

Lab 01 – Setting up the Lake

Welcome to the first lab!

We're going to make some assumptions before we kick things off. We are assuming that you:

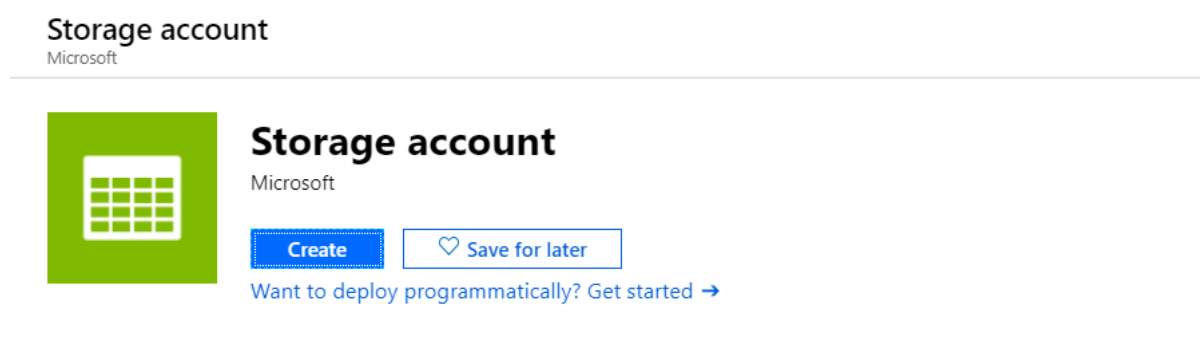
- Have access to an Azure subscription (even a [free trial](#))
- Have sufficient access on this subscription to create resources
- Have access to a Service Principal (or the access to create one)

The above aren't show-stoppers and can be worked around – but we haven't taken them into account in the instructions below!

Lab 01.A – Creating A Data Lake

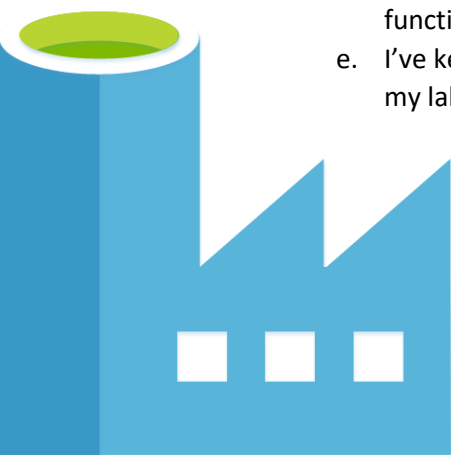
First we need to create a new storage account to house our lake. Unlike the first generation of Microsoft Data Lakes, this is not a separate resource and is created through the normal blob process.

1. In the Portal, click "Create a Resource" and search for "Storage". You should find the following.



Microsoft Azure provides scalable, durable cloud storage, backup, and recovery solutions for any data, big or small. It works with the infrastructure you already have to cost-effectively enhance your existing applications and business continuity strategy, and provide the storage required by your cloud applications, including unstructured text or binary data such as video, audio, and images.

2. Click "Create" and you'll see a short form to fill in. Bear the following in mind:
 - a. I've created a new resource group for the labs – but you can use any!
 - b. The blob name has to be unique – not just for you, but across all blobs
 - c. Pick your region carefully – you want all the items you create to live in the same region otherwise you'll incur egress costs
 - d. Make sure you choose "StorageV2" – this is the only version that Data Lake functionality is enabled for
 - e. I've kept things cheap by turning off geo-replication – I don't need backup copies of my lake, I would definitely turn this on for a production environment!



Lab 01 – Setting up the Lake

PROJECT DETAILS

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

* Subscription

* Resource group [Create new](#)

INSTANCE DETAILS

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

* Storage account name

* Location

Performance ☒ Standard ☐ Premium

Account kind

Replication

Access tier (default) ☐ Cool ☒ Hot

- You'll then want to click the "Advanced" button, where we will see the option to enable "Hierarchal Namespace" – this is essentially making it a Data Lake Store Gen 2 resource

[Basics](#) [Advanced](#) [Tags](#) [Review + create](#)

SECURITY

Secure transfer required ☐ Disabled ☒ Enabled

VIRTUAL NETWORKS

Allow access from ☒ All networks ☐ Selected network
 All networks will be able to access this storage account. [Learn more](#)

DATA PROTECTION

Blob soft delete ☐ Disabled ☐ Enabled
 Blob soft delete and hierarchical namespace cannot be enabled simultaneously.

DATA LAKE STORAGE GEN2

Hierarchical namespace ☐ Disabled ☒ Enabled

- You can now click "Review and Create" and finally "Create" once it has validated your request. This will now create a blob storage account for you with that all important hierarchal namespace enabled. It might take a few minutes.
- Once complete, you can navigate to the resource and you should see the all important ADLS Gen 2 box:

Lab 01 – Setting up the Lake

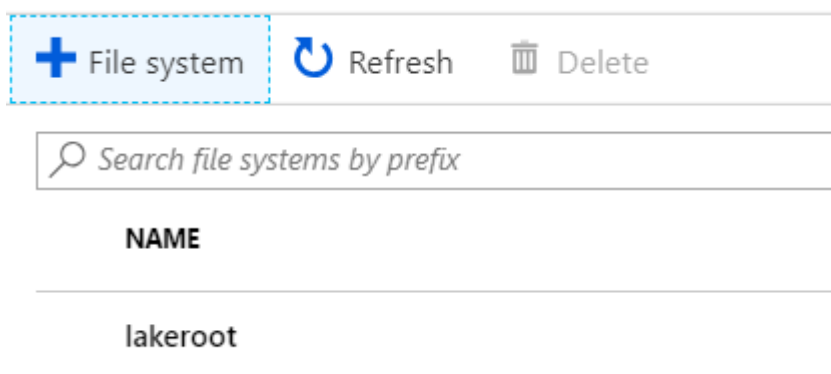


Data Lake Gen2 file systems

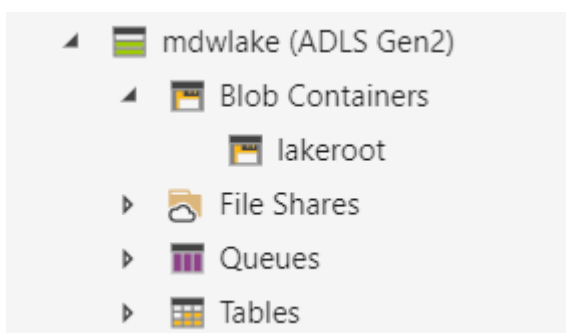
Massively scalable data lake storage

[Learn more](#)

- Click into this box and you'll be prompted to click "+ File System" to create a new lake structure. Do this now.
- I've called mine "lakeroot" so I can now see the following:



- If you click into your new file structure, you'll see a large Data Lake icon and a message telling you to use "Azure Storage Explorer" to interact with the lake from now on. Make sure you have this installed locally.
- If you have not used Storage Explorer before, you'll need to enter your Azure Subscription credentials. Once you do, you should see a list of your storage accounts and the new hierarchal namespace will appear under blob containers. Mine looks as follows:



Lab 01 – Setting up the Lake

Lab 01.B – Creating A Data Factory

Now that we have a lake ready to use, we can setup Data Factory to start copying data into it.

1. Use the “Create a Resource” button again and this time search for Data Factory to find the following:

Data Factory Microsoft



Data Factory

Microsoft

Create

Save for later

Microsoft Azure Data Factory is a cloud-based data integration service that automates the movement and transformation of data. You can quickly create, deploy, schedule, and monitor highly-available, fault tolerant data flow pipelines. Move and transform data of all shapes and sizes, and deliver the results to a range of destination storage services. Monitor all of your data pipelines and service health at a glance with a rich visual experience. Easily consume the data produced with BI, analytics tools, and other applications to drive key business insights and decisions.

2. Click create and follow the instructions

* Name ⓘ

mdwfactory



* Subscription

Microsoft Azure Sponsorship



* Resource Group ⓘ



Create new



Use existing

MDWBoot



Version ⓘ

V2



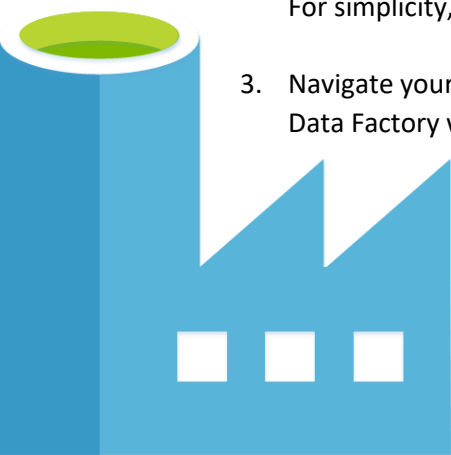
* Location ⓘ

UK South

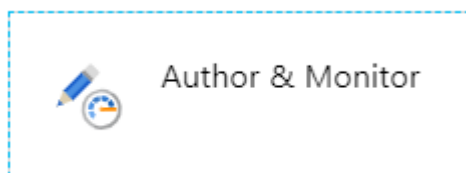



For simplicity, I've not enabled git – but you can do if you have the details ready to go.

3. Navigate your new data factory, and click on the “Author and Manage” button to open the Data Factory workspace




Lab 01 – Setting up the Lake

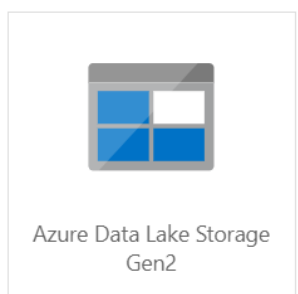


- You'll see the entry screen load, but we want to go straight to our pipeline creation, so click on the  button, or the "Create Pipeline" button:



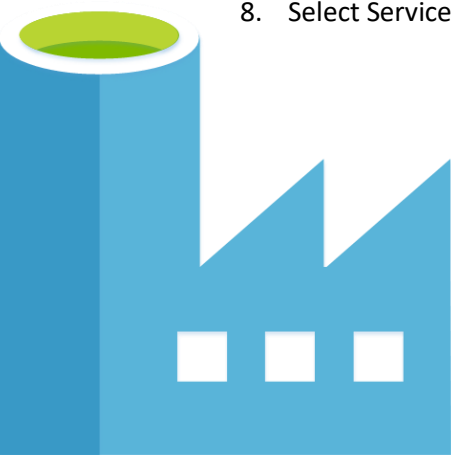
- Our first step is to ensure the Data Factory has access to the lake, so we can go straight to the  Connections button.

- Click on the +New button and select the Data Lake Store Gen 2 icon:



This will open up a linked services blade, specifically for ADLS Gen 2.

- We could use a blob key, but it's better if we allow use a service principal account to do this. We could allow the ADF Managed Service Identity to have access – this is a special service account managed by data factory itself, but I prefer to have different service principals to enable more fine grain on access levels.
- Select Service Principal and you'll be prompted for further details:



Lab 01 – Setting up the Lake

← New Linked Service (Azure Data Lake Storage G... ×

Connect via integration runtime *
AutoResolveIntegrationRuntime

Authentication method
Service Principal

Account selection method
☒ From Azure subscription
 ☐ Enter manually

Azure subscription
Microsoft Azure Sponsorship (452253a1-4716-4f1d-b374-fee0c7035fcf)

Storage account name *
mdwlake

Tenant *
eea0d0c1-f710-4f1e-9b5c-b819f0f5a786

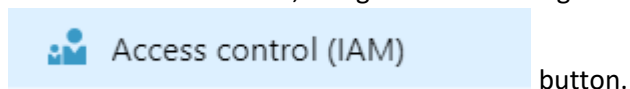
Service principal ID *

Service principal key *
 Service principal key Azure Key Vault

Now, before we go any further, we haven't actually given our Service Principal access to our lake yet! Before you create your linked service in data factory, let's make sure it has access. Don't close your data factory window as we'll be coming back here later!

If you haven't got a Service Principal yet – follow the optional lab 1.1!

- Back in the Azure Portal, navigate to the Storage Account you set up. Click on the



button.

- We want to create a new role assignment for our lake user, so click on "Add" or the "Add a Role Assignment" button.

We want to create a new "Storage Blob Data Owner", which gives our Service Principal access to read and write data in the lake.

Add role assignment ×

Role ⓘ
Storage Blob Data Owner

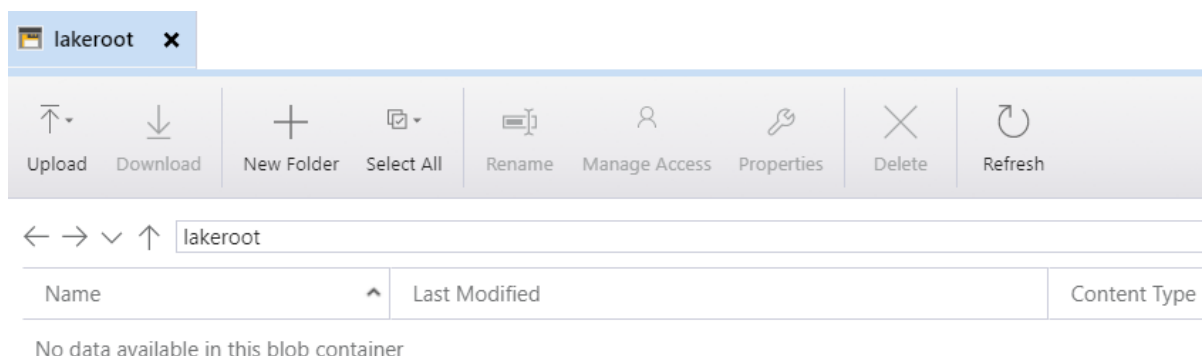
Assign access to ⓘ
Azure AD user, group, or service principal

Select ⓘ
Search by name or email address ✓

Note: You should be able to use the more restrictive roles (storage blob data reader/writer) but ADF sometimes errors with these roles – we assume this will be fixed over time!

Lab 01 – Setting up the Lake

11. Switch back to Azure Storage Explorer and navigate to your lake namespace

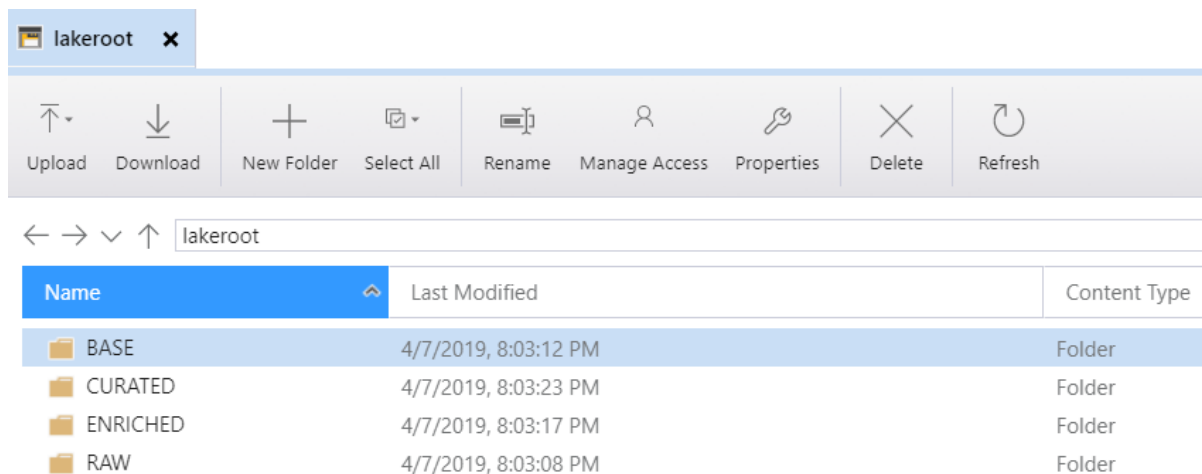


You'll see many of our options are greyed out, this is because we need to start off with some objects.

12. We can then use the +New Folder button to create the base set of data lake folders:

RAW
BASE
ENRICHED
CURATED

13. Your lake should now look as follows:



14. We can now return to Data Factory to finish setting up our Linked Service.
15. Enter your Service Principal ID and Key then click the "Test Connection" button to ensure access has been set up correctly:



Lab 01 – Setting up the Lake

Tenant *

Service principal ID *

Service principal key *

Annotations

+ New

Cancel

Test connection

Finish

Connection successful

16. If you see the “Connection Successful” button, you can click “Finish” and you now have a Data Factory set up to work with your Data Lake.

Connections

Triggers

Linked Services

Integration Runtimes

+ New

Name

Actions

ADLS_MDWLake