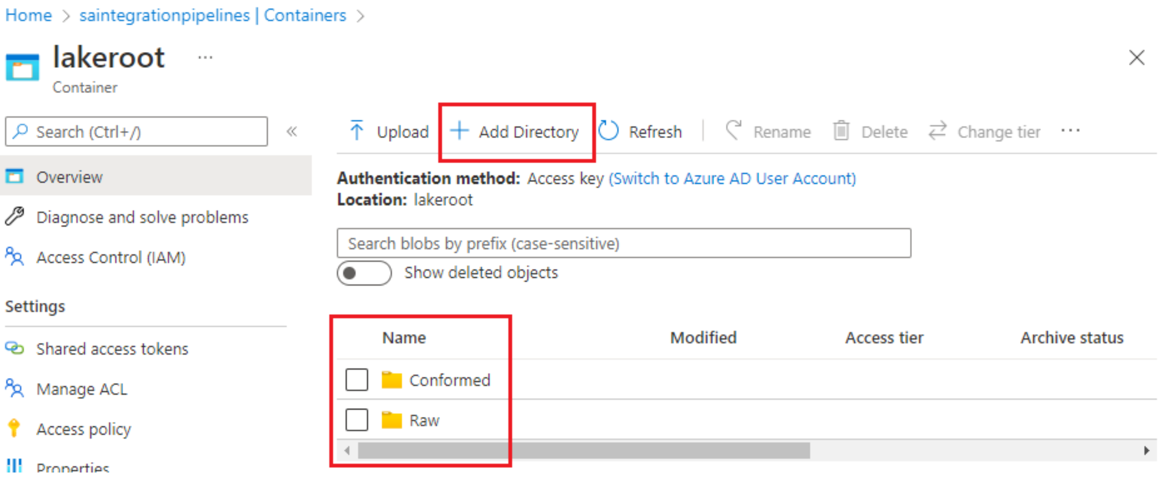


saintegrationpipelines | Containers

Storage account

Name	Last modified	Public access level	Lease state
□ $logs	8/28/2022, 10:55:48 ...	Private	Available
□ lakeroot	8/28/2022, 11:17:06 ...	Private	Available

6. Click on the new “lakeroot” entry to open the container. The menu bar above the list now contains a “+ Add Directory” button – use this to create two directories in the container: “Raw” and “Conformed”.



lakeroot

Container

Authentication method: Access key (Switch to Azure AD User Account)
Location: lakeroot

Name	Modified	Access tier	Archive status
□ Conformed			
□ Raw			

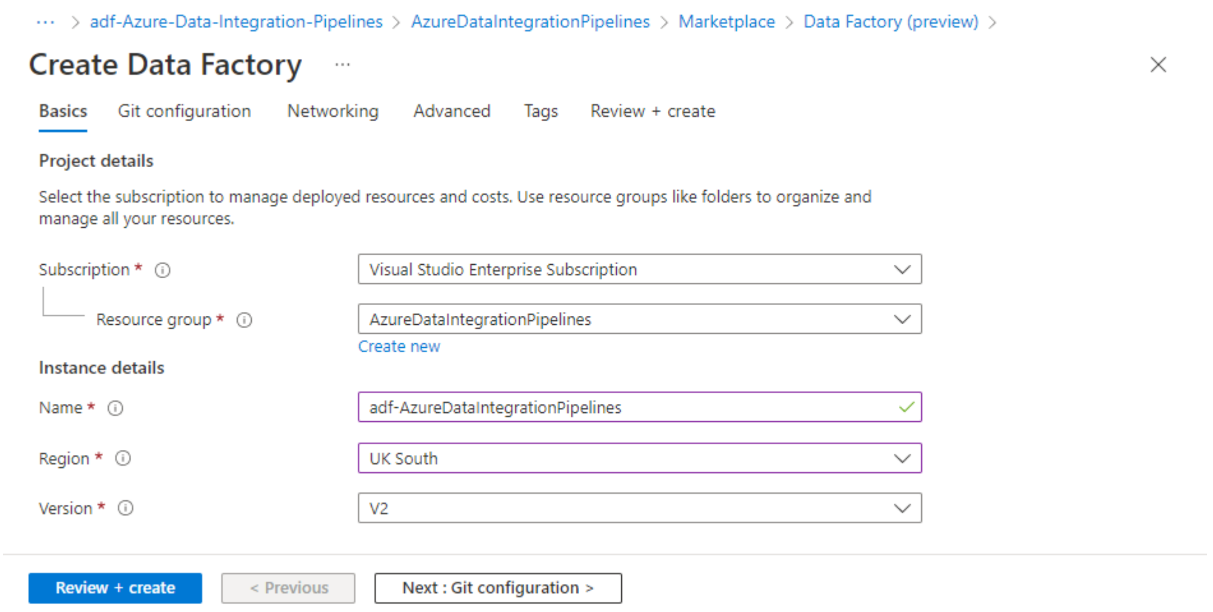
Lab 1.3 – Create an Azure Data Factory

The main event! It’s time to create your Azure Data Factory instance.

1. In the Azure portal, click “Create a resource” and search for “Data Factory”. Click “Create” on the overview screen.

2. Complete the “Basics” tab like this:

    - Choose your subscription and the resource group you created in Lab 1.1.
    - Enter a data factory name – this must be globally unique.
    - Choose the same location (region) you specified for your storage account. **This has cost implications** – transferring data from a storage account in one region to a data factory in another incurs an outbound data transfer charge.
    - Choose version V2.



… > adf-Azure-Data-Integration-Pipelines > AzureDataIntegrationPipelines > Marketplace > Data Factory (preview) >

Create Data Factory …

Basics | Git configuration | Networking | Advanced | Tags | Review + create

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 * : AzureDataIntegrationPipelines

Create new

Instance details

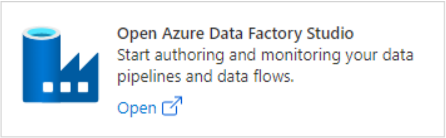
Name * : adf-AzureDataIntegrationPipelines

Region * : UK South

Version * : V2

3. On the “Git configuration” tab, tick the “Configure Git later” checkbox – we’ll do this after the factory has been created because it makes setup a bit easier.

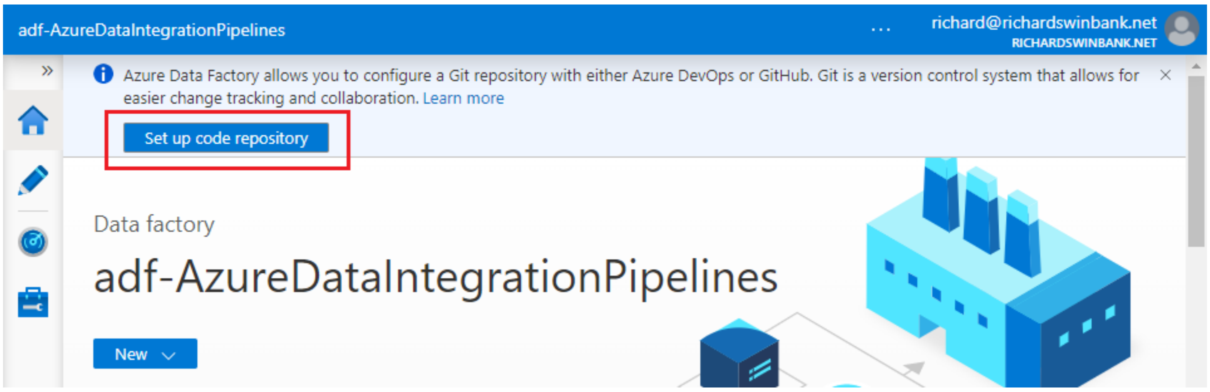
4. Click “Review + create” to accept defaults on the remaining tabs and skip to the “Review + create” tab. Click “Create”. When factory deployment is complete, click on “Go to resource”, then on the “Open Azure Data Factory Studio” tile to launch ADF Studio.



Open Azure Data Factory Studio

Start authoring and monitoring your data pipelines and data flows.

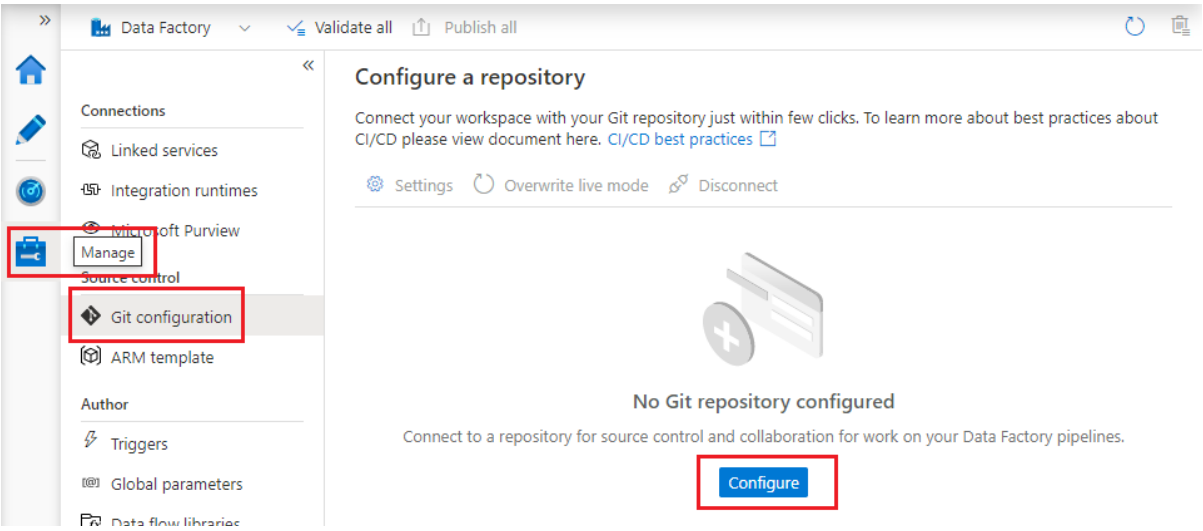
5. Now we’ll connect the new data factory to its GitHub repository. When ADF Studio opens, click the “Set up code repository” button.



adf-AzureDataIntegrationPipelines … richard@richardswinbank.net RICHARDSWINBANK.NET

Azure Data Factory allows you to configure a Git repository with either Azure DevOps or GitHub. Git is a version control system that allows for easier change tracking and collaboration. Learn more

If you don’t see the button, open the Manage hub using the “Manage” button (toolbox icon) on the leftmost sidebar, then click the “Configure” button on the “Git configuration” page.



Configure a repository

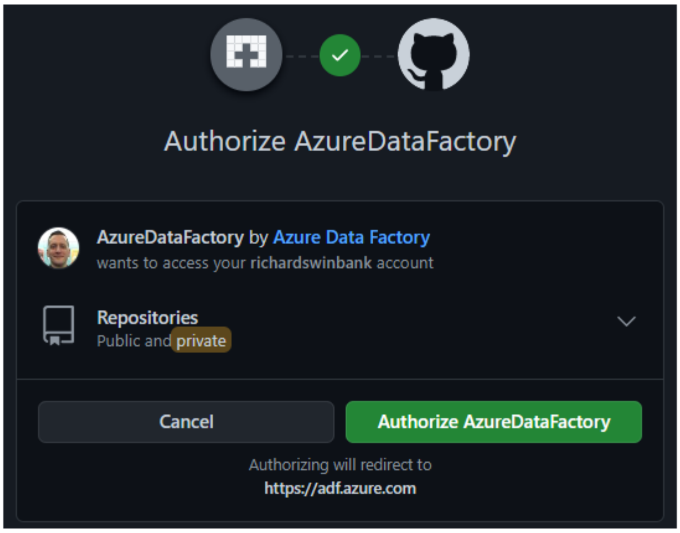
Connect your workspace with your Git repository just within few clicks. To learn more about best practices about CI/CD please view document here. CI/CD best practices

No Git repository configured

Connect to a repository for source control and collaboration for work on your Data Factory pipelines.

6. Either route in step 5 takes you to the "Configure a repository" flyout. Choose "GitHub" from the "Repository type" dropdown list, then enter your GitHub account name into the "GitHub repository owner" field and click "Continue".

7. GitHub prompts you to sign in and to authorise the "AzureDataFactory" application. Follow the instructions to sign into your account and confirm authorisation.



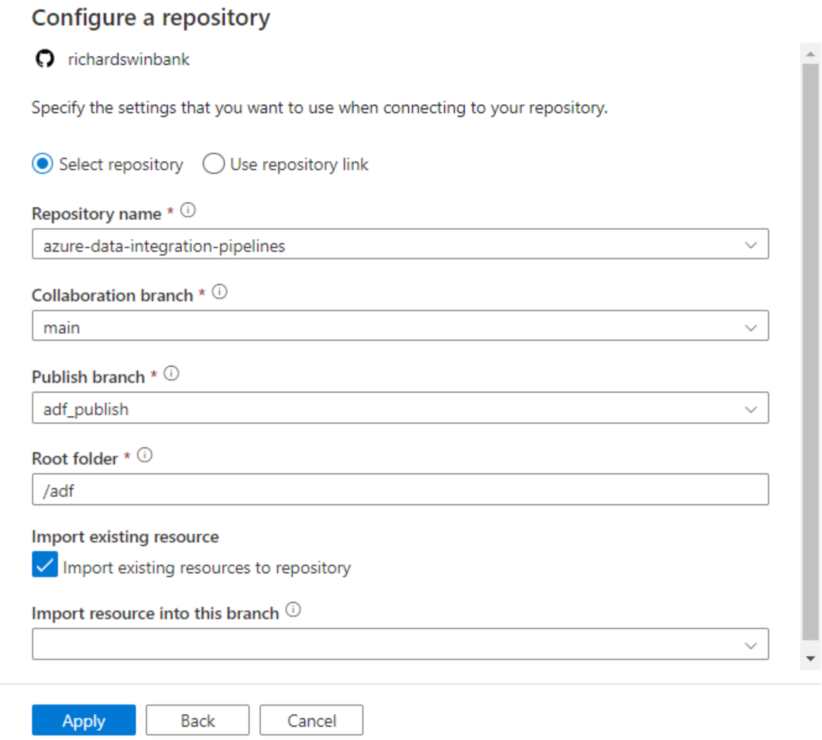
Authorize AzureDataFactory

AzureDataFactory by Azure Data Factory
wants to access your richardswinbank account

Authorizing will redirect to
https://adf.azure.com

8. Once authorised:

   - Select your repository name from the **Repository name** dropdown.
   - Choose a **Collaboration branch**. This is the central codebase – the branch into which all data engineers' feature branches would be merged – often "main" (or "master" in older repositories).
   - Specify a **Root folder**. This is the repository folder where ADF Studio will save data factory artifacts – you can use the root folder ("/") if you wish, but specifying a subfolder allows you to store other files in the repository, separate from ADF artifacts.
   - Click "Apply".



Configure a repository

richardswinbank

Specify the settings that you want to use when connecting to your repository.

● Select repository ○ Use repository link

Repository name * ⓘ: azure-data-integration-pipelines

Collaboration branch * ⓘ: main

Publish branch * ⓘ: adf_publish

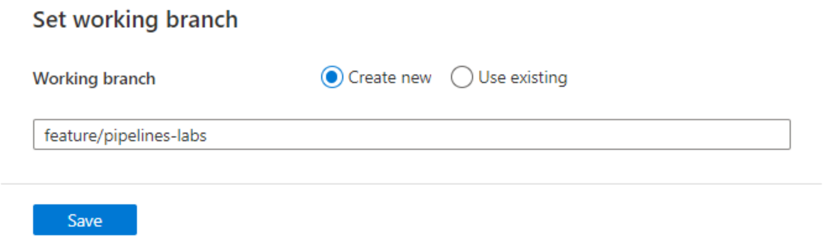
Root folder * ⓘ: /adf

Import existing resource

☑ Import existing resources to repository

Import resource into this branch ⓘ:

9. Finally, ADF Studio asks you to choose your working branch – this setting is for your own ADF Studio session, not for the data factory as a whole or for other engineers. Click “Create new”, enter a feature branch name and click “Save”.



Set working branch

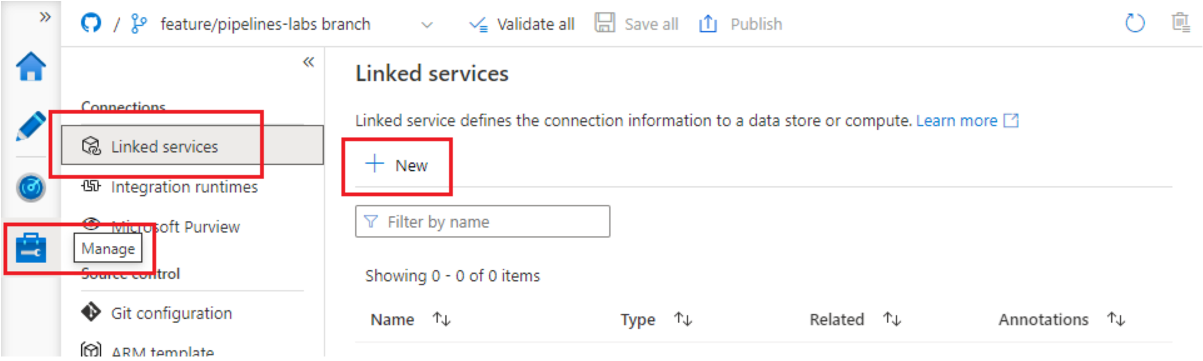
Working branch ● Create new ○ Use existing

feature/pipelines-labs

Lab 1.4 – Connect to the data lake

To enable ADF pipelines to use data in the lake you will need a data factory linked service connection.

1. Navigate to the Manage hub (as described in Lab 1.3), then open the “Linked services” page and click “+ New”.



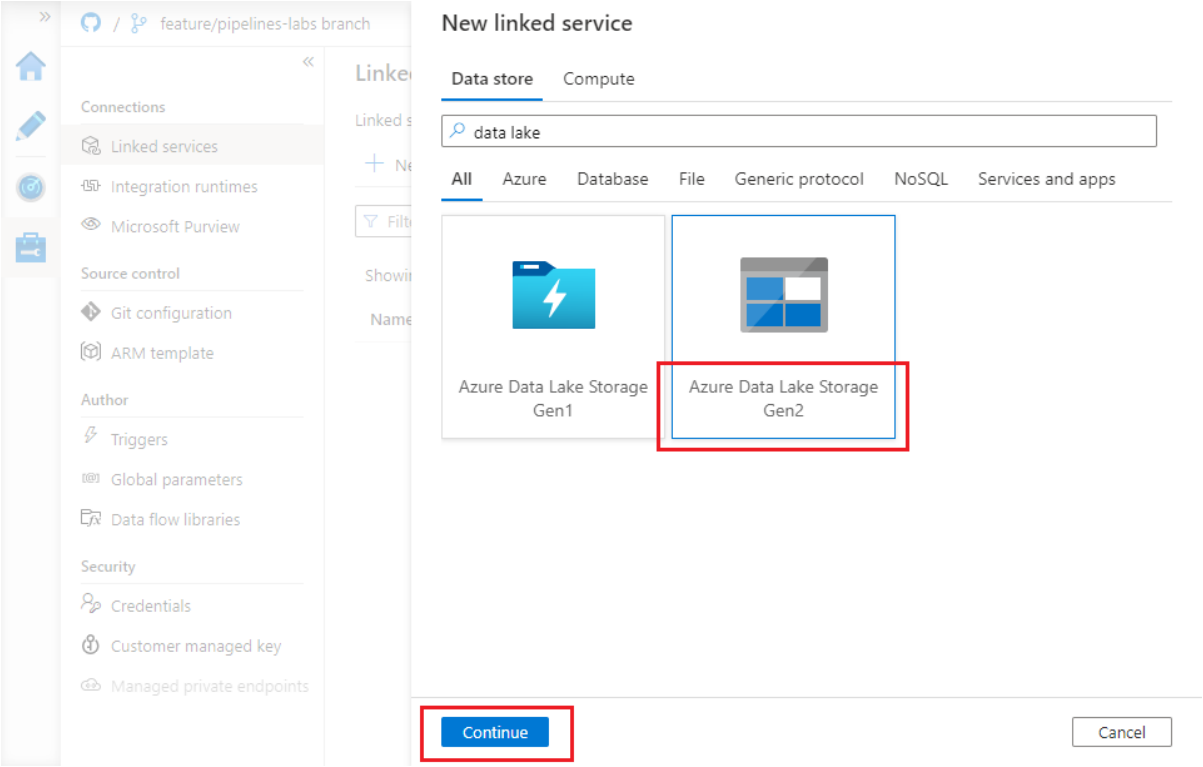
Linked services

Linked service defines the connection information to a data store or compute. Learn more

Showing 0 - 0 of 0 items

Name	Type	Related	Annotations

2. Search for “data lake”, then choose “Azure Data Lake Storage Gen2” and click “Continue”.

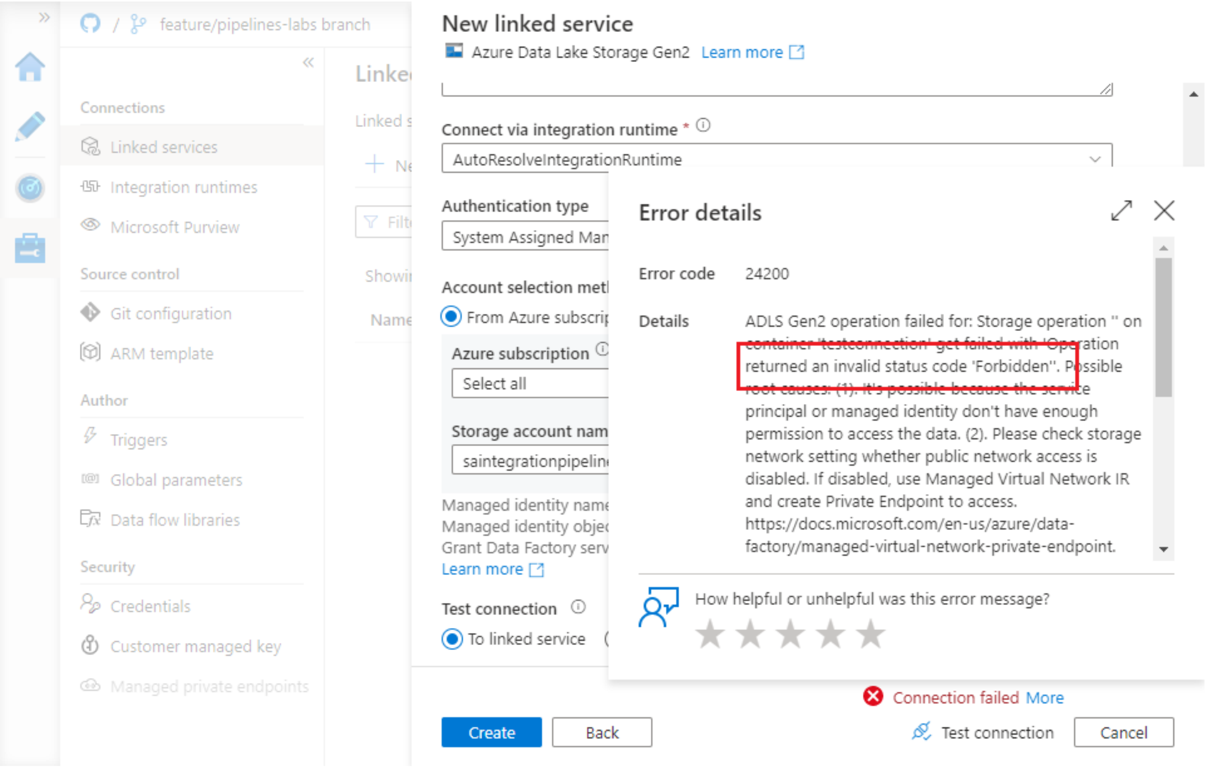


3. Configure linked service details on the “New linked service (Azure Data Lake Storage Gen2)” blade like this:

    - Give it a name.
    - Choose Authentication method “System Assigned Managed Identity”. The default method (“Account key”) requires extra work to pass keys around securely – a “System-Assigned Managed Identity” is an Azure Active Directory service principal, created automatically for your data factory when you created it.
    - Choose your data lake storage account from the “Storage account name” dropdown.

4. Click “Test connection” at the bottom of the blade. The connection test will fail, because the factory’s MSI does not have access to the data lake yet – you will receive an error message like “Operation returned an invalid status code ‘Forbidden’”.

    You will grant the necessary access in a moment – for now, just click “Create” to save the linked service. You do not need explicitly to save linked service changes – they are automatically committed to your GitHub repository.



New linked service

Azure Data Lake Storage Gen2

Connect via integration runtime *: AutoResolveIntegrationRuntime

Authentication type: System Assigned Man

Account selection met: ● From Azure subscrip

Azure subscription: Select all

Storage account nam: saintegrationpipelin

Managed identity name
Managed identity objec
Grant Data Factory serv

Test connection: ● To linked service

Error details

Error code: 24200

Details: ADLS Gen2 operation failed for: Storage operation '' on container 'testconnection' get failed with 'Operation returned an invalid status code 'Forbidden''. Possible root causes: (1). It's possible because the service principal or managed identity don't have enough permission to access the data. (2). Please check storage network setting whether public network access is disabled. If disabled, use Managed Virtual Network IR and create Private Endpoint to access. https://docs.microsoft.com/en-us/azure/data-factory/managed-virtual-network-private-endpoint.

How helpful or unhelpful was this error message? ★★★★★

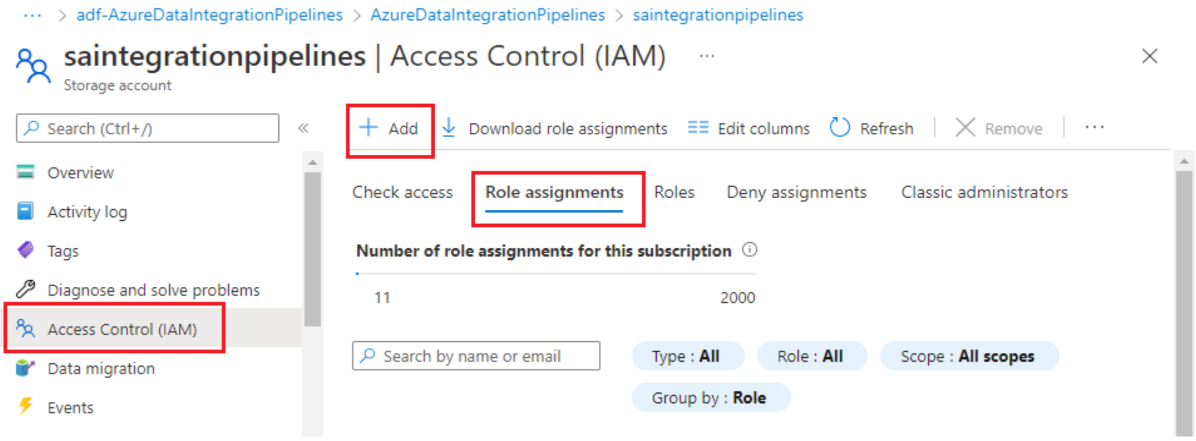
Connection failed More

Lab 1.5 – Grant access to the data lake

You can manage access to Azure resources in the Azure portal. Open a new browser tab to allow you to keep ADF Studio open.

1. Browse to your data lake resource blade – you can find it in the list of resources on the portal home page, or by selecting your resource group to see resources inside it, or by using the search box in the portal's top menu bar.

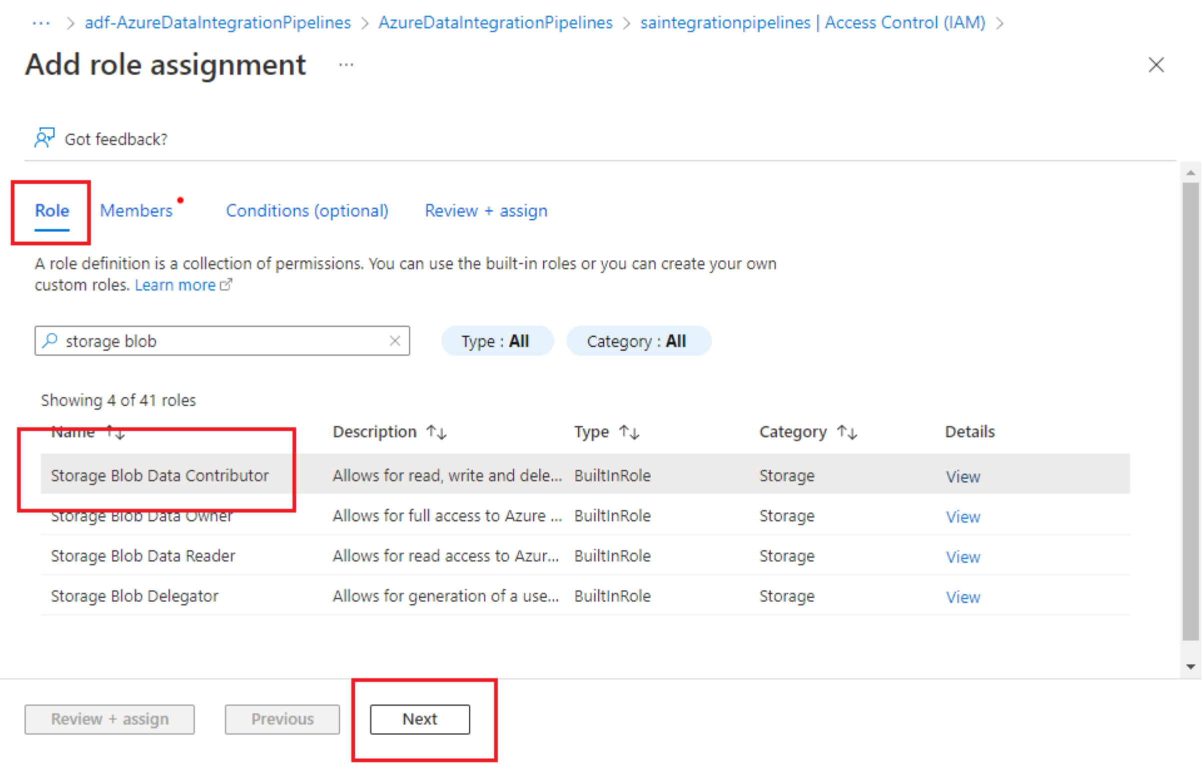
2. Click "Access control (IAM)" in the storage account (data lake) resource blade, then select the "Role assignments" tab. Click "+ Add" and select "Add role assignment" from the dropdown list.



Number of role assignments for this subscription

11 2000

3. On the "Role" tab of the "Add role assignment" blade, choose the "Storage Blob Data Contributor" role – this authorises read, write and delete access in your data lake – then click "Next".



... > adf-AzureDataIntegrationPipelines > AzureDataIntegrationPipelines > saintegrationpipelines | Access Control (IAM) >

Add role assignment

Got feedback?

Role | Members | Conditions (optional) | Review + assign

A role definition is a collection of permissions. You can use the built-in roles or you can create your own custom roles. Learn more

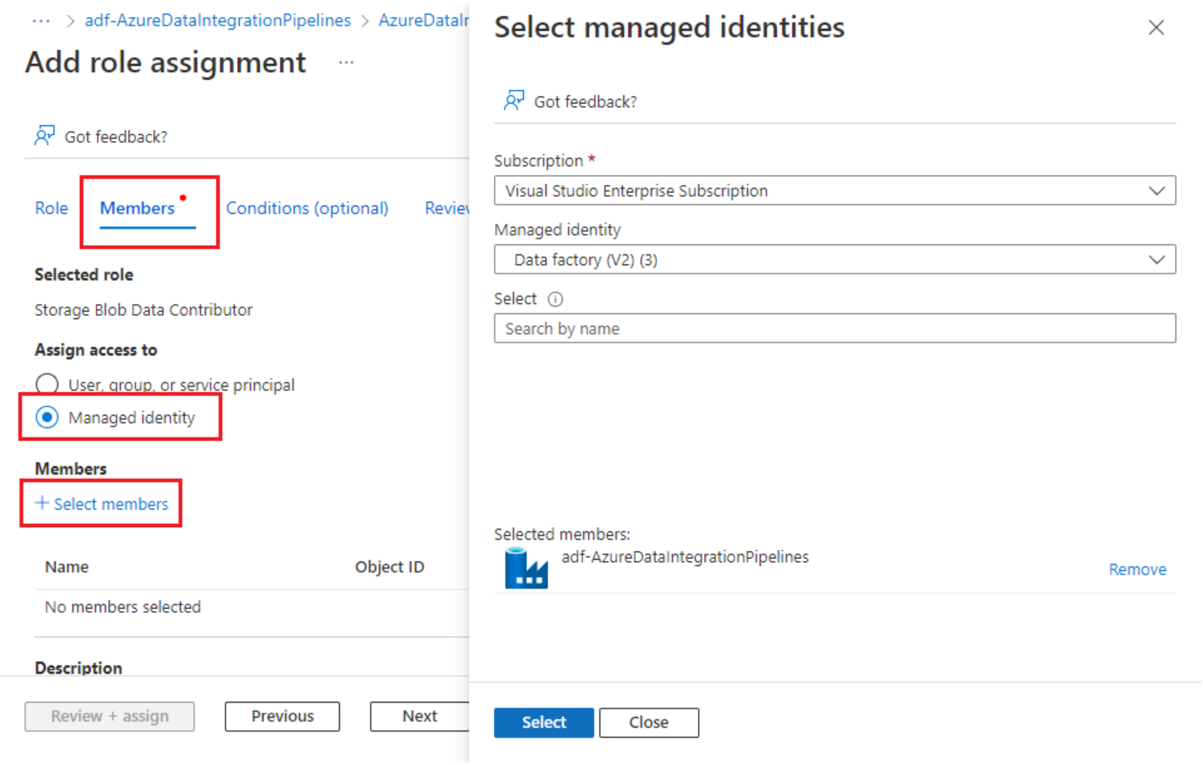
storage blob

Type : **All** | Category : **All**

Showing 4 of 41 roles

Name ↑↓	Description ↑↓	Type ↑↓	Category ↑↓	Details
Storage Blob Data Contributor	Allows for read, write and dele...	BuiltInRole	Storage	View
Storage Blob Data Owner	Allows for full access to Azure ...	BuiltInRole	Storage	View
Storage Blob Data Reader	Allows for read access to Azur...	BuiltInRole	Storage	View
Storage Blob Delegator	Allows for generation of a use...	BuiltInRole	Storage	View

4. On the “Members” tab, set “Assigned access to” to “Managed identity”, then click “+ Select members”.



... > adf-AzureDataIntegrationPipelines > AzureDataIr

Add role assignment

Got feedback?

Role | Members | Conditions (optional) | Revie

Selected role

Storage Blob Data Contributor

Assign access to

○ User, group, or service principal

● Managed identity

Members

+ Select members

Name	Object ID
No members selected	

Description

Select managed identities

Got feedback?

Subscription *

Visual Studio Enterprise Subscription

Managed identity

Data factory (V2) (3)

Select

Search by name

Selected members:

adf-AzureDataIntegrationPipelines

Choose your Azure Data Factory instance from the “Select managed identities” flyout, click “Select” to dismiss the flyout, then click “Review + assign”. On the “Review + assign” tab, click the button of the same name.

5. Return to the Manage hub in ADF Studio, and click on your data lake linked service to re-open the editing blade. Click “Test connection” again, and verify that this time the connection test succeeds. If the test fails, wait a few minutes, then try again – it may take a short time for permission changes to take effect. Click “Cancel” to close the editing blade.

Recap

In Lab 1 you:

- created an Azure resource group
- created data lake storage
- created an instance Azure Data Factory
- created and authorised a linked service connection from the factory to your data lake storage.