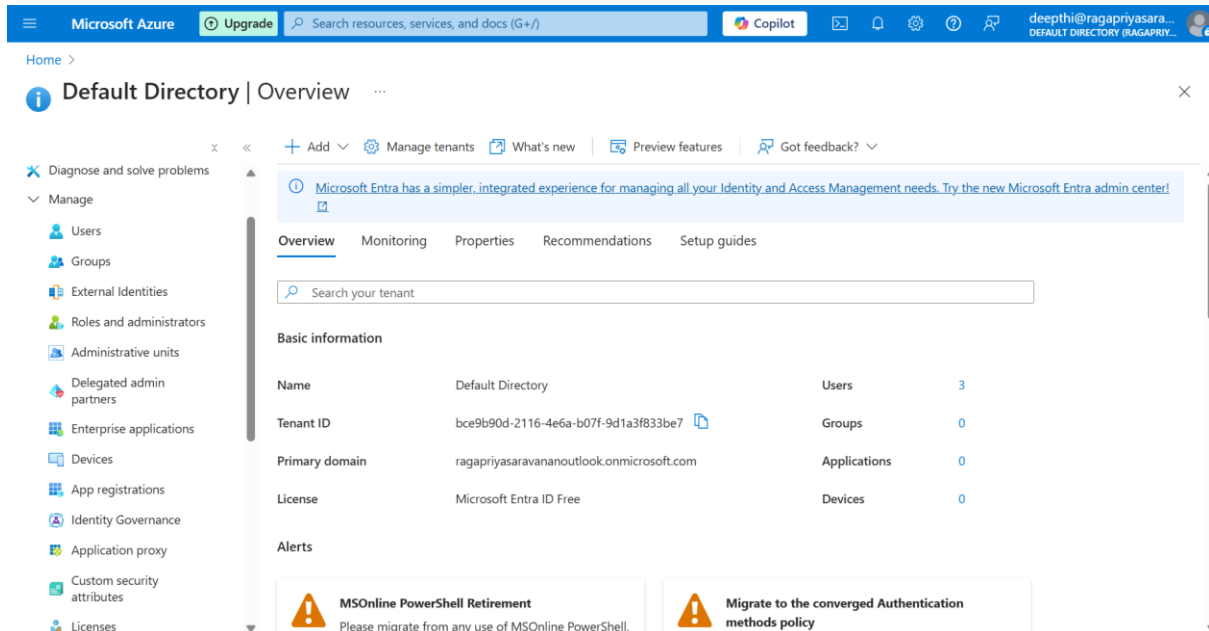


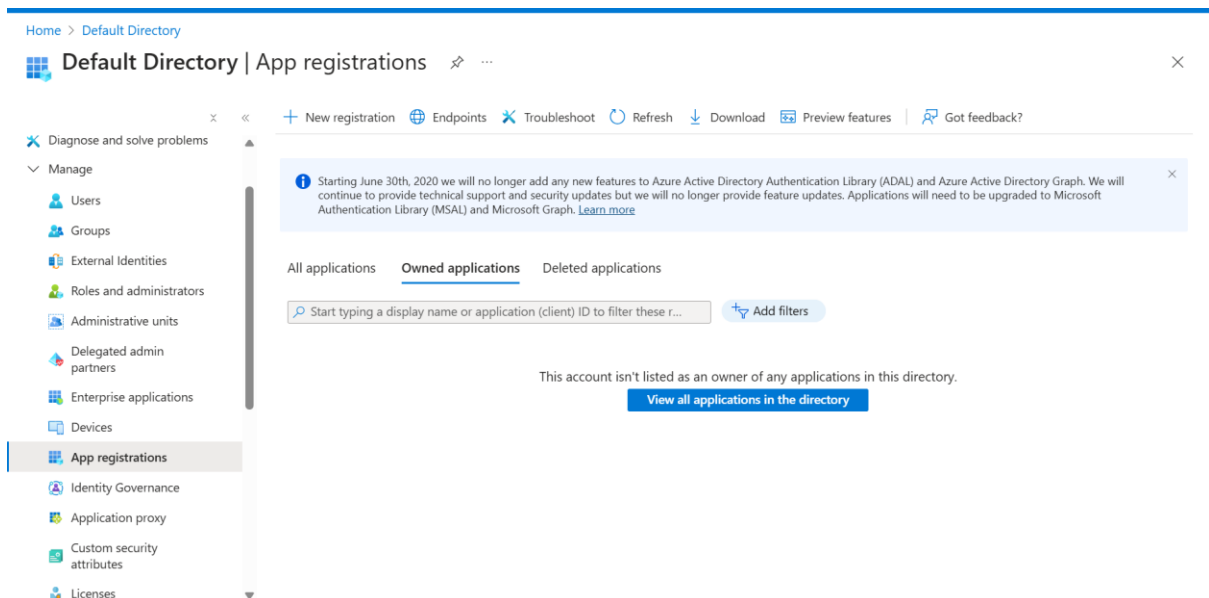
Mount to ADLS gen2 using Service Principle

Create a service principle in Azure account

Go to Azure account home page and search for Microsoft Entra ID and go to manage section



Here go to App registration



Create new registration

Give name and opt for second option in supported account types and click register

Register an application

* Name

The user-facing display name for this application (this can be changed later).

Supported account types

Who can use this application or access this API?

- ☐ Accounts in this organizational directory only (Default Directory only - Single tenant)
- ☒ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant)
- ☐ Accounts in any organizational directory (Any Microsoft Entra ID tenant - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
- ☐ Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)

By proceeding, you agree to the [Microsoft Platform Policies](#)

[Register](#)

Copy Application ID and Tentant ID and store somewhere and click on Client credentials

databricksadlsmount

[Delete](#) [Endpoints](#) [Preview features](#)

Overview

- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
- Support + Troubleshooting

Essentials

Display name

databricksadlsmount

Application (client) ID

ac3d15b7-7c2b-4bdf-ad29-378334a887db

Object ID

edb87781-b8c3-46e0-a9ed-45ae6

Directory (tenant) ID

bce9b90d-2116-4e6a-b07f-9d1a3f833be7

Supported account types

[Multiple organizations](#)

Client credentials

[Add a certificate or secret](#)

Redirect URIs

[Add a Redirect URI](#)

Application ID URI

[Add an Application ID URI](#)

Managed application in local directory

databricksadlsmount

Welcome to the new and improved App registrations. Looking to learn how it's changed from App registrations (Legacy)? [Learn more](#)

Starting June 30th, 2020 we will no longer add any new features to Azure Active Directory Authentication Library (ADAL) and Azure Active Directory Graph. We will continue to provide technical support and security updates but we will no longer provide feature updates. Applications will need to be upgraded to Microsoft Authentication Library (MSAL) and Microsoft Graph. [Learn more](#)

Starting November 9th, 2020 end users will no longer be able to grant consent to newly registered multitenant apps without verified publishers. [Add MPN ID to verify publisher](#)

databricksadlsmount | Certificates & secrets

[Got feedback?](#)

Overview

- Quickstart
- Integration assistant
- Diagnose and solve problems
- Manage
 - Branding & properties
 - Authentication
 - Certificates & secrets**
 - Token configuration
 - API permissions
 - Expose an API
 - App roles
 - Owners
 - Roles and administrators
 - Manifest
- Support + Troubleshooting

Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.

Application registration certificates, secrets and federated credentials can be found in the tabs below.

Certificates (0) **Client secrets (0)** Federated credentials (0)

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

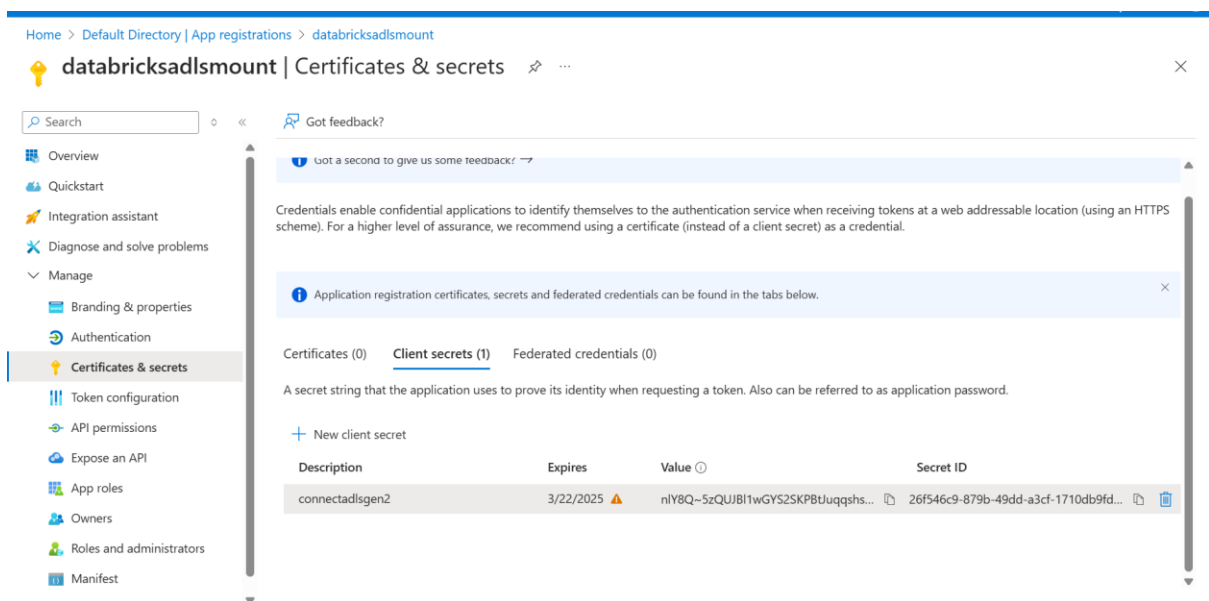
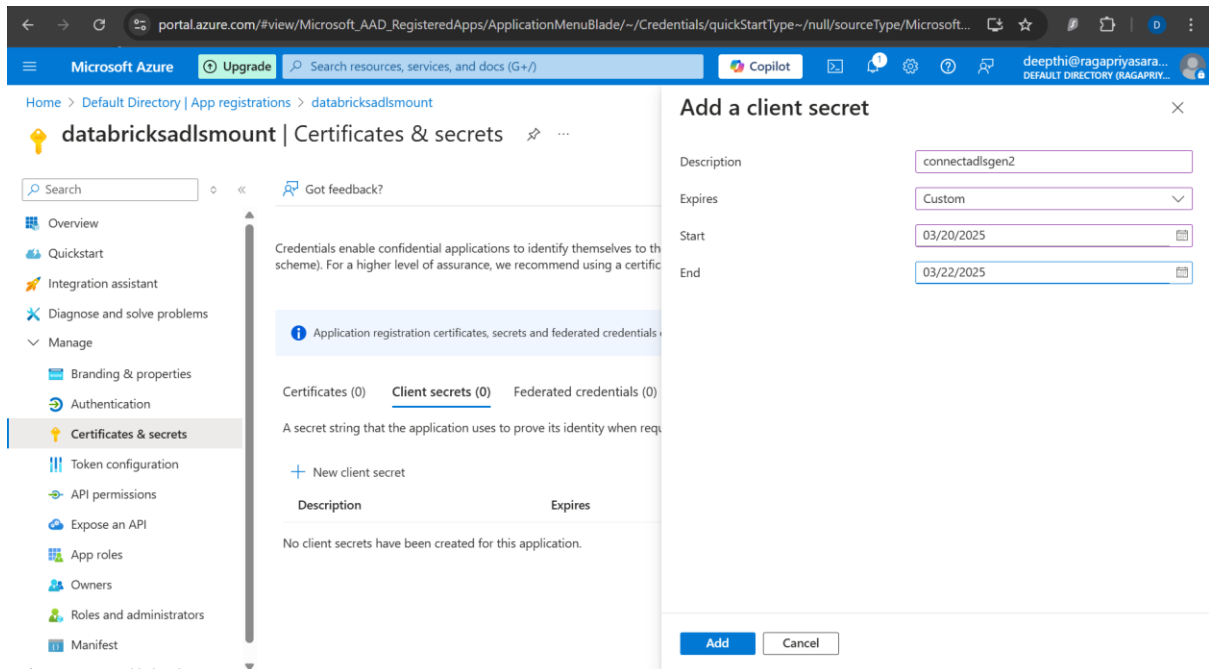
[+ New client secret](#)

Description	Expires	Value	Secret ID
-------------	---------	-------	-----------

No client secrets have been created for this application.

Create new client secret

Give description and select expires , opt for custom and give start and end dates



Now go to key vault and create secrets using these details

App ID key

App ID value you will get from

Microsoft Entra ID-> Manage -> App register -> There Application ID

Value ID: Microsoft Entra ID-> Manage -> App register -> All Applications-> Select the one which we created-> manage-> certificates and secrets -> value
need to copy the value key once application is created after it will not display



Create a secret ...



Upload options	Manual
Name *	AppID ✓
Secret value *	***** ✓
Content type (optional)	
Set activation date	<input type="checkbox"/>
Set expiration date	<input checked="" type="checkbox"/>
Enabled	Yes No
Tags	0 tags

Create Cancel



Create a secret ...



Upload options	Manual
Name *	AppValue ✓
Secret value *	***** ✓
Content type (optional)	
Set activation date	<input type="checkbox"/>
Set expiration date	<input type="checkbox"/>
Enabled	Yes No
Tags	0 tags

Create Cancel

2 keys are generated

+ Generate/Import Refresh Restore Backup Manage deleted secrets View sample code

i The secret 'AppValue' has been successfully created.

Name	Type	Status	Expiration date
AppValue		✓ Enabled	
AppID		✓ Enabled	

Now have to create scope with these 2 keys

Got to [https://adb-](https://adb-2147694267598473.13.azuredatabricks.net/?o=2147694267598473#secrets/createScope)

[2147694267598473.13.azuredatabricks.net/?o=2147694267598473#secrets/createScope](https://adb-2147694267598473.13.azuredatabricks.net/?o=2147694267598473#secrets/createScope)

Create Secret Scope

Cancel

Create

A store for secrets that is identified by a name and backed by a specific store type. [Learn more](#)

Scope Name ?

askvconnections

Manage Principal ?

Creator

Azure Key Vault ?

DNS Name

https://blobkvdatabricks.vault.azure.net/

Resource ID

/resourceGroups/DKresource/providers/Microsoft.KeyVault/vaults/blobkvdatabricks

Now we have to give storage blob Data Contributor role to service principle

Go to storage account-> Access control-> Add role assignment

Home > Storage accounts > adlsdeepthi | Access Control (IAM) >

Add role assignment

Role Members Conditions 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](#)

Job function roles Privileged administrator roles

Grant access to Azure resources based on job function, such as the ability to create virtual machines.

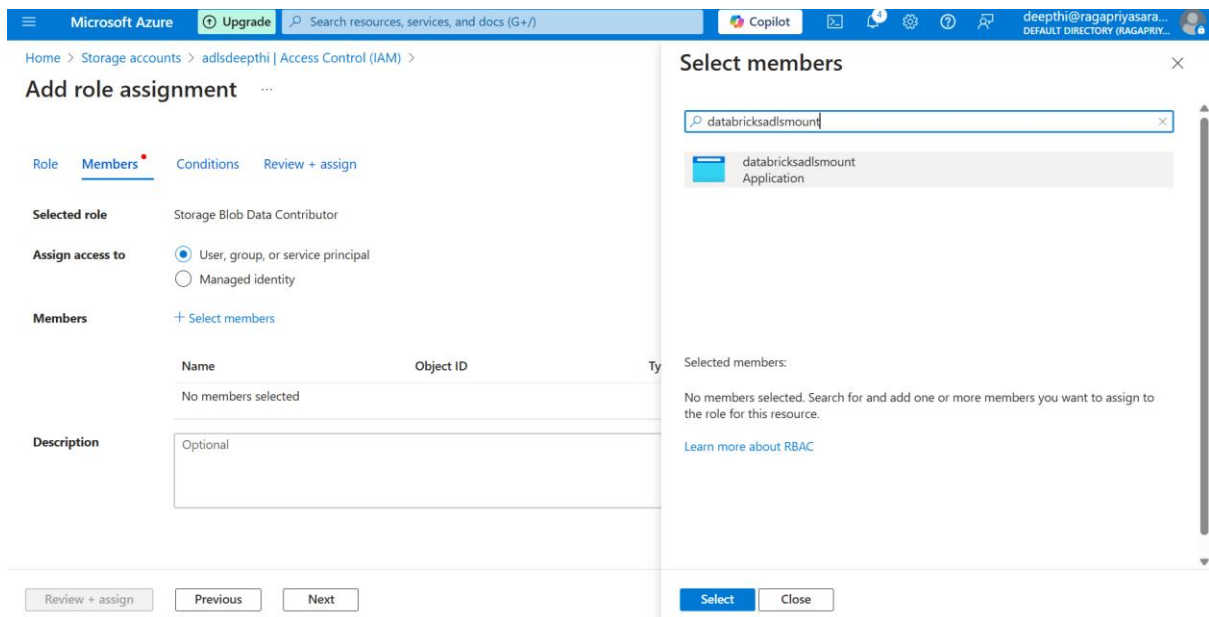
blob Type: All Category: All

Name	Description	Type	Category	Details
Defender CSPM Storage Data Scan...	Grants access to read blobs and files. This role is used by the data scanner of Dfender CSPM.	BuiltInRole	None	View
Defender for Storage Data Scanner	Grants access to read blobs and update index tags. This role is used by the data scanner of ...	BuiltInRole	None	View
Storage Blob Data Contributor	Allows for read, write and delete access to Azure Storage blob containers and data	BuiltInRole	Storage	View
Storage Blob Data Owner	Allows for full access to Azure Storage blob containers and data, including assigning POSIX ...	BuiltInRole	Storage	View
Storage Blob Data Reader	Allows for read access to Azure Storage blob containers and data	BuiltInRole	Storage	View
Storage Blob Delegator	Allows for generation of a user delegation key which can be used to sign SAS tokens	BuiltInRole	Storage	View

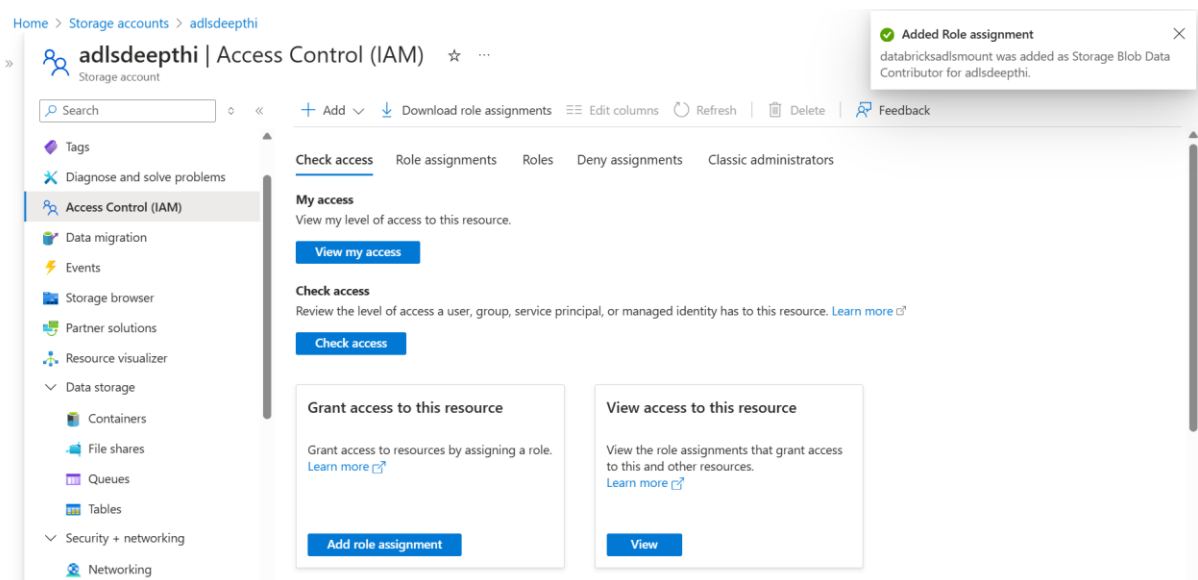
Review + assign Previous Next

Feedback

Click on next and search for application name



Role as been assigned



Now we have to write code to mount Databricks to ADLS gen2 using service principle

```
configs = {"fs.azure.account.auth.type": "OAuth",
          "fs.azure.account.oauth.provider.type":
"org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
          "fs.azure.account.oauth2.client.id":
dbutils.secrets.get(scope="askvconnections",key="AppID"),
          "fs.azure.account.oauth2.client.secret":
dbutils.secrets.get(scope="askvconnections",key="AppValue"),
          "fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/bce9b90d-
2116-4e6a-b07f-9d1a3f833be7/oauth2/token"}
```

```

dbutils.fs.mount(

source = "abfss://ewd@adlsdeepthi.dfs.core.windows.net/",

mount_point = "/mnt/ewd",

extra_configs = configs)

```

```

02:25 PM (1s) 2
dbutils.secrets.listScopes()

[SecretScope(name='askvconnection'), SecretScope(name='askvconnections')]

02:25 PM (1s) 3 Python
dbutils.secrets.list("askvconnections")

[SecretMetadata(key='AppID'),
 SecretMetadata(key='AppValue'),
 SecretMetadata(key='sas-key')]

2 minutes ago (13s) 4 Python
configs = {"fs.azure.account.auth.type": "OAuth",
           "fs.azure.account.oauth.provider.type": "org.apache.hadoop.fs.azurebfs.oauth2.ClientCredsTokenProvider",
           "fs.azure.account.oauth2.client.id": dbutils.secrets.get(scope="askvconnections",key="AppID"),
           "fs.azure.account.oauth2.client.secret": dbutils.secrets.get(scope="askvconnections",key="AppValue"),
           "fs.azure.account.oauth2.client.endpoint": "https://login.microsoftonline.com/bce9b90d-2116-4e6a-b07f-9d1a3f833be7/oauth2/token"}

dbutils.fs.mount(
    source = "abfss://ewd@adlsdeepthi.dfs.core.windows.net/",
    mount_point = "/mnt/ewd",
    extra_configs = configs)

True

```

Mount is successful

All the files of that container

```

Just now (1s) 5 Python
dbutils.fs.ls("/mnt/ewd")

[FileInfo(path='dbfs:/mnt/ewd/Customer/', name='Customer/', size=0, modificationTime=1741712870000),
 FileInfo(path='dbfs:/mnt/ewd/Customers/', name='Customers/', size=0, modificationTime=1740606687000),
 FileInfo(path='dbfs:/mnt/ewd/Data_Migration_SCD_2/', name='Data_Migration_SCD_2/', size=0, modificationTime=1741746870000),
 FileInfo(path='dbfs:/mnt/ewd/Data_Migration_SCD/', name='Data_Migration_SCD/', size=0, modificationTime=1741731988000),
 FileInfo(path='dbfs:/mnt/ewd/Delta_file/', name='Delta_file/', size=0, modificationTime=1741632189000),
 FileInfo(path='dbfs:/mnt/ewd/Employee/', name='Employee/', size=0, modificationTime=1741712848000),
 FileInfo(path='dbfs:/mnt/ewd/Orders/', name='Orders/', size=0, modificationTime=1740610891000),
 FileInfo(path='dbfs:/mnt/ewd/Product/', name='Product/', size=0, modificationTime=1740621793000),
 FileInfo(path='dbfs:/mnt/ewd/Rest/', name='Rest/', size=0, modificationTime=1741716963000),
 FileInfo(path='dbfs:/mnt/ewd/Rest_Ap/', name='Rest_Ap/', size=0, modificationTime=1741728729000),
 FileInfo(path='dbfs:/mnt/ewd/Sales/', name='Sales/', size=0, modificationTime=1740613864000),
 FileInfo(path='dbfs:/mnt/ewd/post', name='post', size=21240, modificationTime=1741729074000),
 FileInfo(path='dbfs:/mnt/ewd/synapse/', name='synapse/', size=0, modificationTime=1741732852000)]

```

Read the data from the container

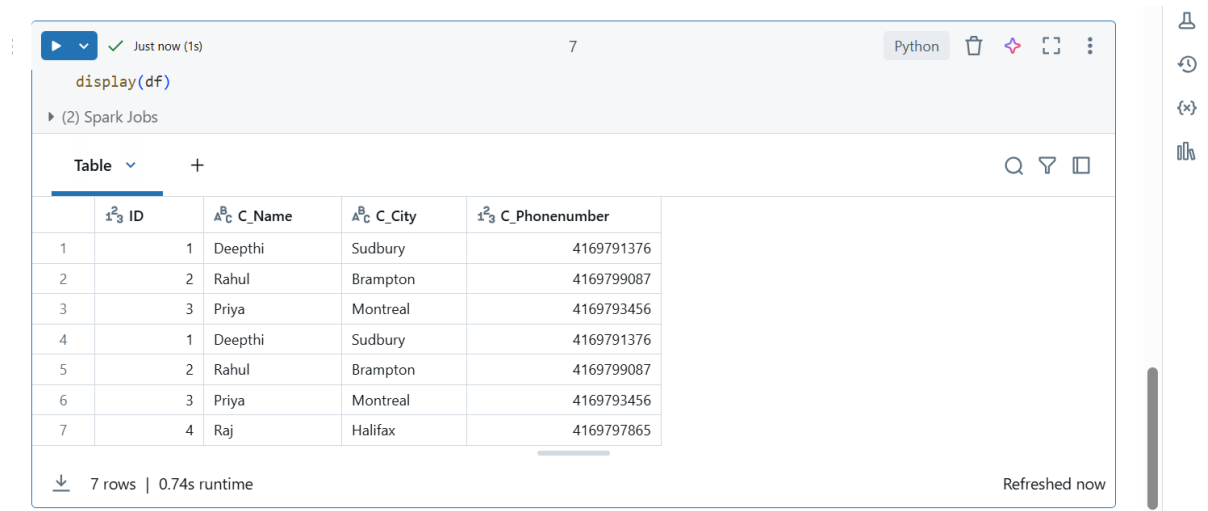
```

Just now (2s) 6 Python
df=spark.read.format("csv").option("header", "true").option("inferSchema", "true").load("/mnt/ewd/Customer/")

(2) Spark Jobs
df: pyspark.sql.dataframe.DataFrame
ID: integer
C_Name: string
C_City: string
C_PhoneNumber: long

```

Display the data

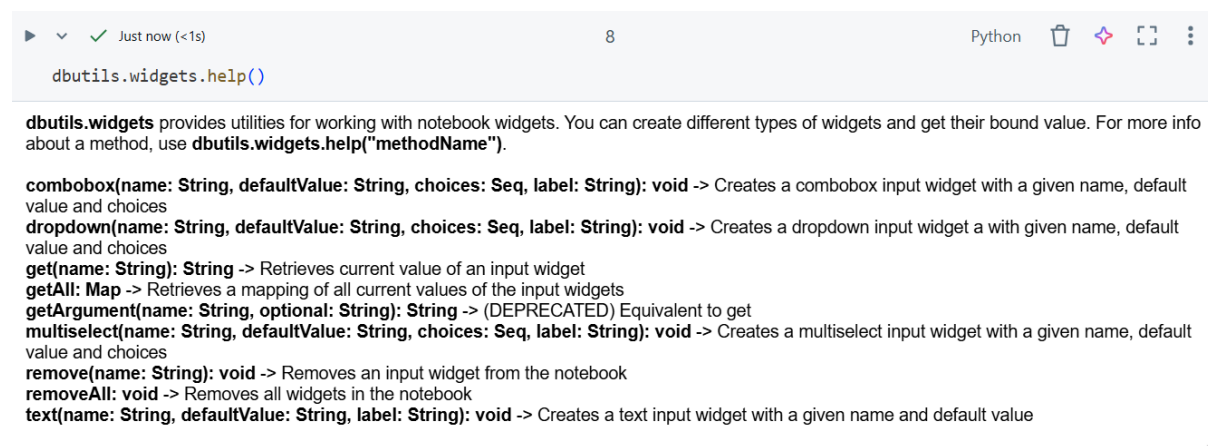


The screenshot shows a Databricks notebook interface. At the top, a code cell contains the command `display(df)`. Below the code, a table is displayed with 7 rows and 5 columns. The columns are labeled `ID`, `C_Name`, `C_City`, and `C_Phonenumner`. The table contains customer data. At the bottom of the table, it says "7 rows | 0.74s runtime". On the right side of the notebook, there is a sidebar with icons for workspace, recent notebooks, and clusters.

	ID	C_Name	C_City	C_Phonenumner
1	1	Deepthi	Sudbury	4169791376
2	2	Rahul	Brampton	4169799087
3	3	Priya	Montreal	4169793456
4	1	Deepthi	Sudbury	4169791376
5	2	Rahul	Brampton	4169799087
6	3	Priya	Montreal	4169793456
7	4	Raj	Halifax	4169797865

How to create a parameter in databricks

These are the commands



The screenshot shows a Databricks notebook interface. The code cell contains the command `dbutils.widgets.help()`. Below the code, the help text for `dbutils.widgets` is displayed. It provides utilities for working with notebook widgets. You can create different types of widgets and get their bound value. For more info about a method, use `dbutils.widgets.help("methodName")`.

combobox(name: String, defaultValue: String, choices: Seq, label: String): void -> Creates a combobox input widget with a given name, default value and choices

dropdown(name: String, defaultValue: String, choices: Seq, label: String): void -> Creates a dropdown input widget a with given name, default value and choices

get(name: String): String -> Retrieves current value of an input widget

getAll: Map -> Retrieves a mapping of all current values of the input widgets

getArgument(name: String, optional: String): String -> (DEPRECATED) Equivalent to get

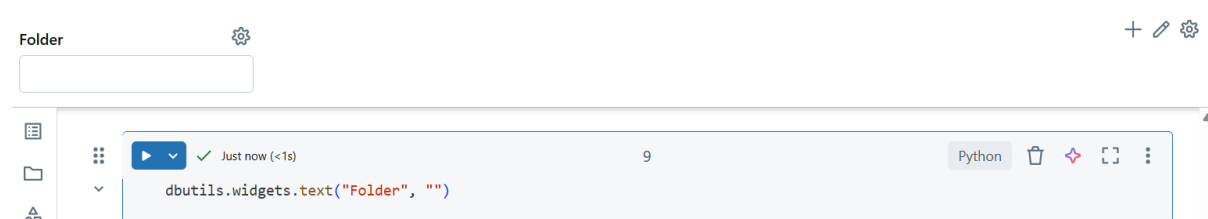
multiselect(name: String, defaultValue: String, choices: Seq, label: String): void -> Creates a multiselect input widget with a given name, default value and choices

remove(name: String): void -> Removes an input widget from the notebook

removeAll: void -> Removes all widgets in the notebook

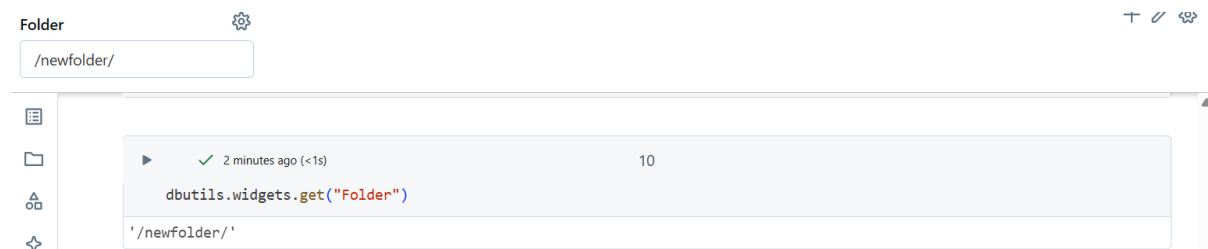
text(name: String, defaultValue: String, label: String): void -> Creates a text input widget with a given name and default value

A folder parameter is created



The screenshot shows a Databricks notebook interface. At the top, there is a "Folder" widget with a settings icon. Below the widget, a code cell contains the command `dbutils.widgets.text("Folder", "")`. The notebook interface shows the code cell and the widget.

Give value in parameter and get the value



The screenshot shows a Databricks notebook interface. At the top, there is a "Folder" widget with a settings icon. The widget now contains the value `/newfolder/`. Below the widget, a code cell contains the command `dbutils.widgets.get("Folder")`. The notebook interface shows the code cell and the widget.

Assigning the value to a variable

▶

✓ 1 minute ago (<1s)

11

```
testpath="/mnt/ewd/"+dbutils.widgets.get("Folder")
print(testpath)
```

/mnt/ewd//newfolder/

Write it to storage account

▶

✓ 1 minute ago (2s)

12

Python

🗑️

✚

🔗

⋮

```
df.write.format("csv").mode("overwrite").save(testpath)
```

▶ (1) Spark Jobs