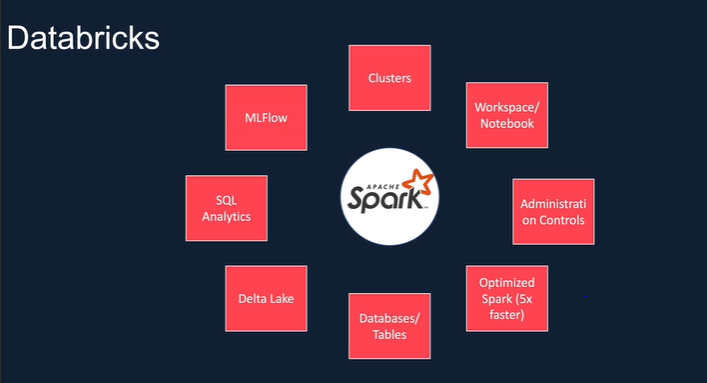
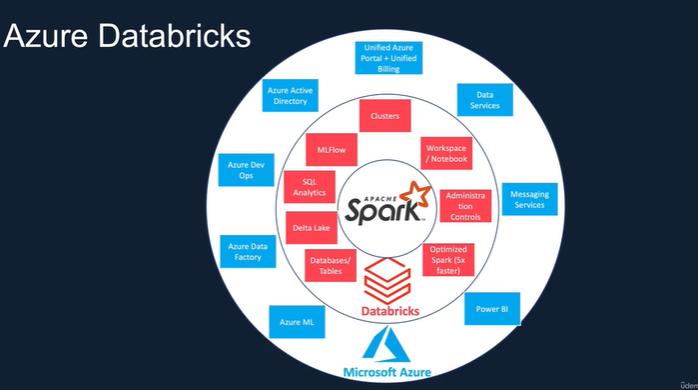
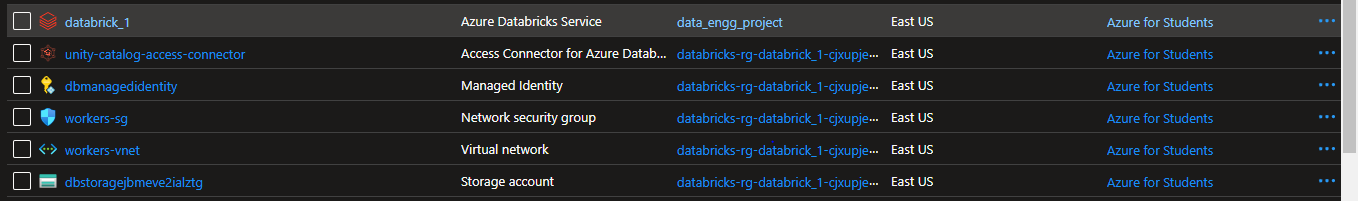
**Session -1:**

Apache spark provides clusters,workspace/Notebook ,Adminstration control,optmized spark,database/tables,delta lake,sql analysis,MLFlow.



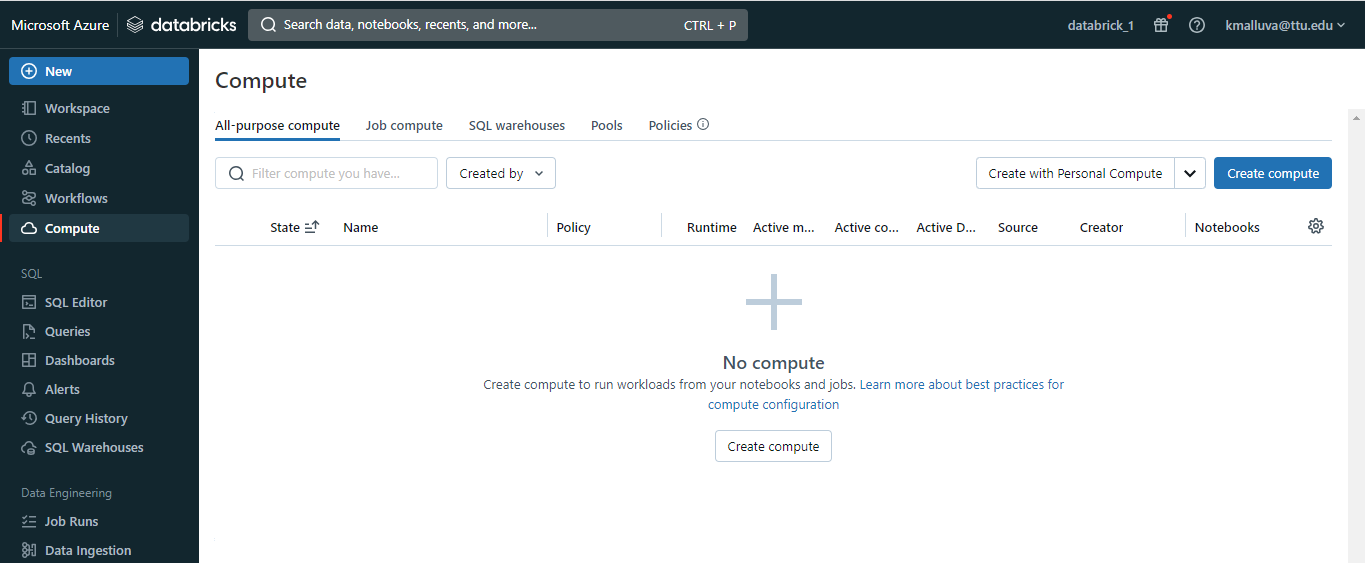


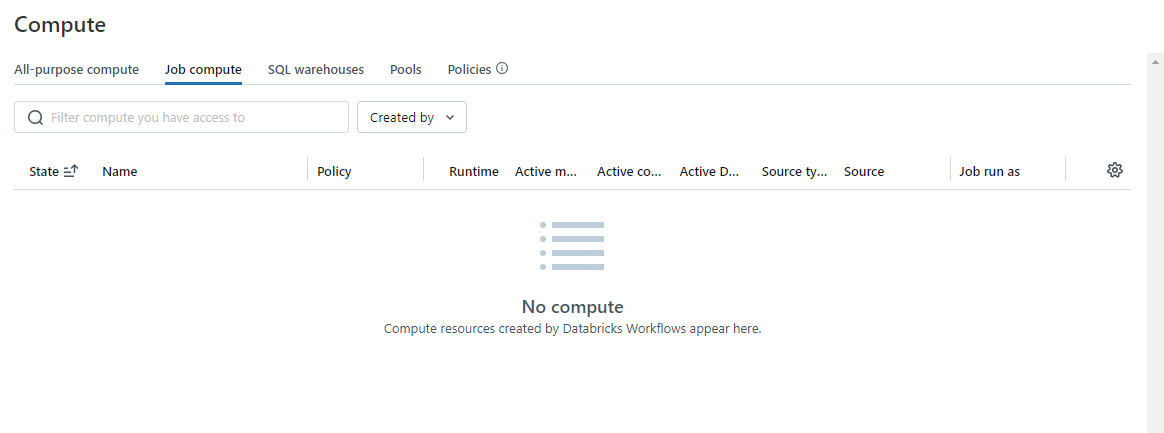


There are two types of cluster

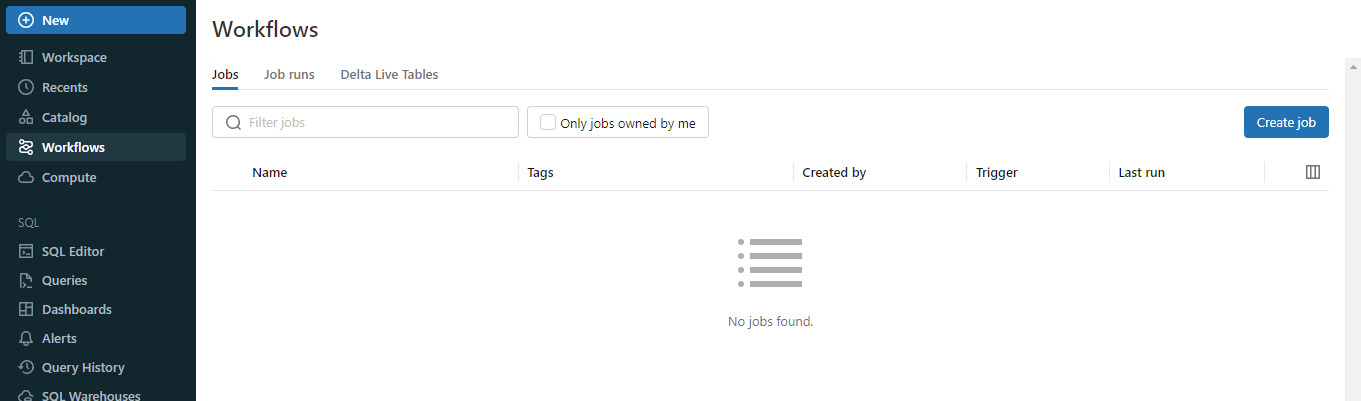
Where all purpose cluster can be created but job cluster cannot be created.

When there a workflow we can create a job cluster ,after end of the job it will get destroied .



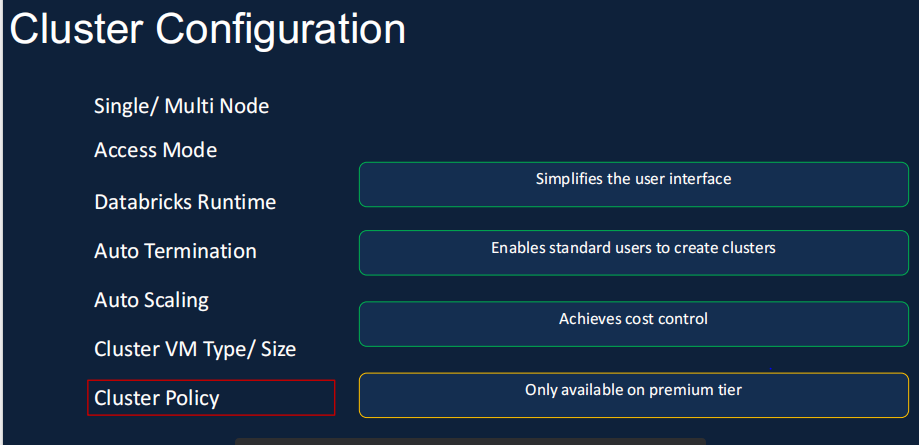


In workflow you can create a job:

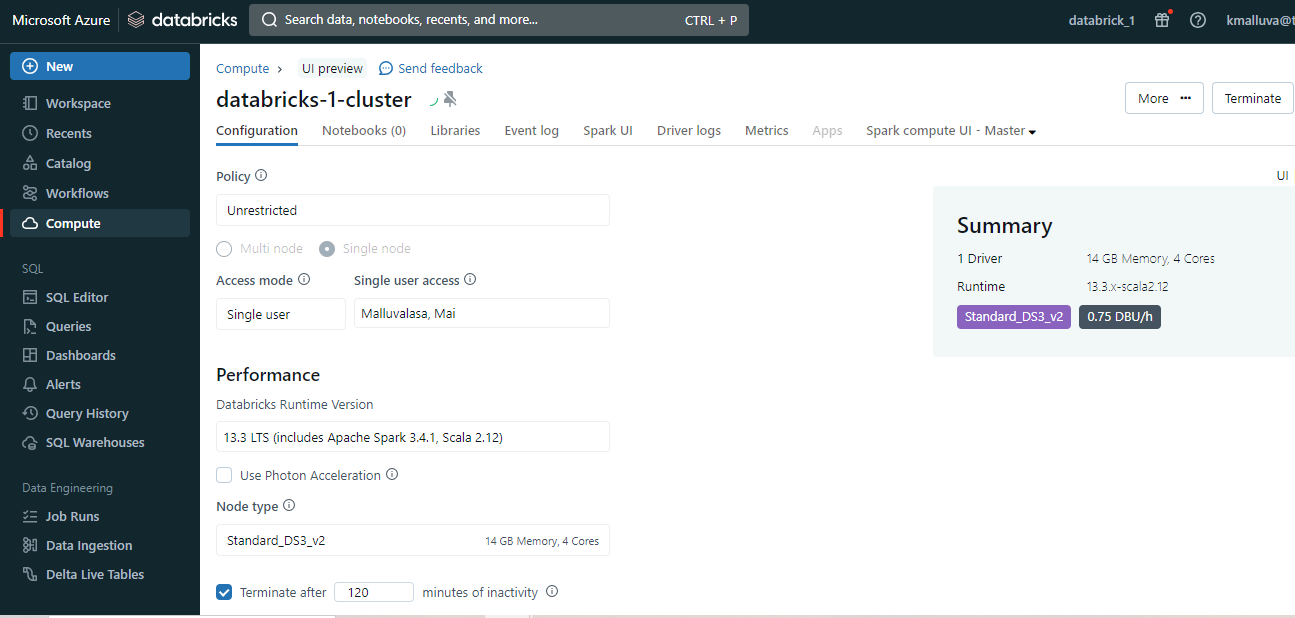


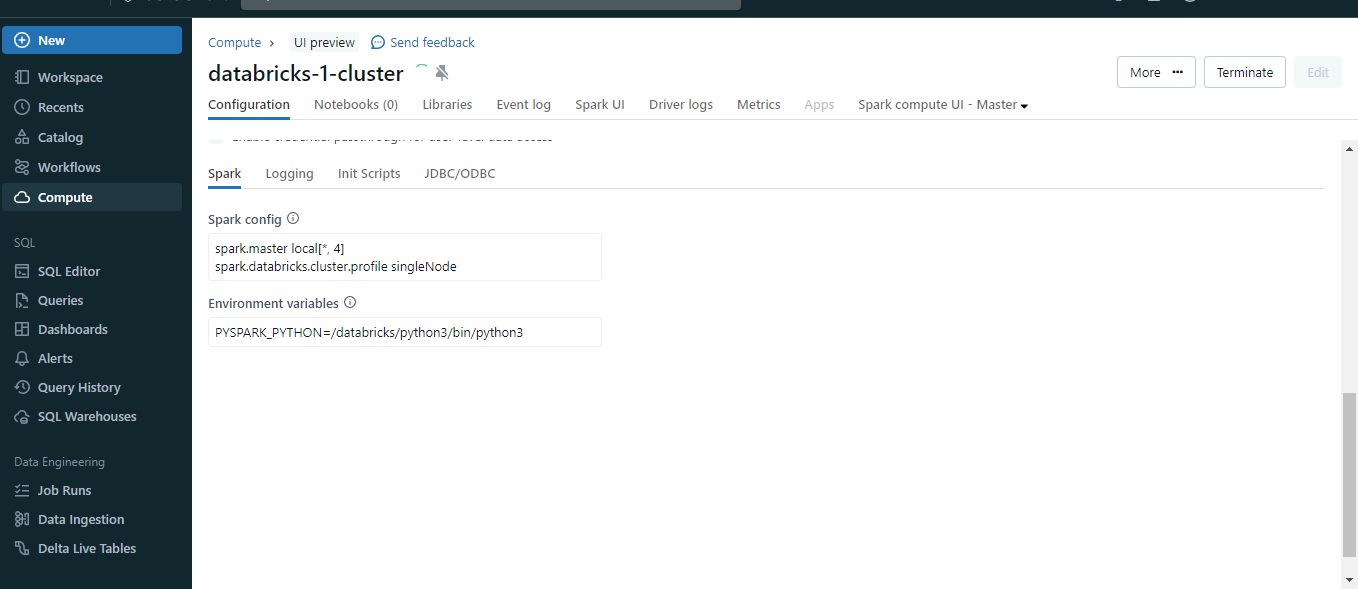
**SESSION -2**

Cluster configuration:

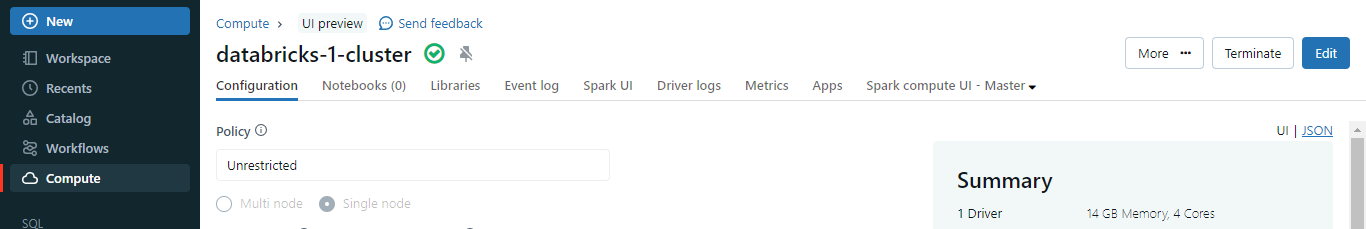


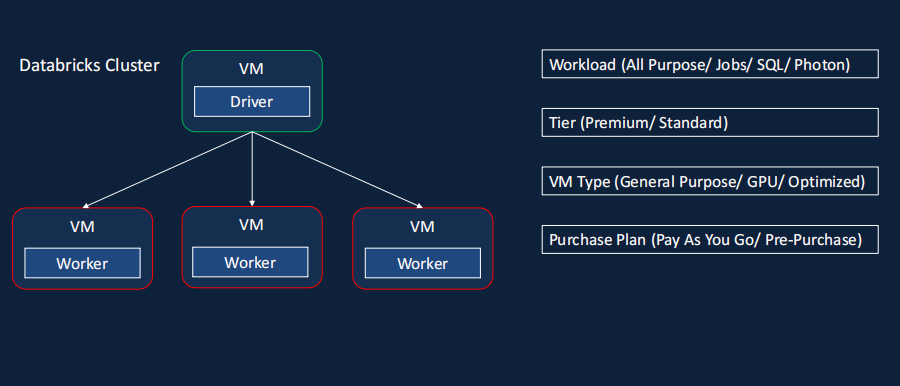
Creating a cluster:

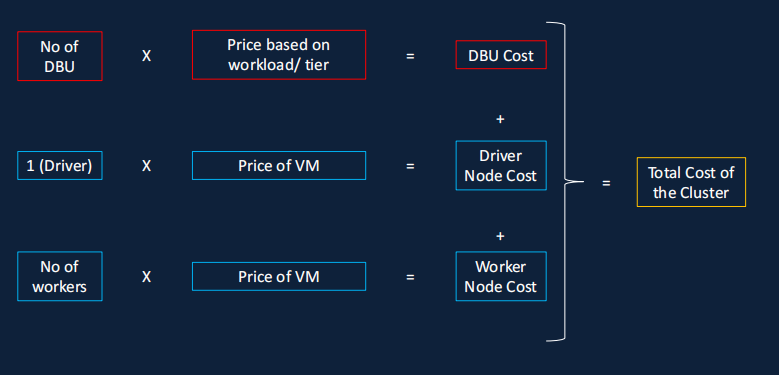




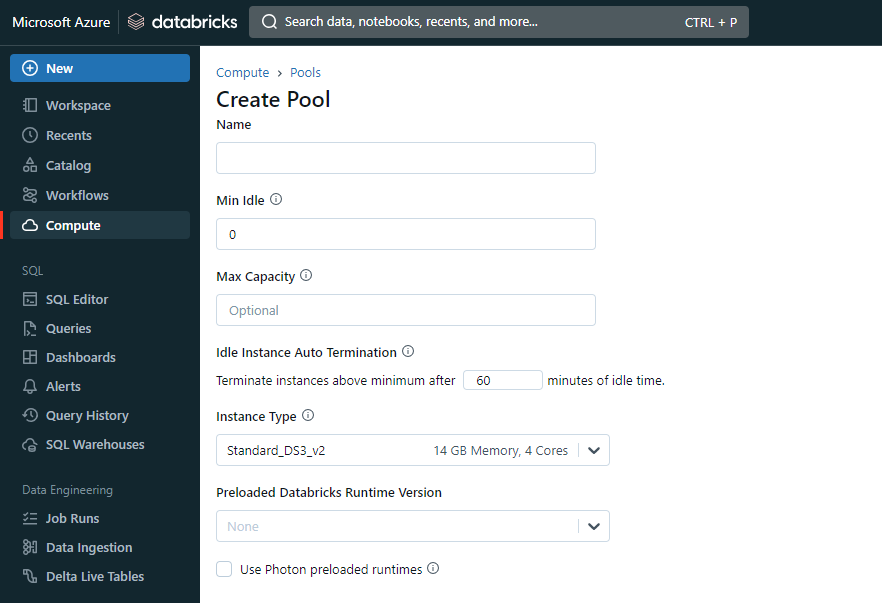
If cluster is created we can see green trick mark

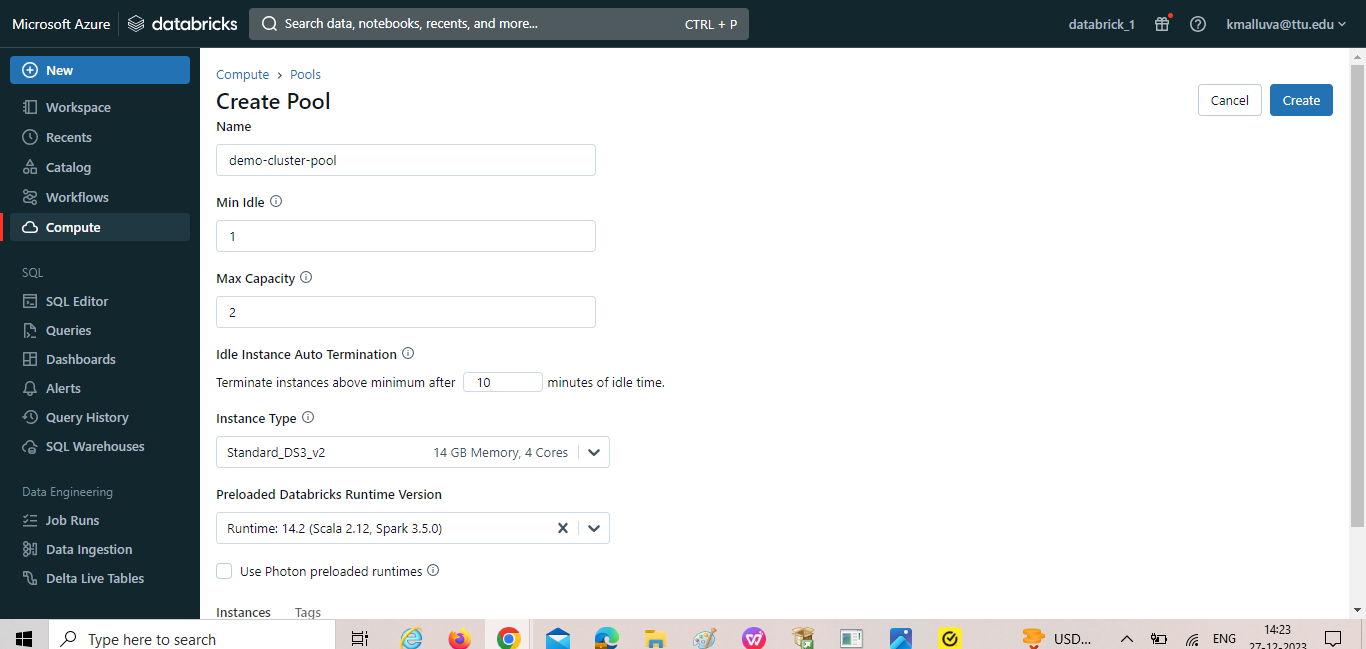


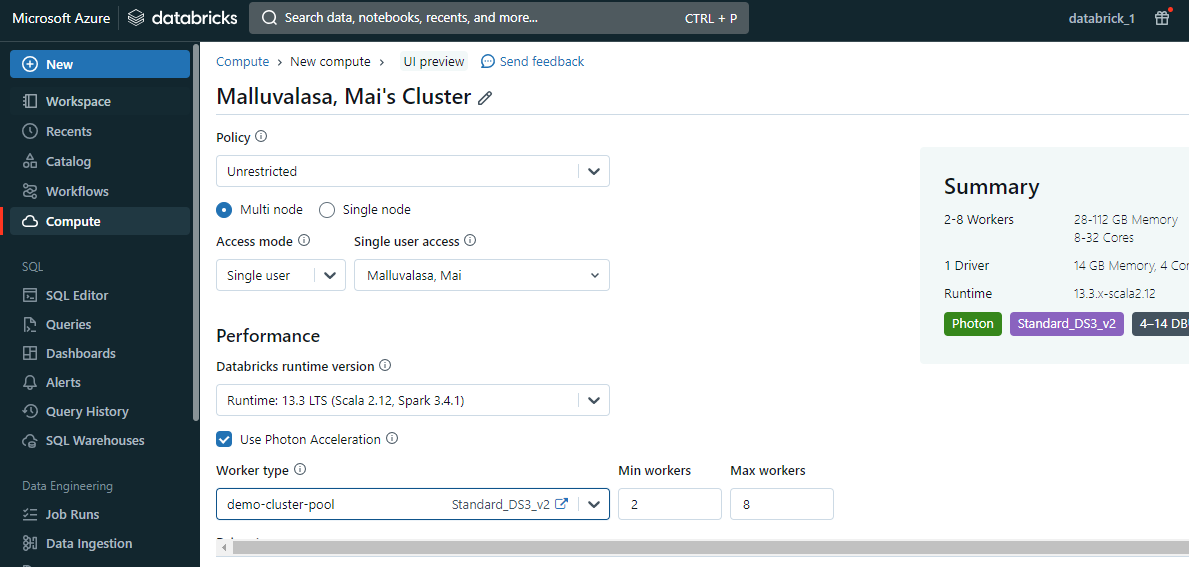




For creating a pool:

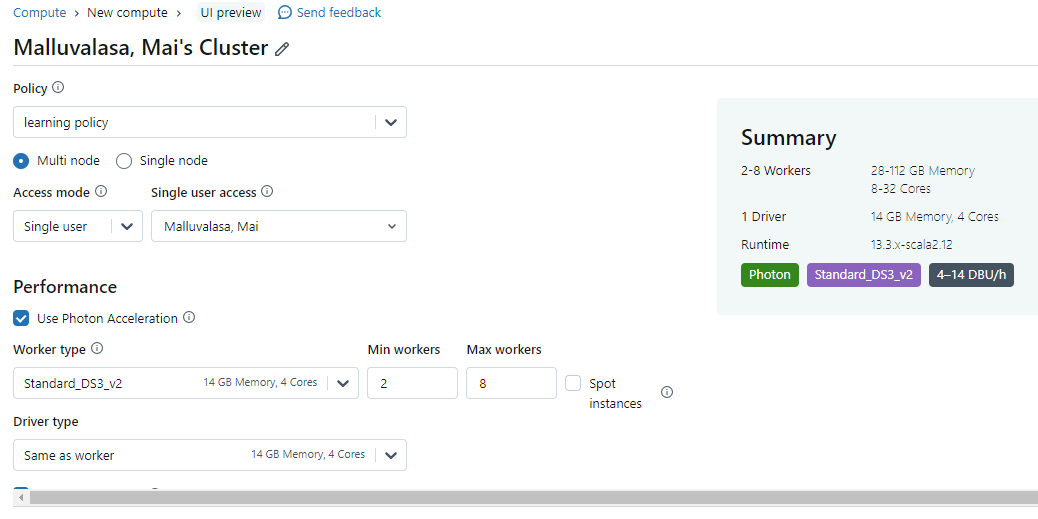
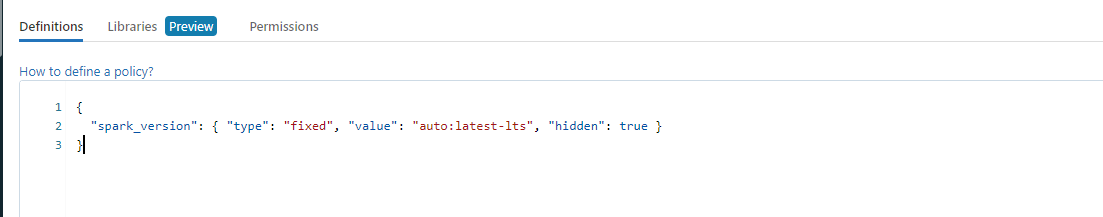
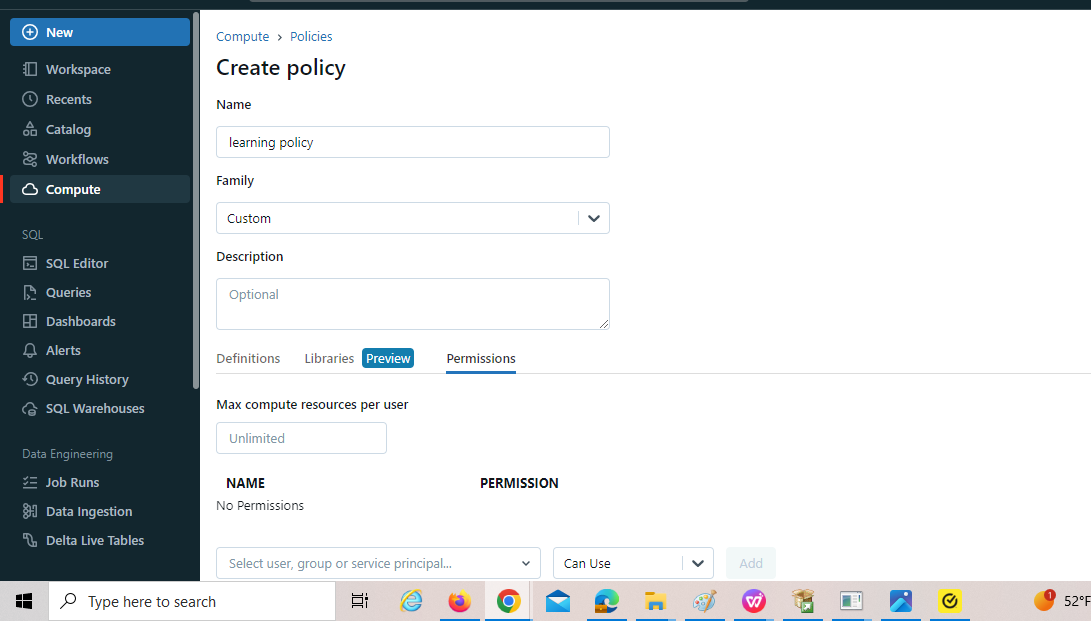
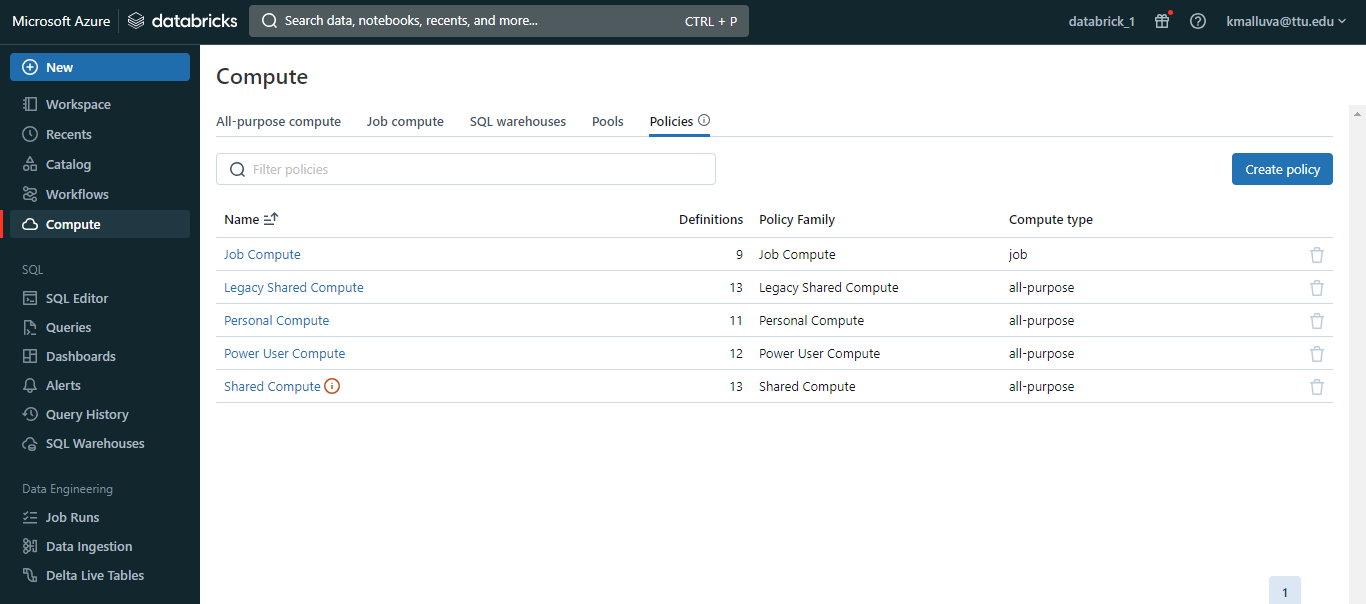




Creating new cluster by adding worker type as demo-cluster 

Where the pool speed up the creation of cluster.

Policy:

{

  "spark\_conf.spark.databricks.cluster.profile": {

    "type": "fixed",

    "value": "singleNode",

    "hidden": true

  },

  "node\_type\_id": {

    "type": "allowlist",

    "values": [

      "Standard\_DS3\_v2",

      "Standard\_D3\_v2",

      "Standard\_DS12\_v2",

      "Standard\_D12\_v2"

    ],

    "defaultValue": "Standard\_DS3\_v2"

  },

  "autotermination\_minutes": {

    "type": "fixed",

    "value": 20

  },

  "spark\_version": {

    "type": "fixed",

    "value": "auto:latest-lts",

    "hidden": true

  },

  "cluster\_type": {

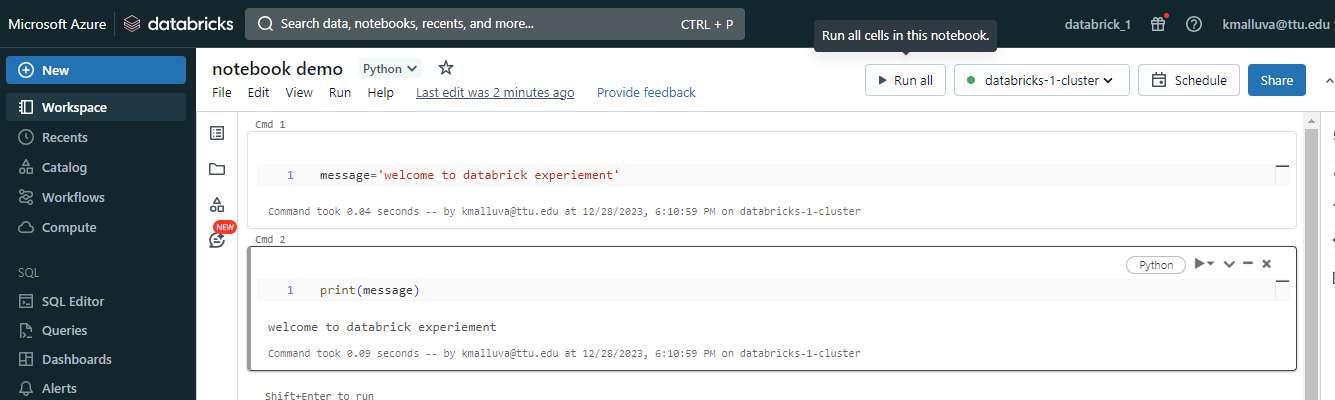
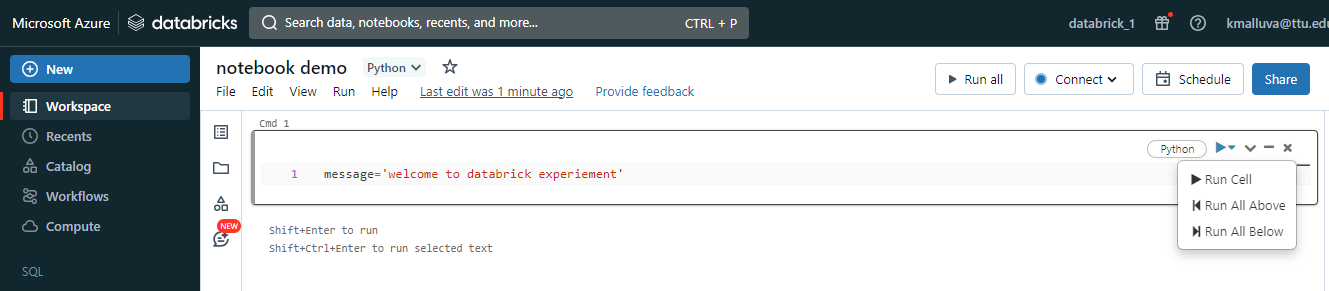
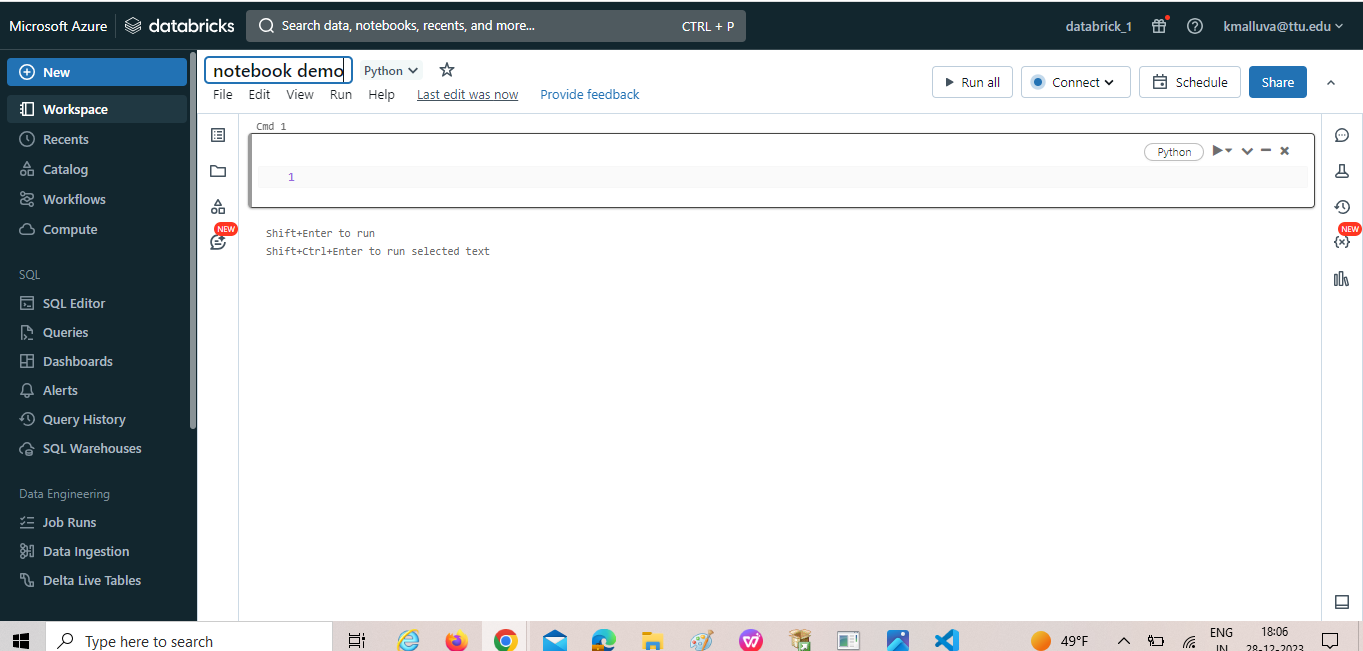
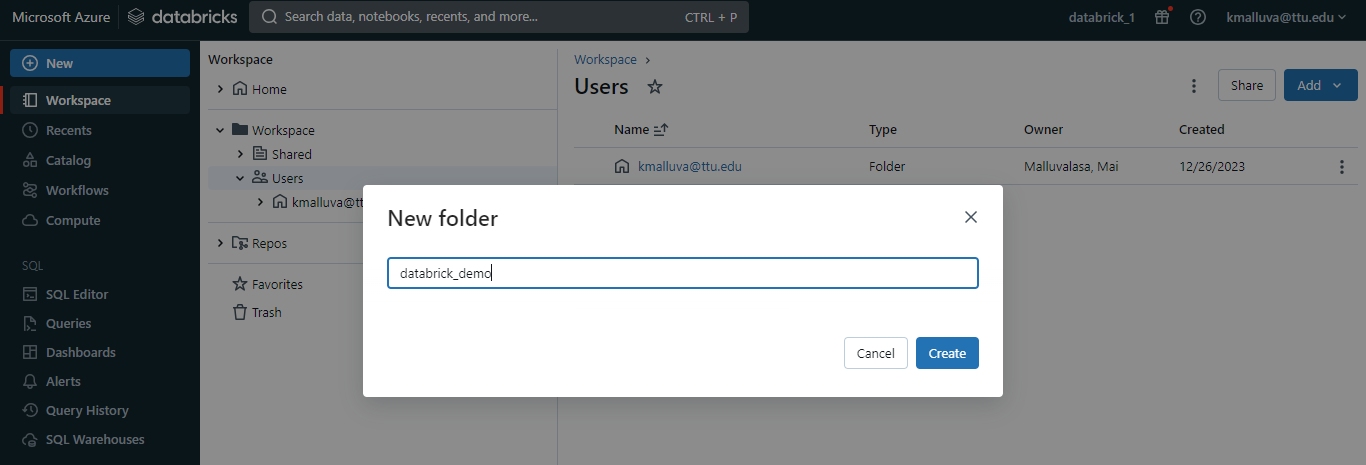
    "type": "fixed",

    "value": "all-purpose"

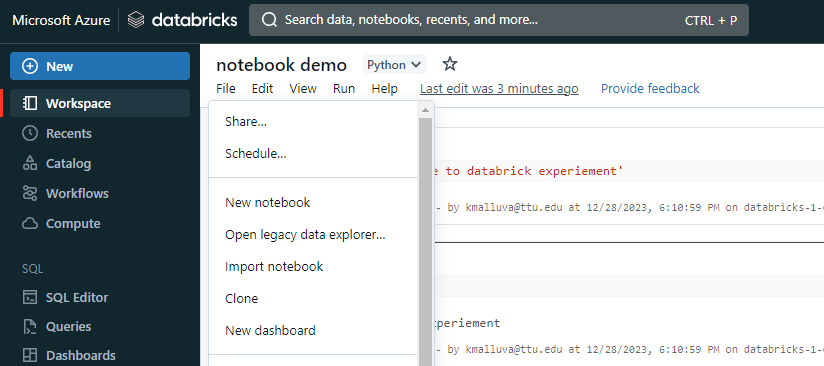
  }

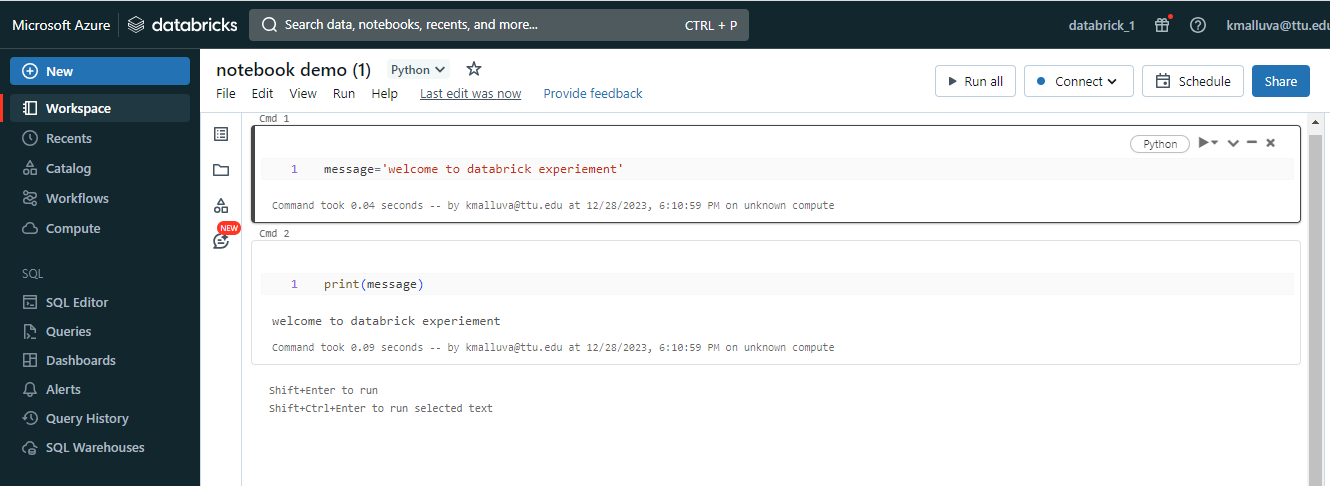
}

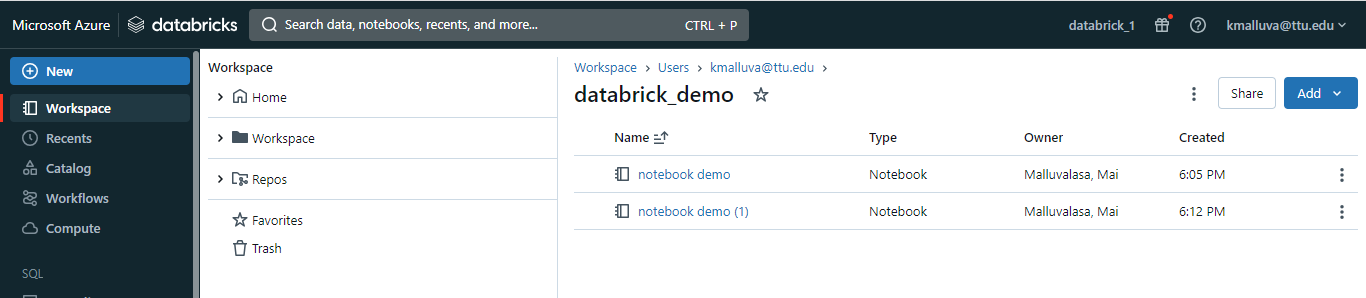
Create notebook:

to duplicate the notebook:

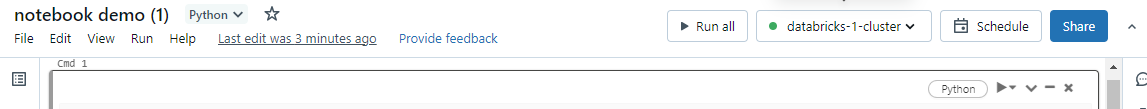
Click on clone

duplicate notebook:



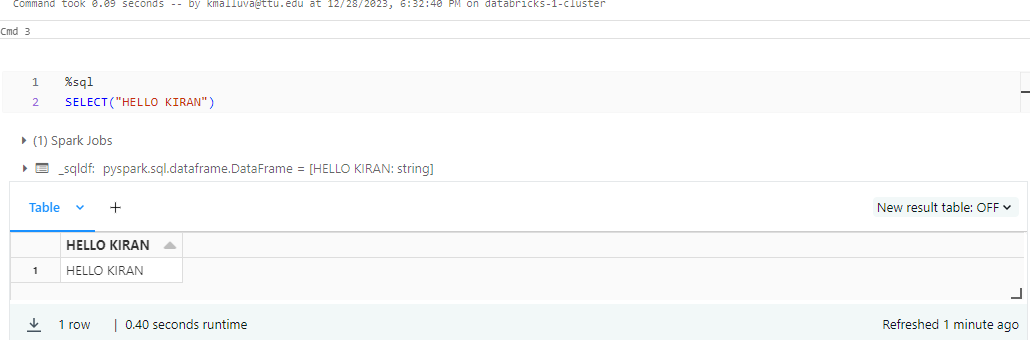


Connected to cluster:

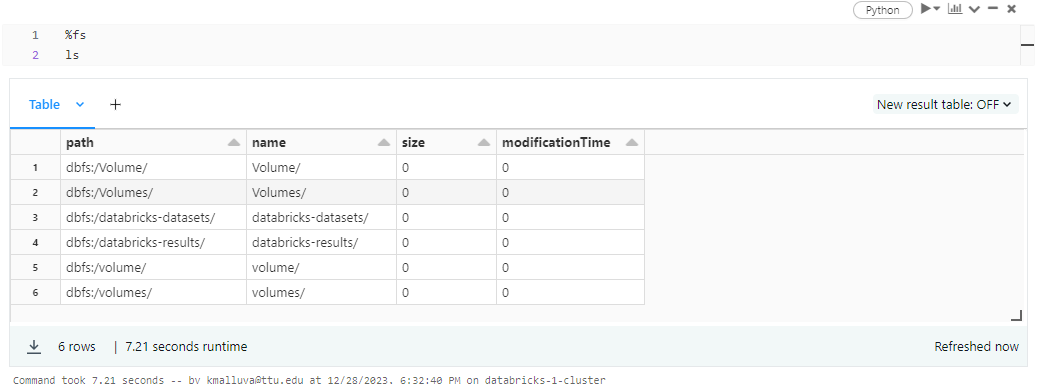


Magic commands:

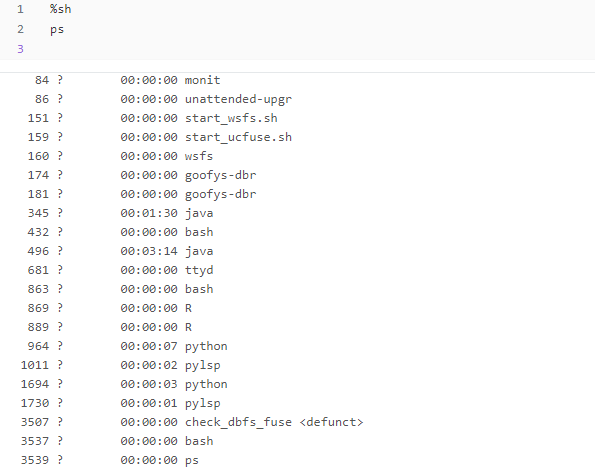
It is used for moving form one lang to another lang in the notebook (eg: we are using python instead of python you want to move to scala lang this command is used.



Magic command for listing out file system:

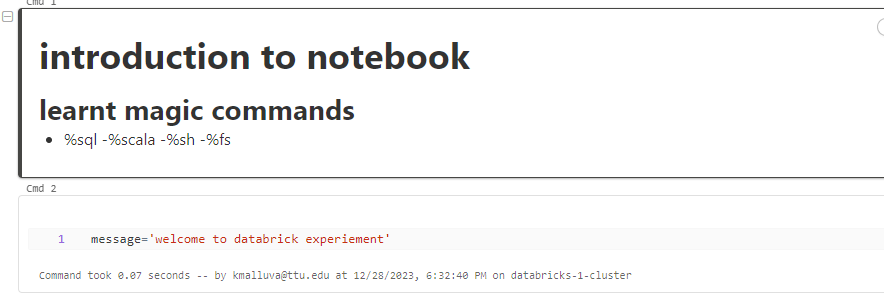


To now what processor are running:



%md

Cell is modified as document

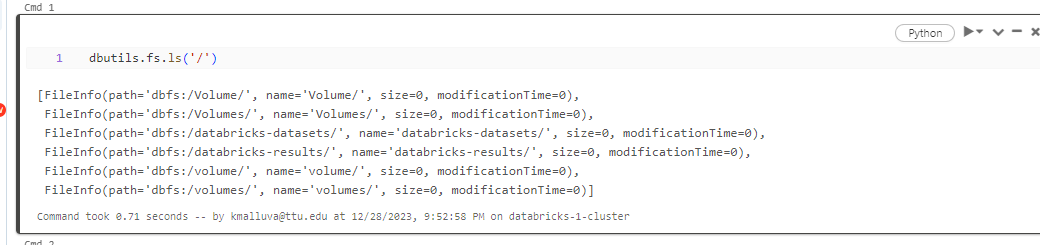


Utilities:



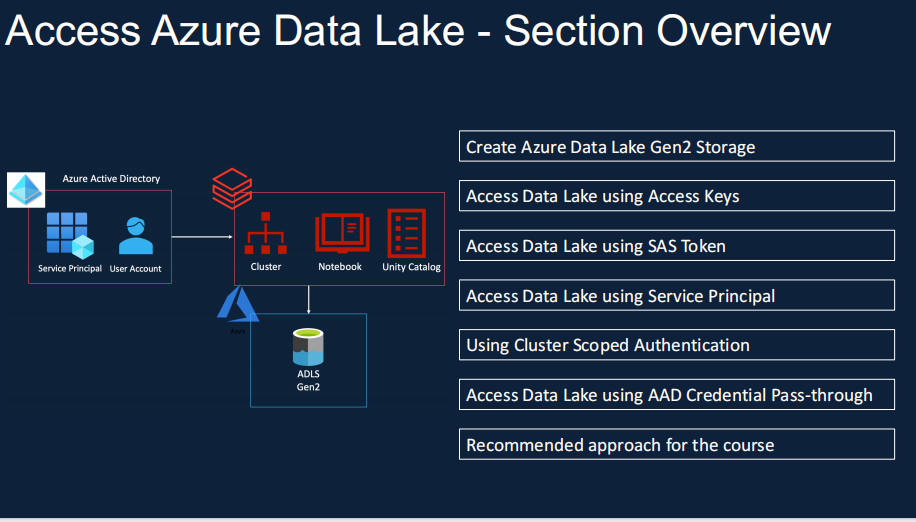
In python:

Dbutils.fs.ls(‘/ ’)

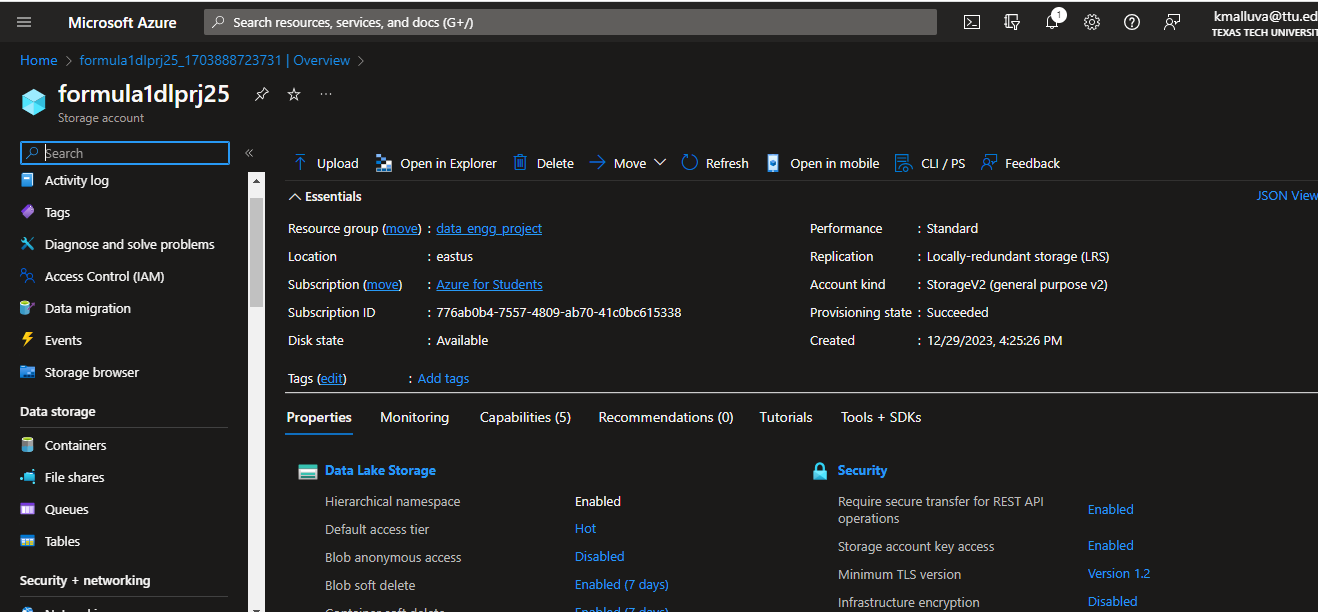


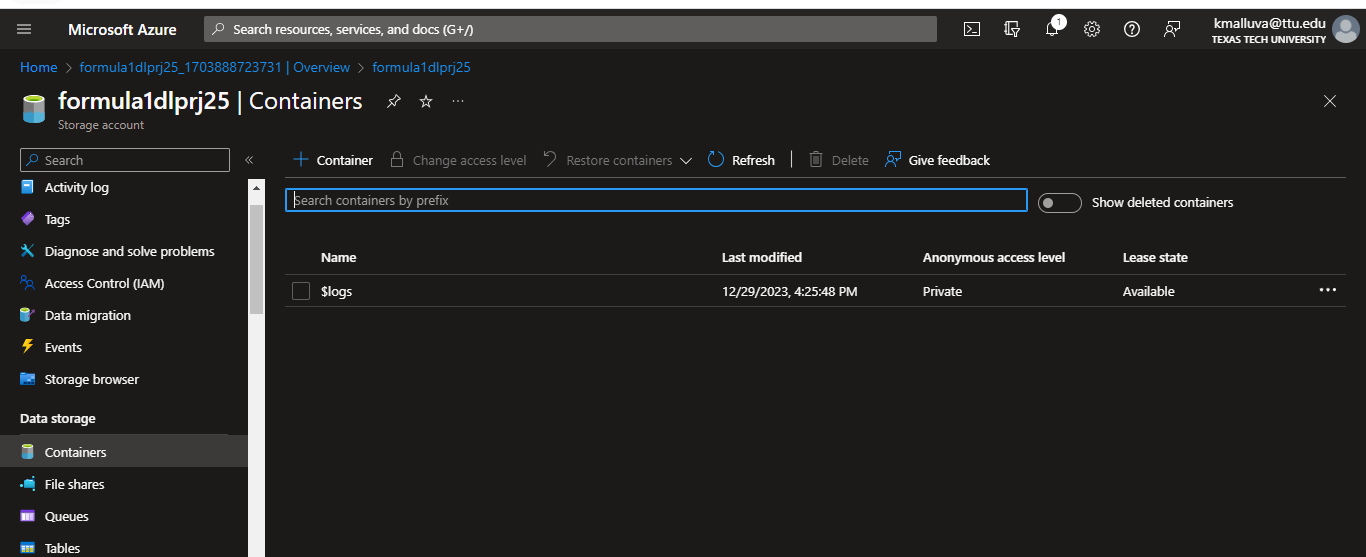
**Next session:**

**Session-3**

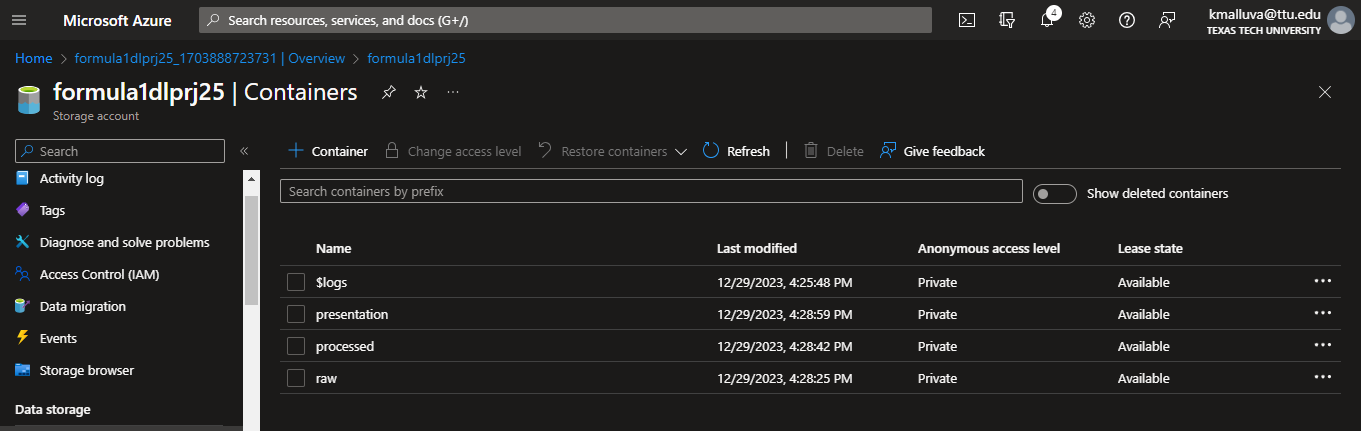


Create a storage account :

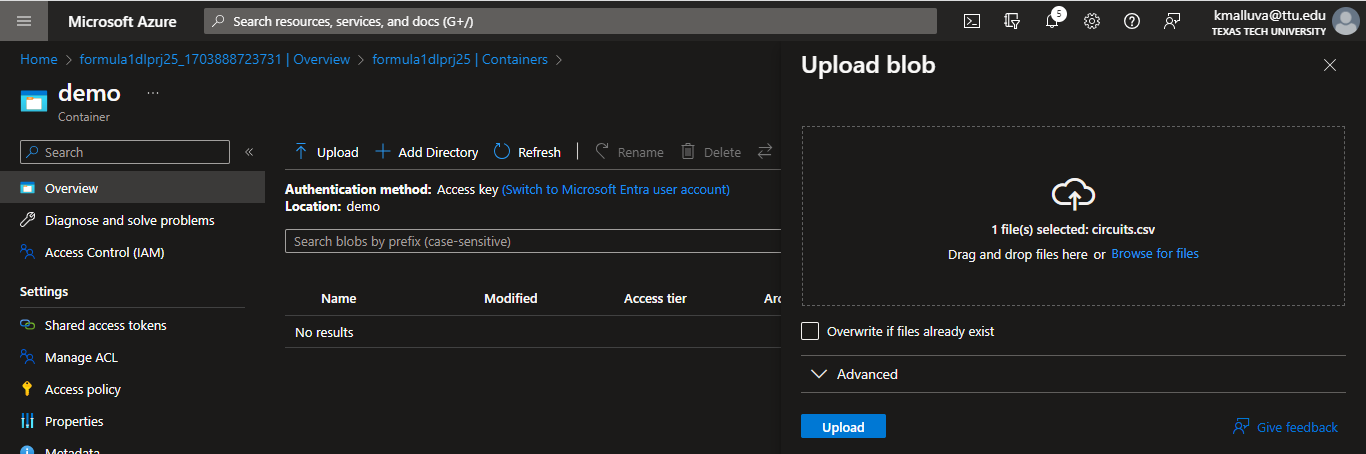


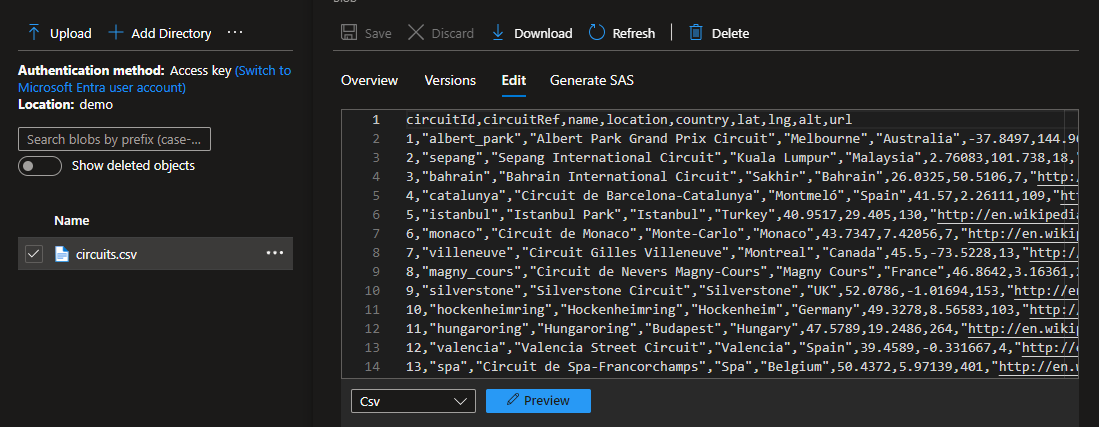


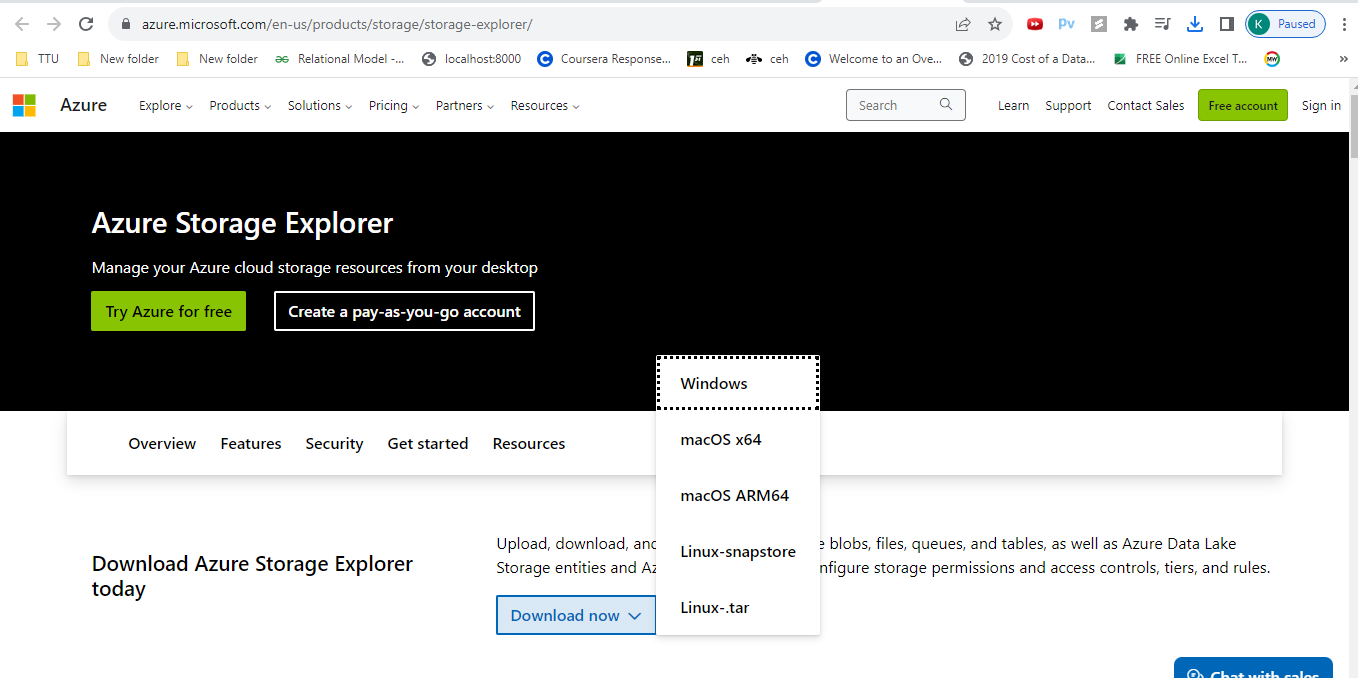
Created a 3 containers:

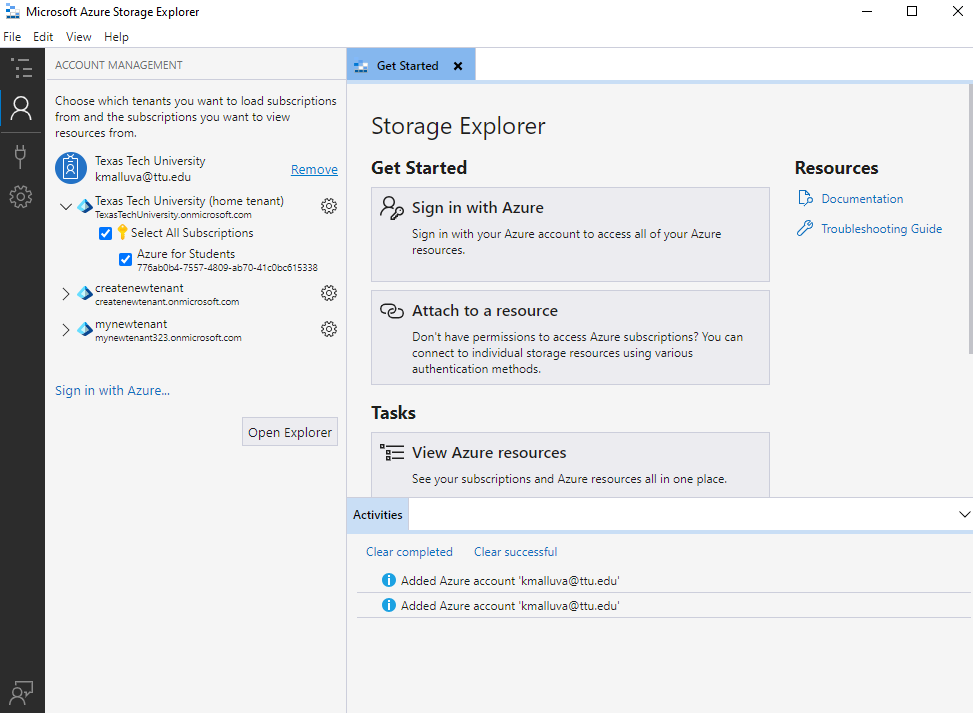


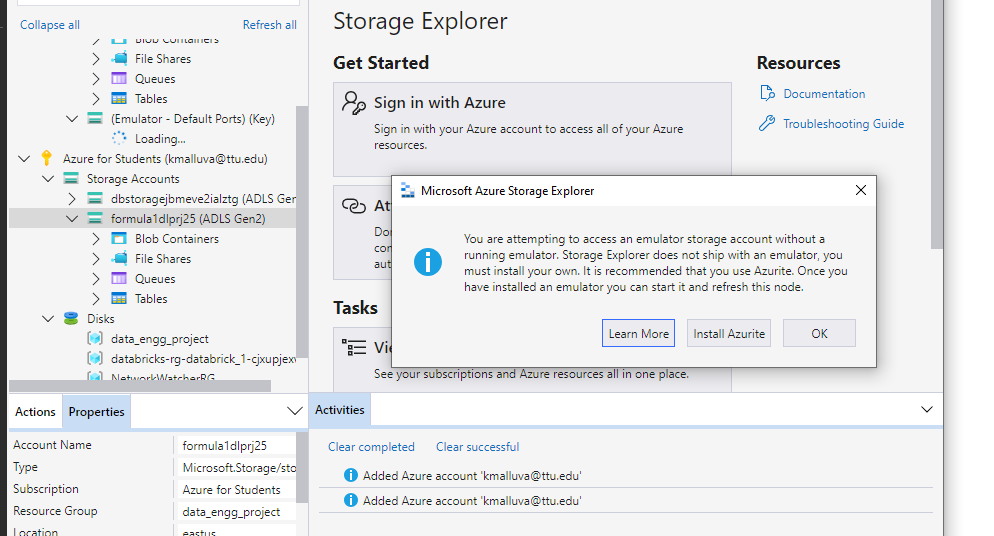
Created a demo container and uploaded .csv file :

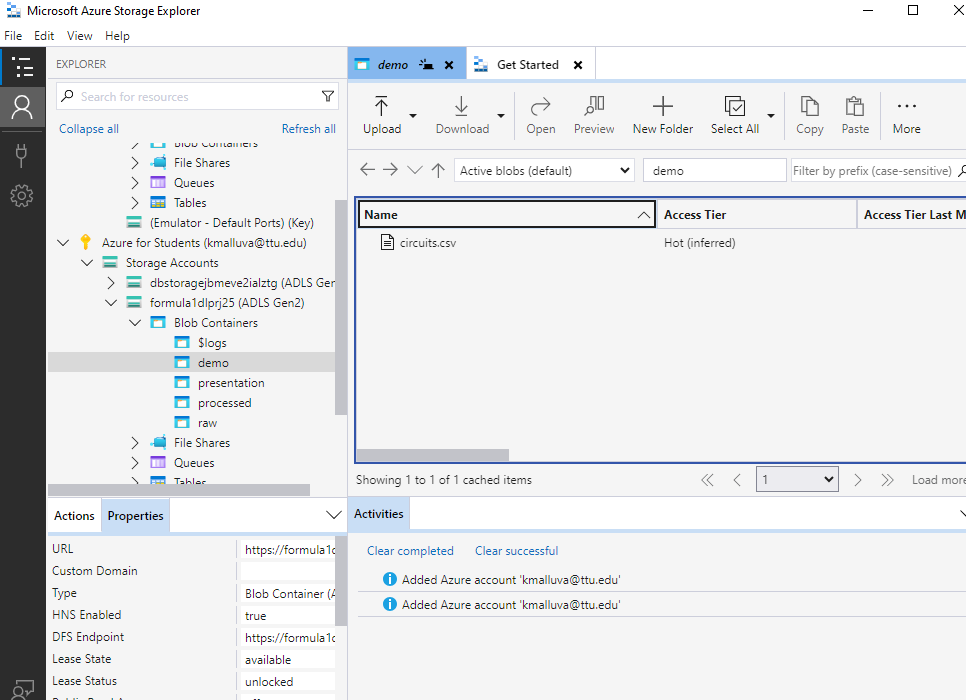




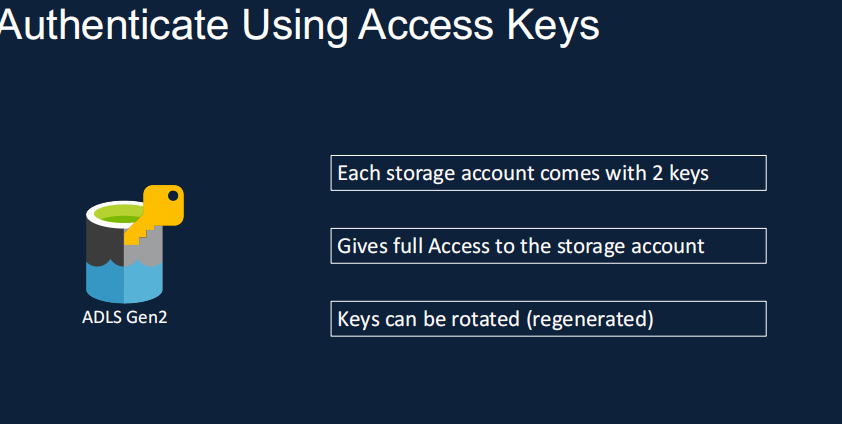




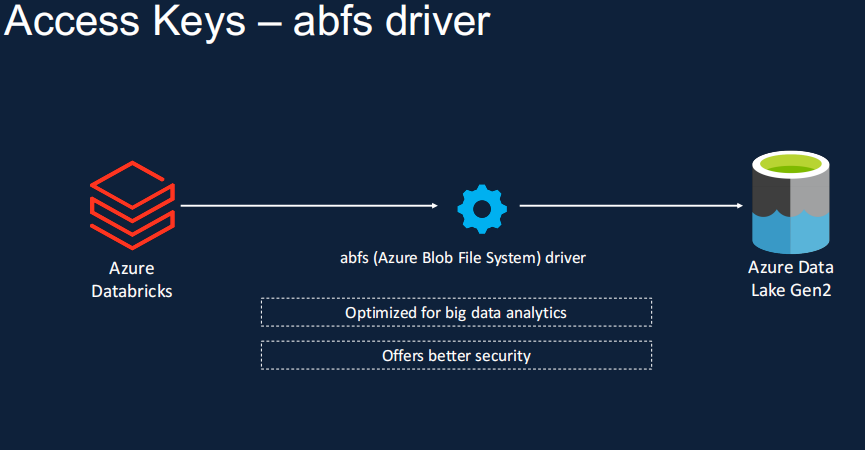


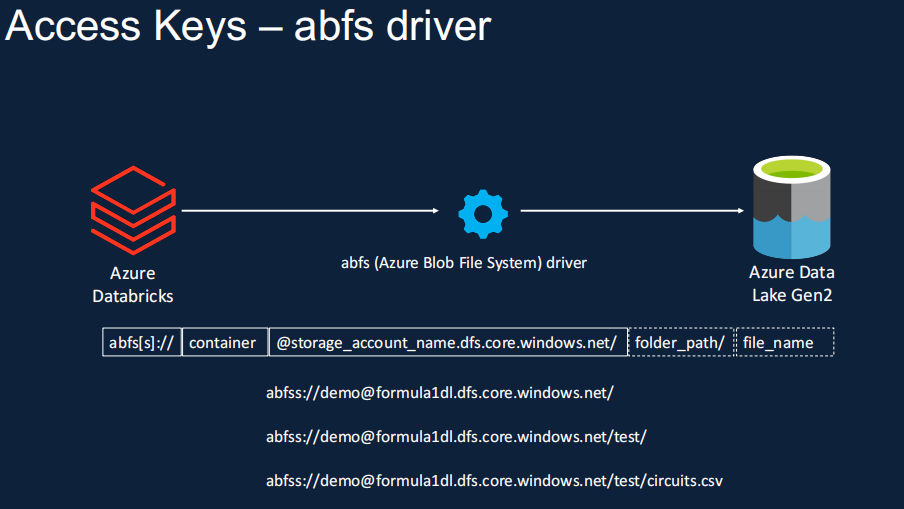


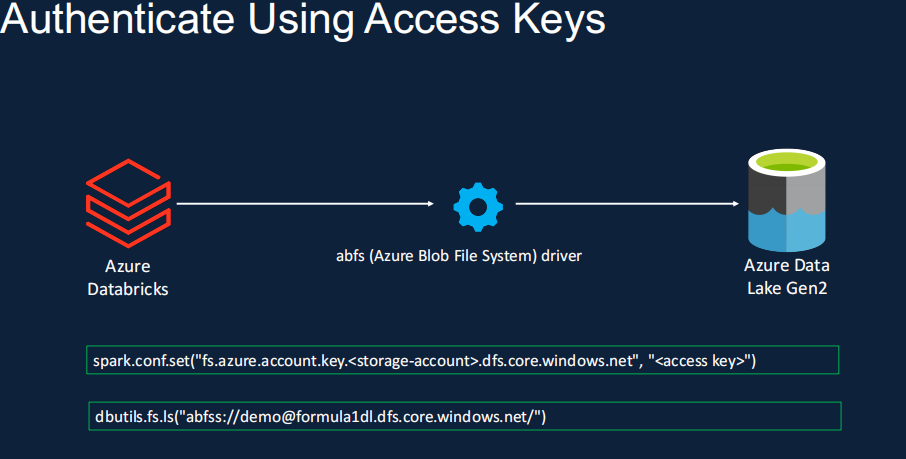
Using access keys:



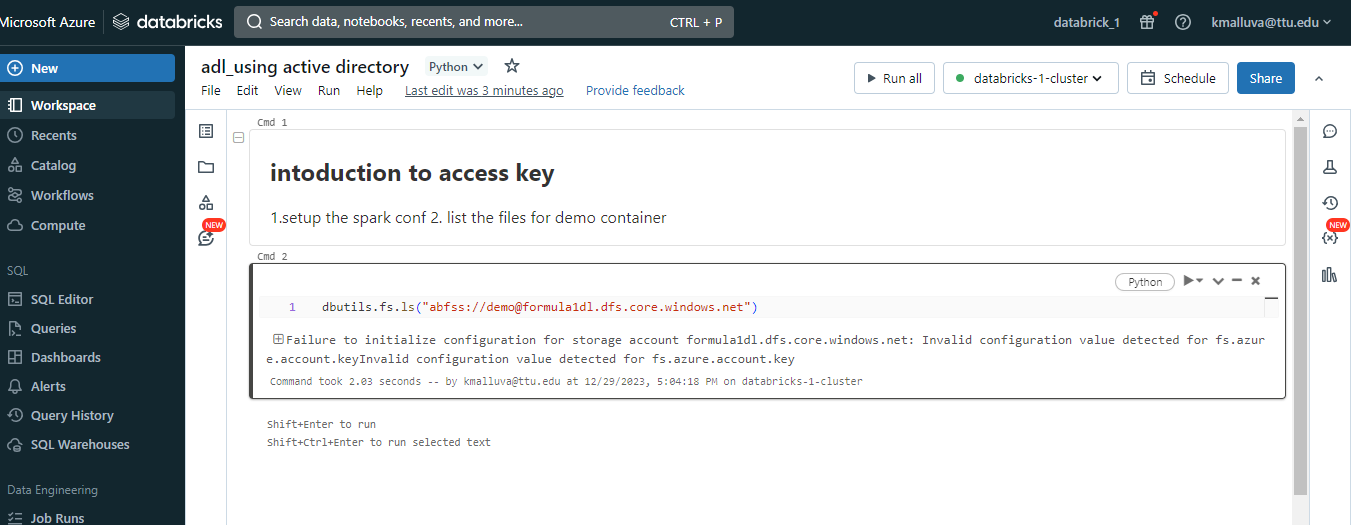




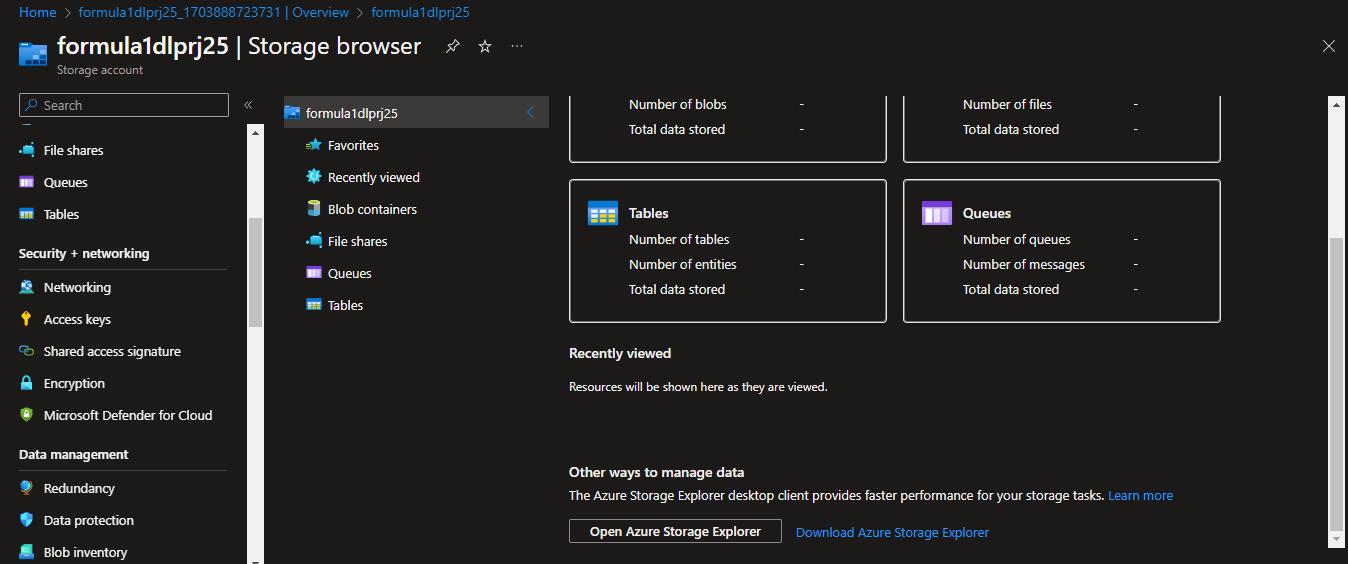


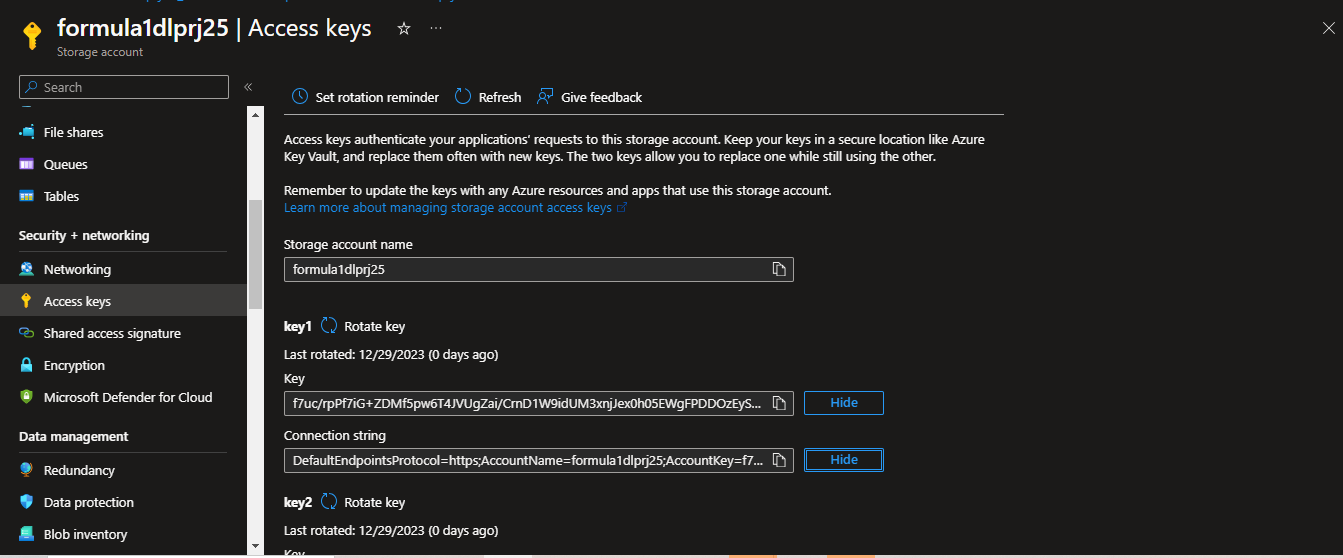


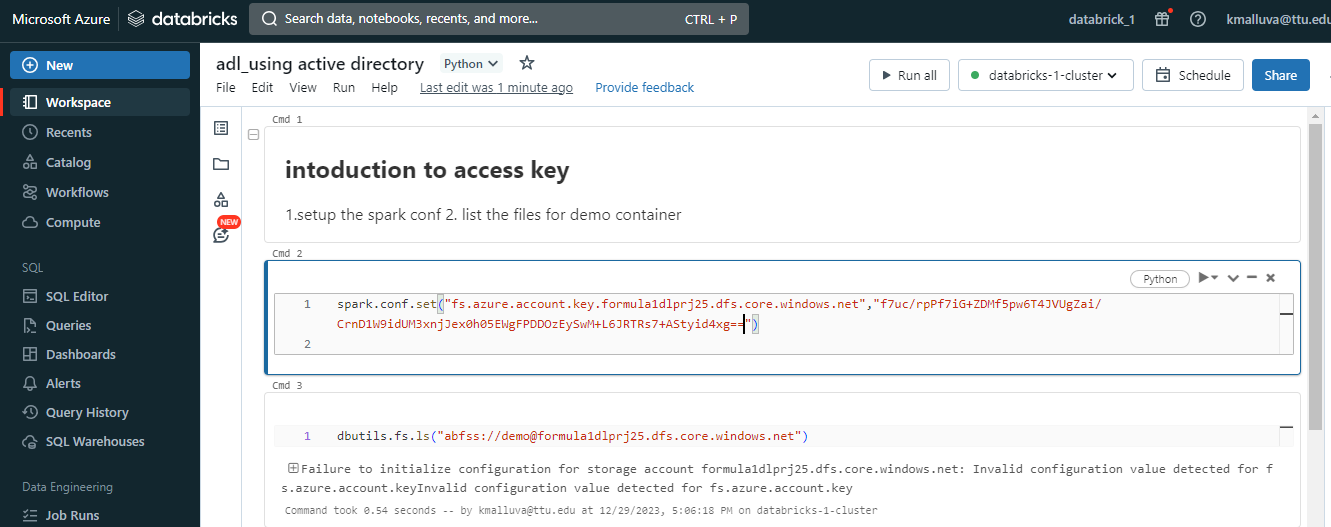
Failed:

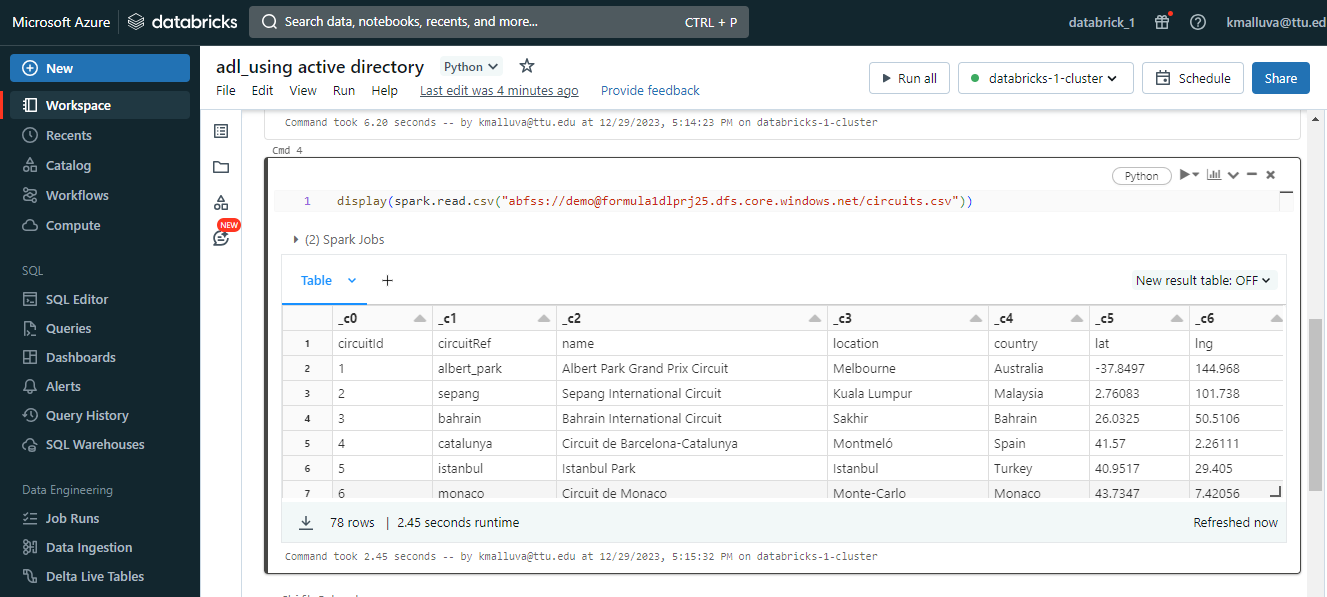


So go to access key









Shared acess signature:

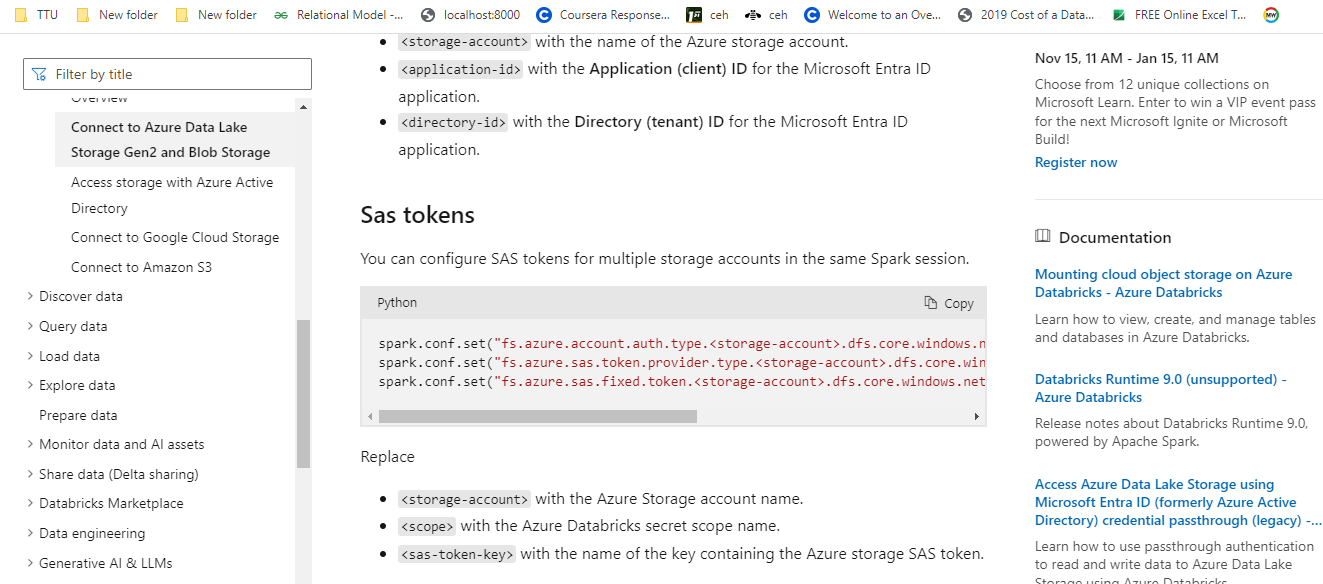


Azure -----------------------------signed access signarture -------------------------------azure data gen 2

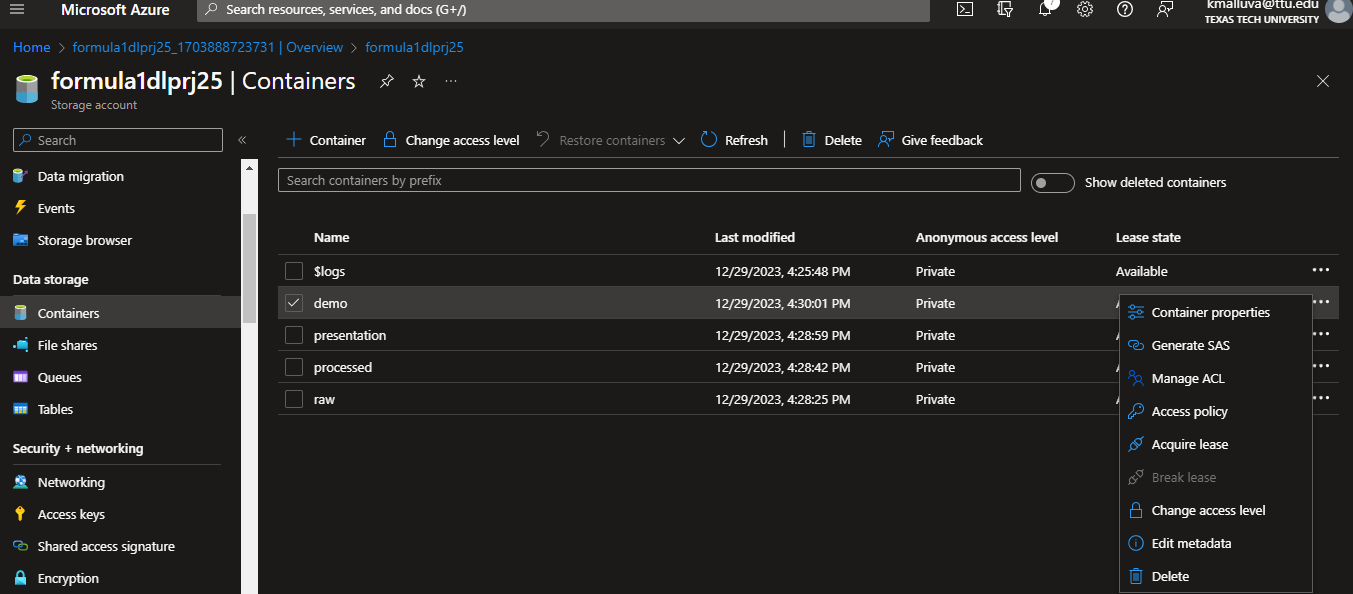
databricks

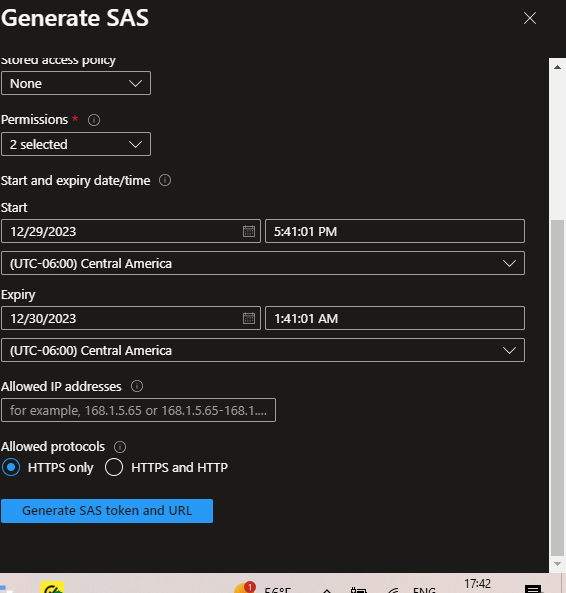
<https://learn.microsoft.com/en-us/azure/databricks/connect/storage/azure-storage#access-azure-data-lake-storage-gen2-or-blob-storage>

Copy from this link :

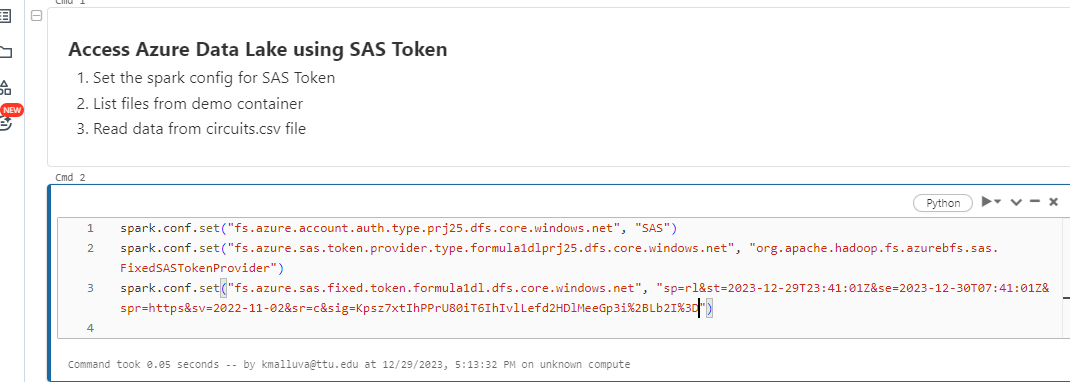


Click on generate sas:



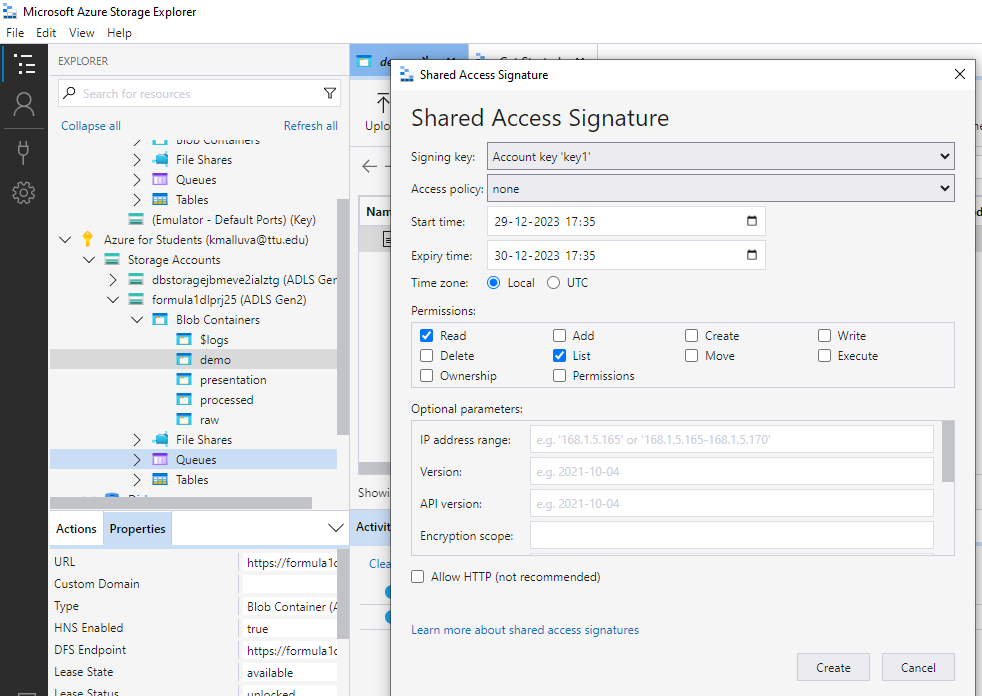


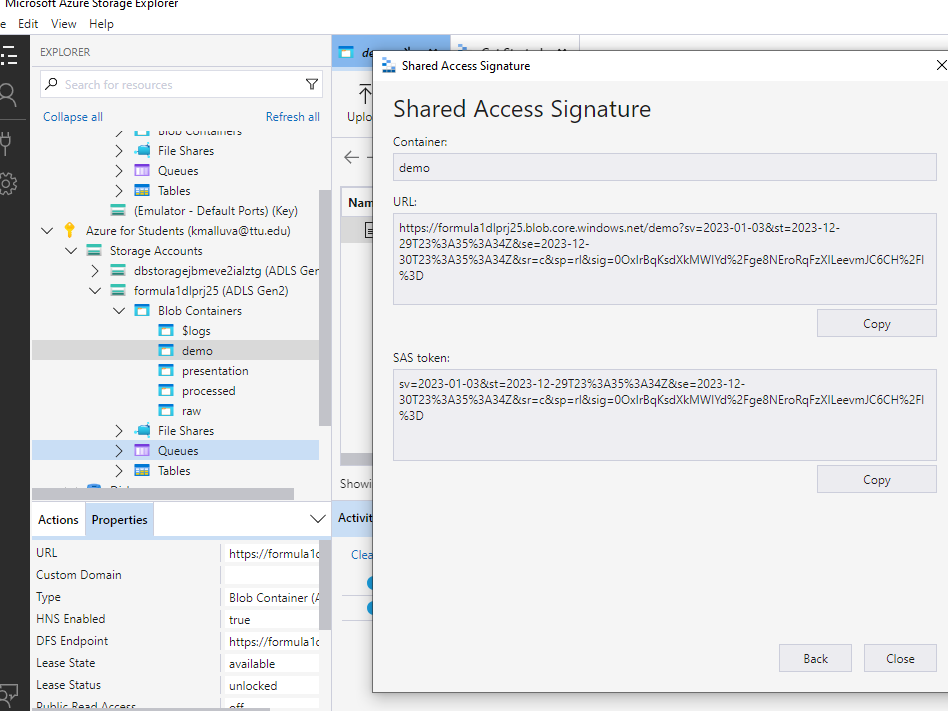
Copied the sas token and pased it:



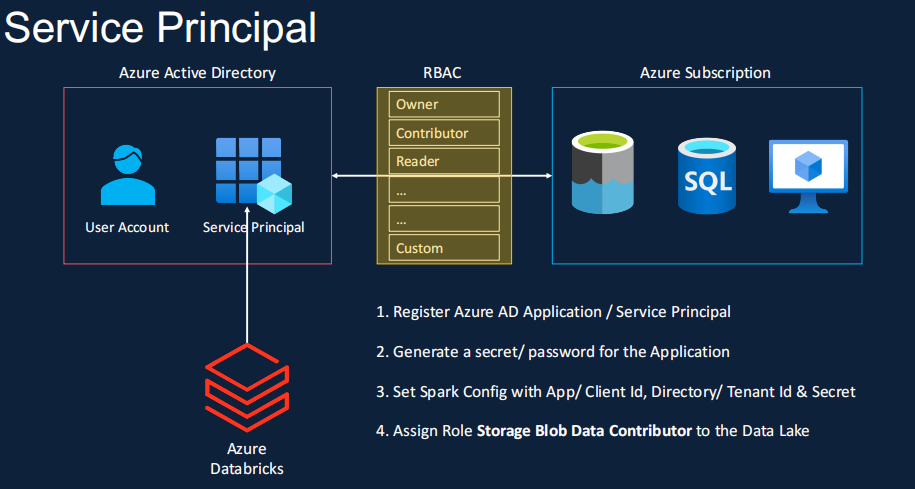
Other method to get SAS TOKEN:

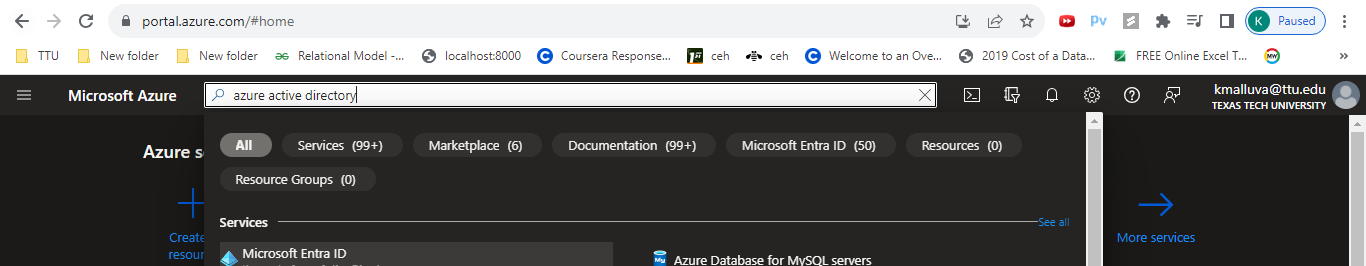
Go to explorer storage account:

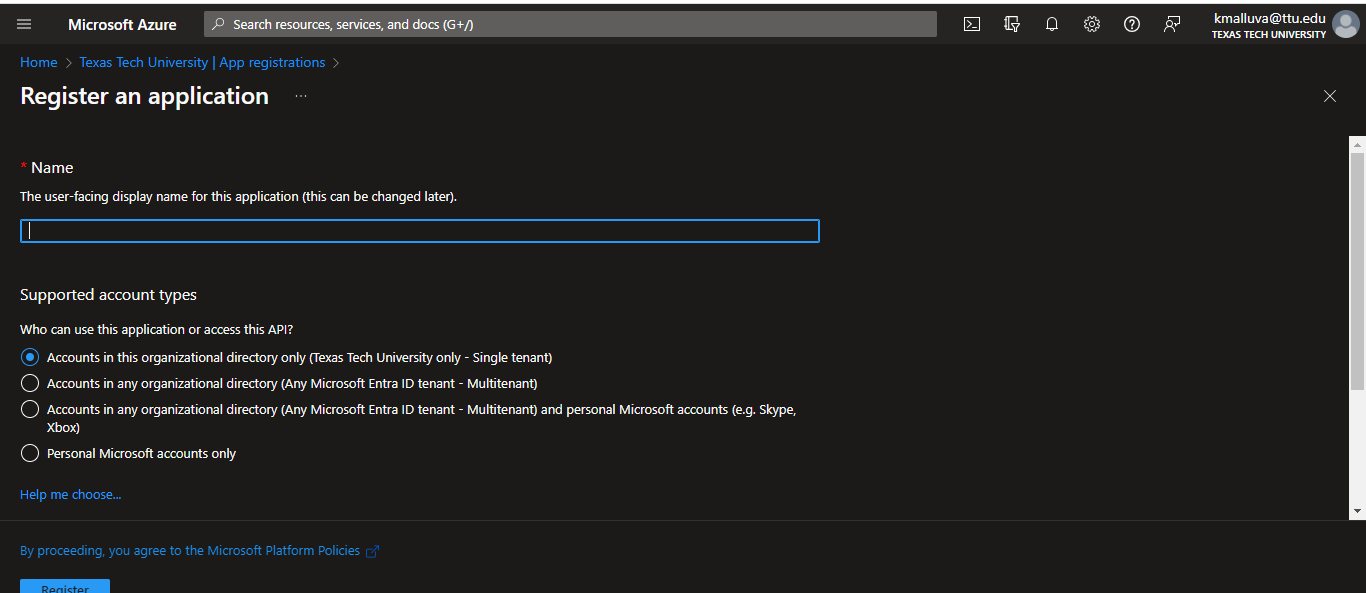


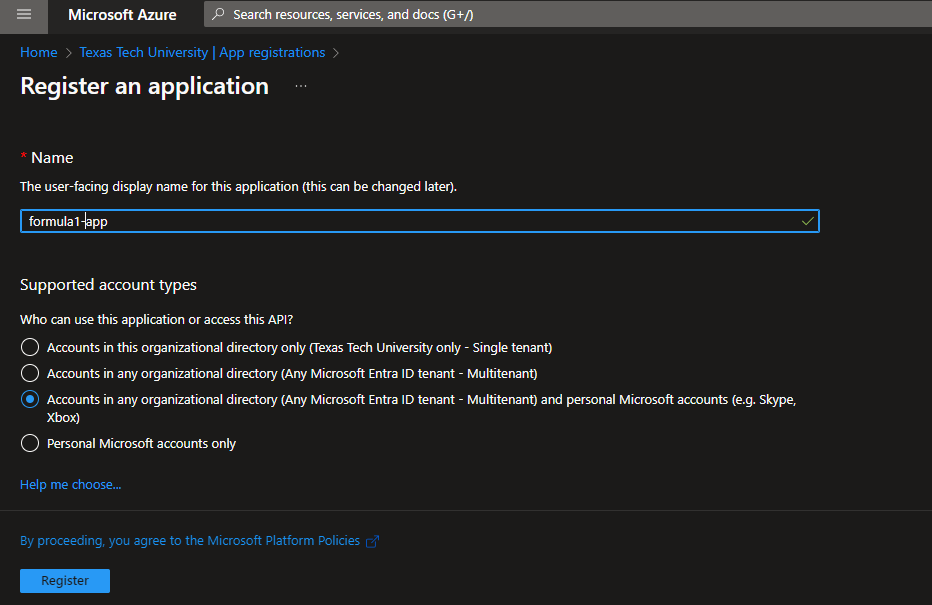


USING SERVICE PRINCIPLE:

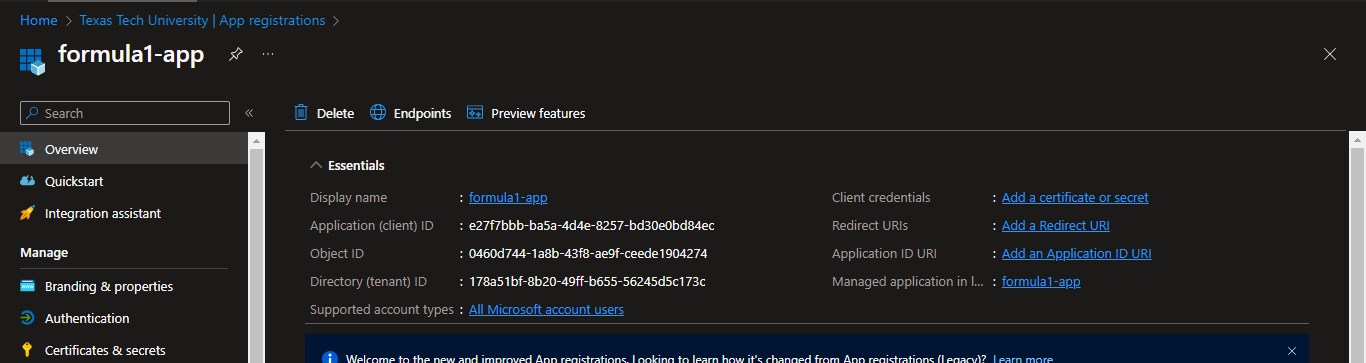








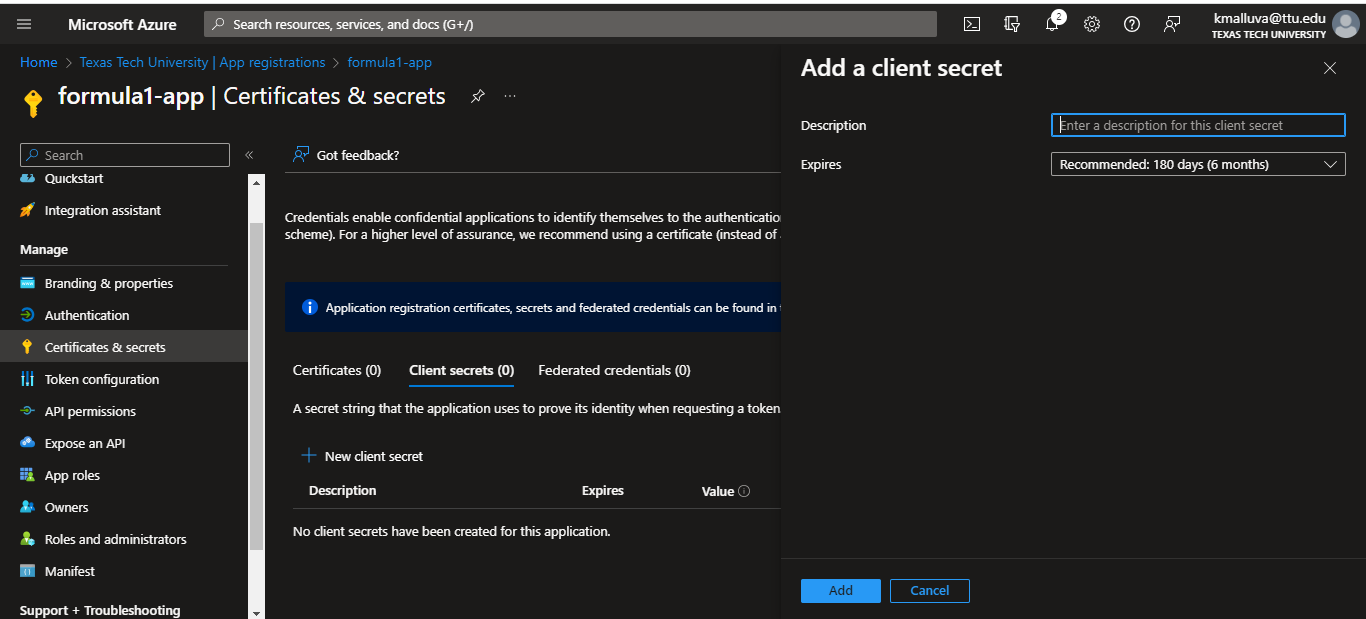
Application id and tentant id is imp:

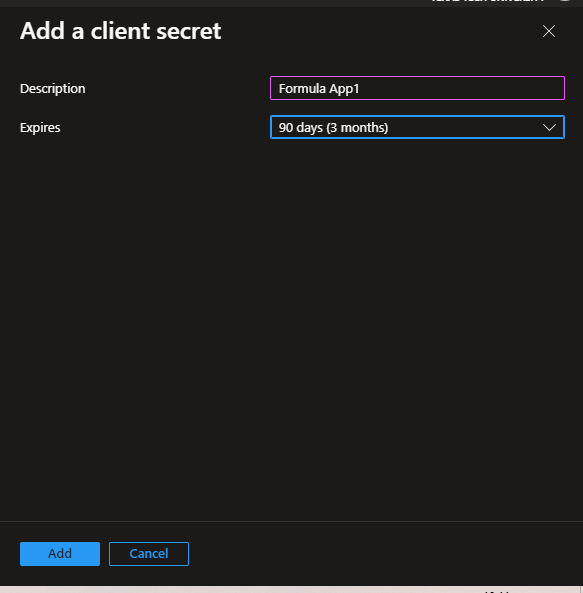


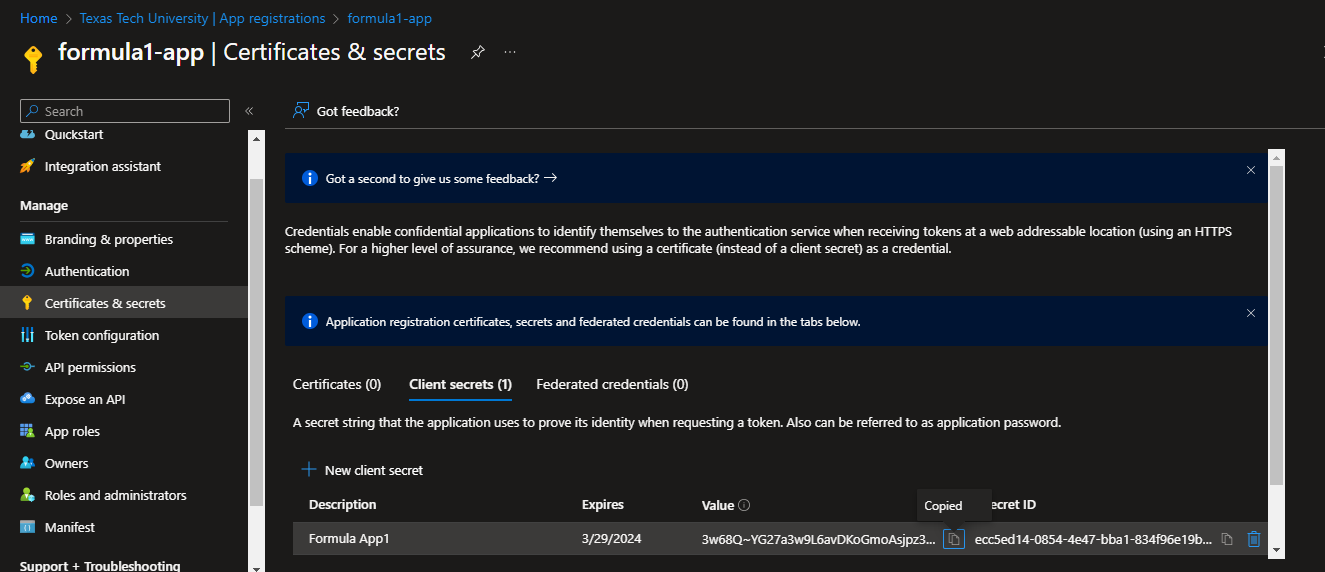
Copy application id and paste in notebook:



Certificates and secrets:

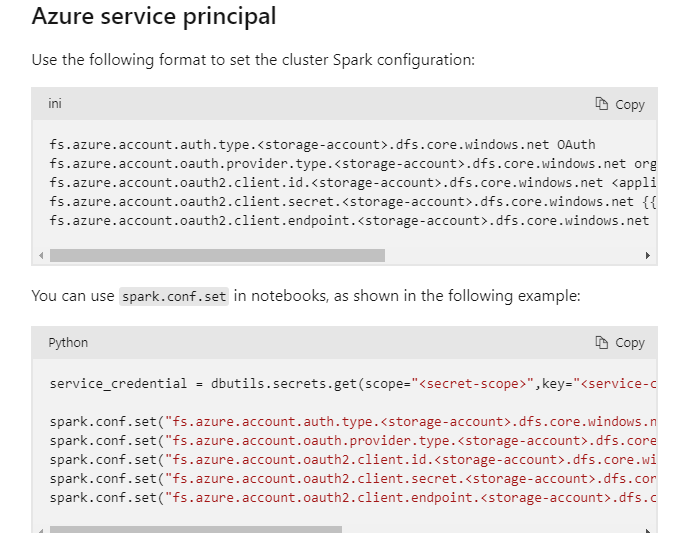






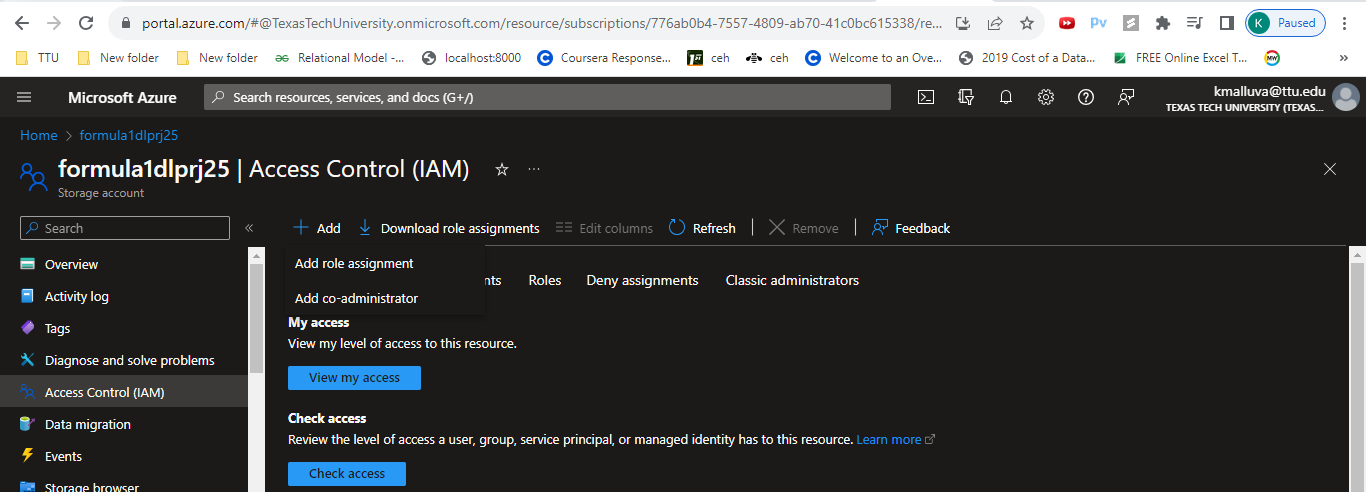
<https://learn.microsoft.com/en-us/azure/databricks/connect/storage/azure-storage#--access-azure-data-lake-storage-gen2-or-blob-storage-using-oauth-20-with-an-azure-service-principal>

Copy form this url:



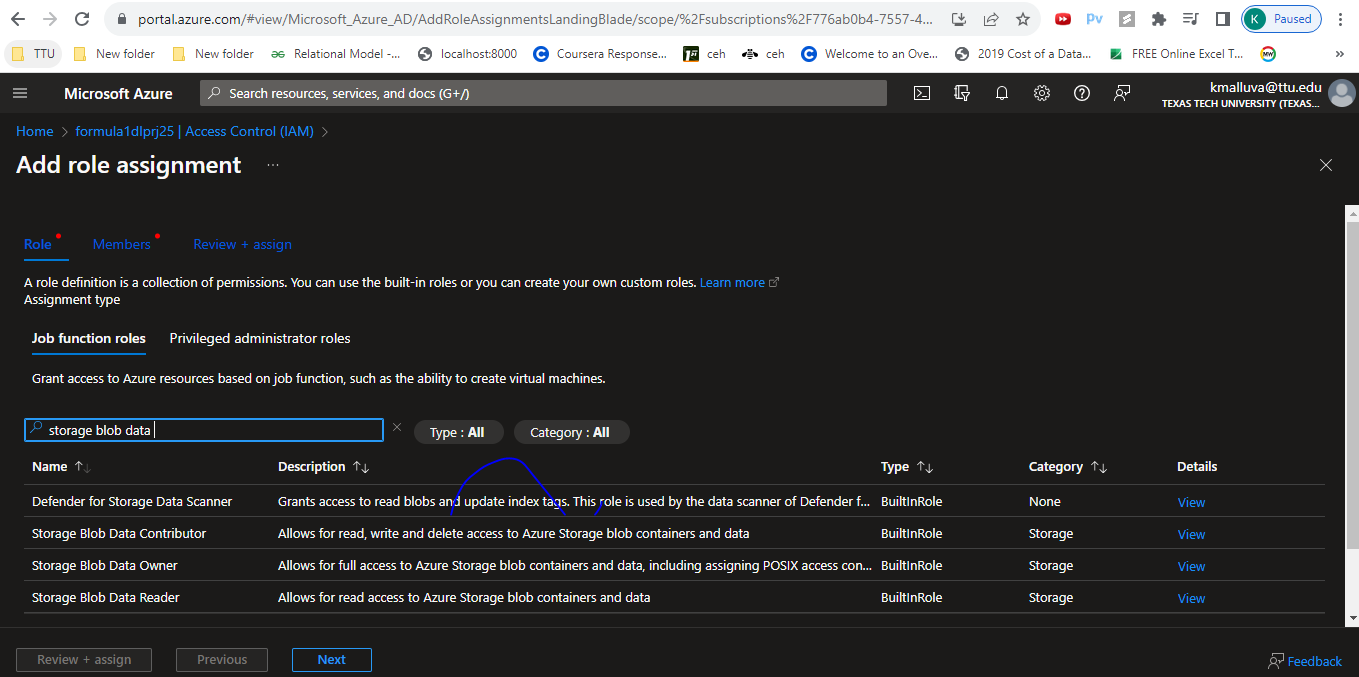


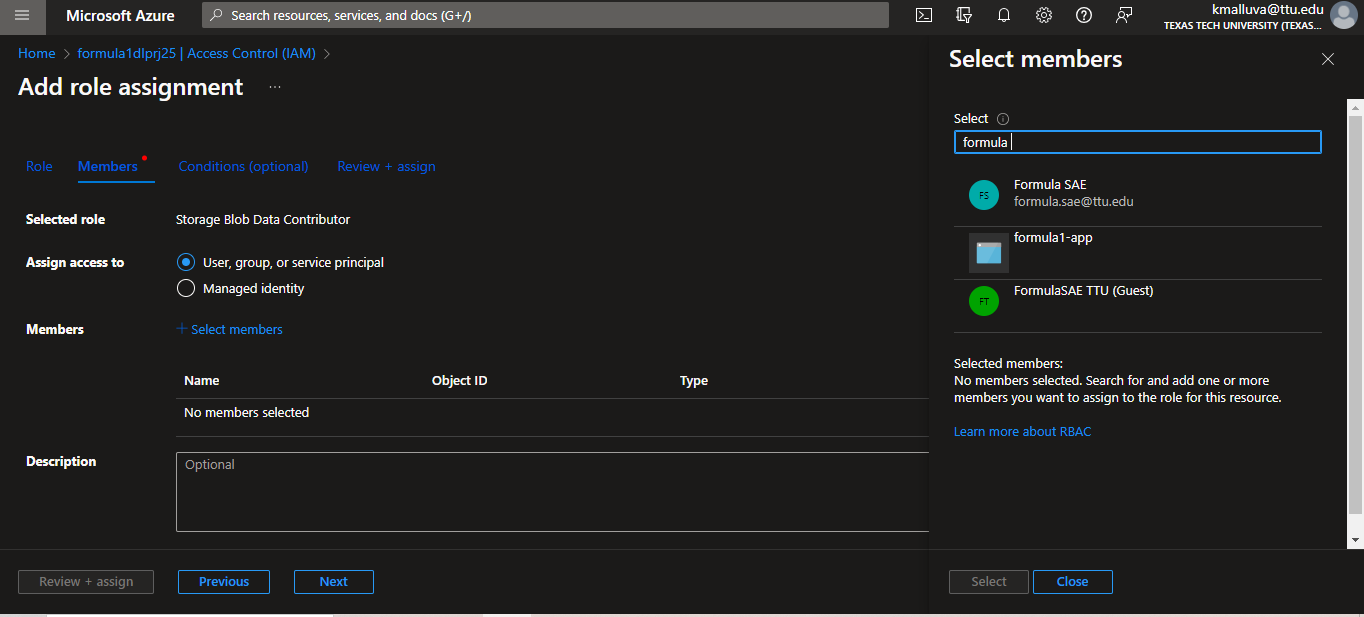
Go to azure portal -> storaage account -> Access Control -> add role assgn ->

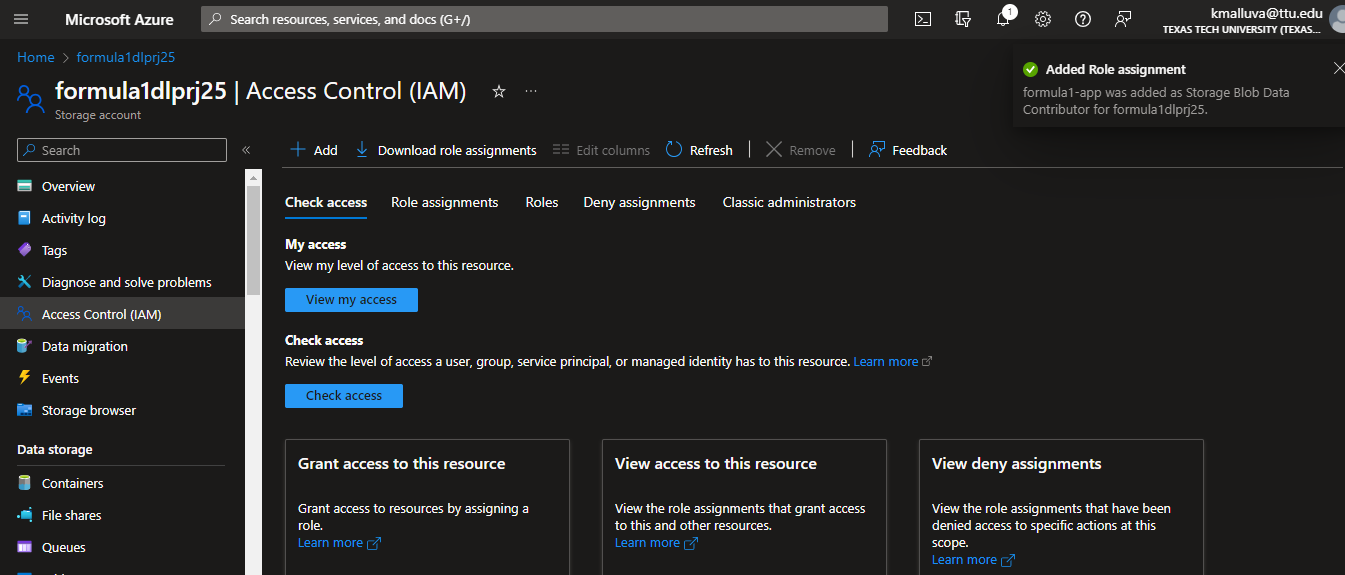


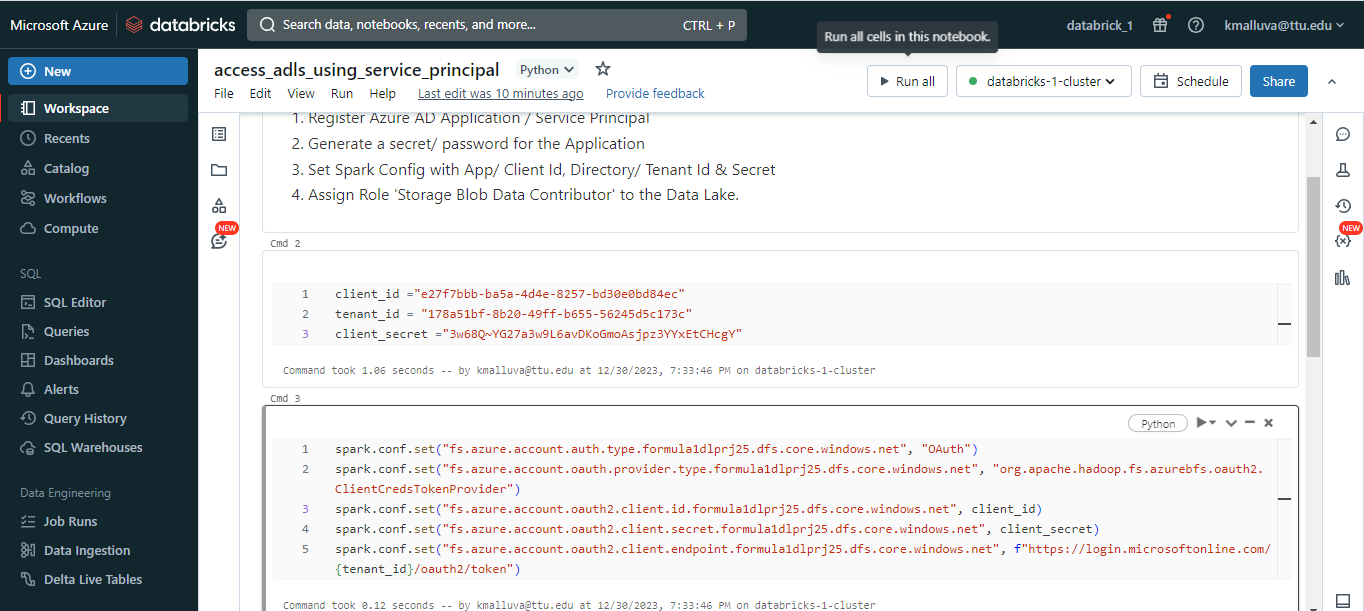
Add role assignment -> search for storage bolb data

contribution

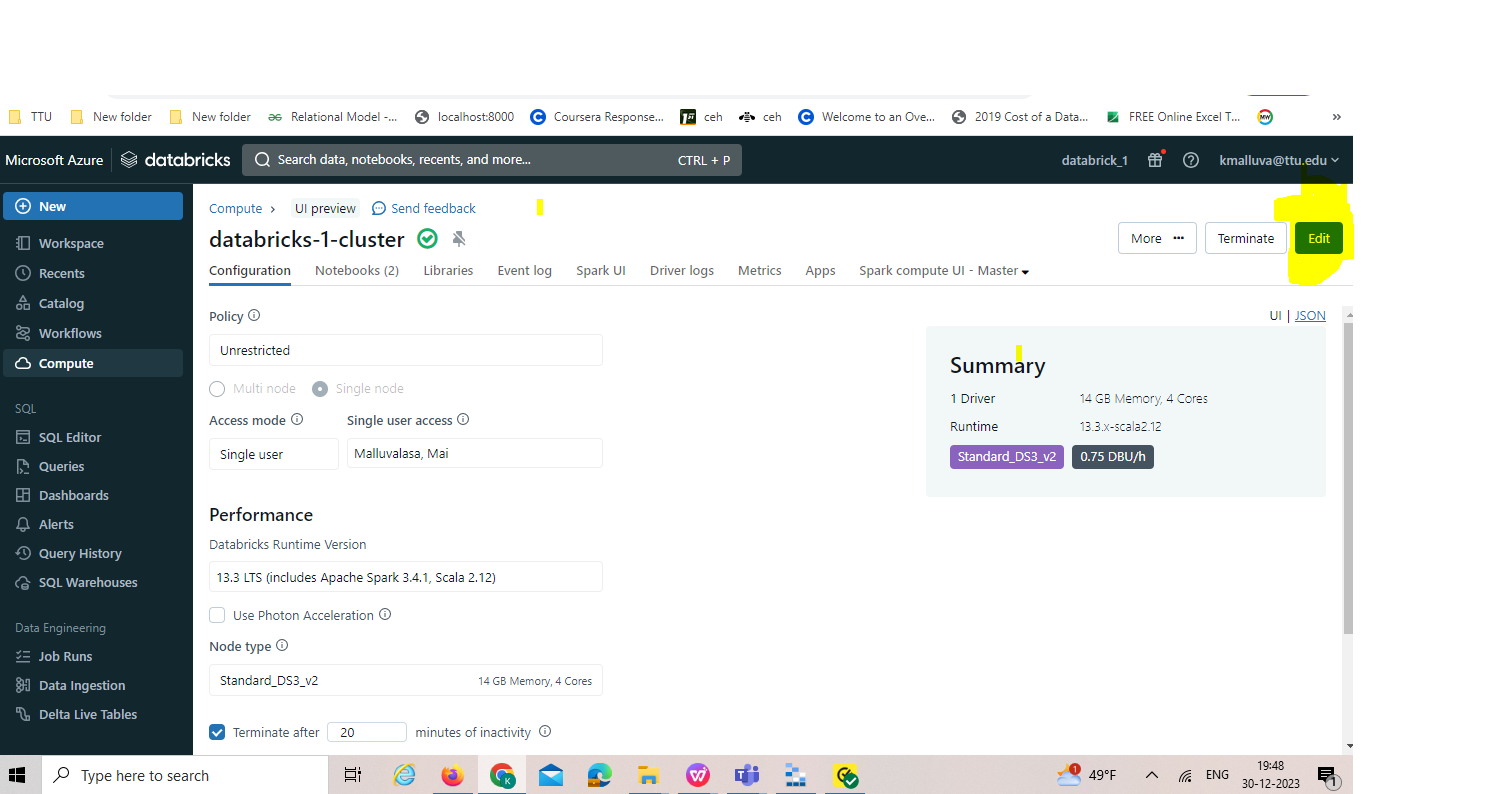


‘

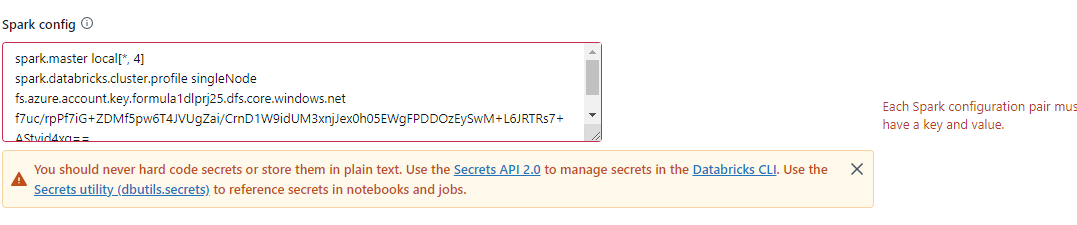


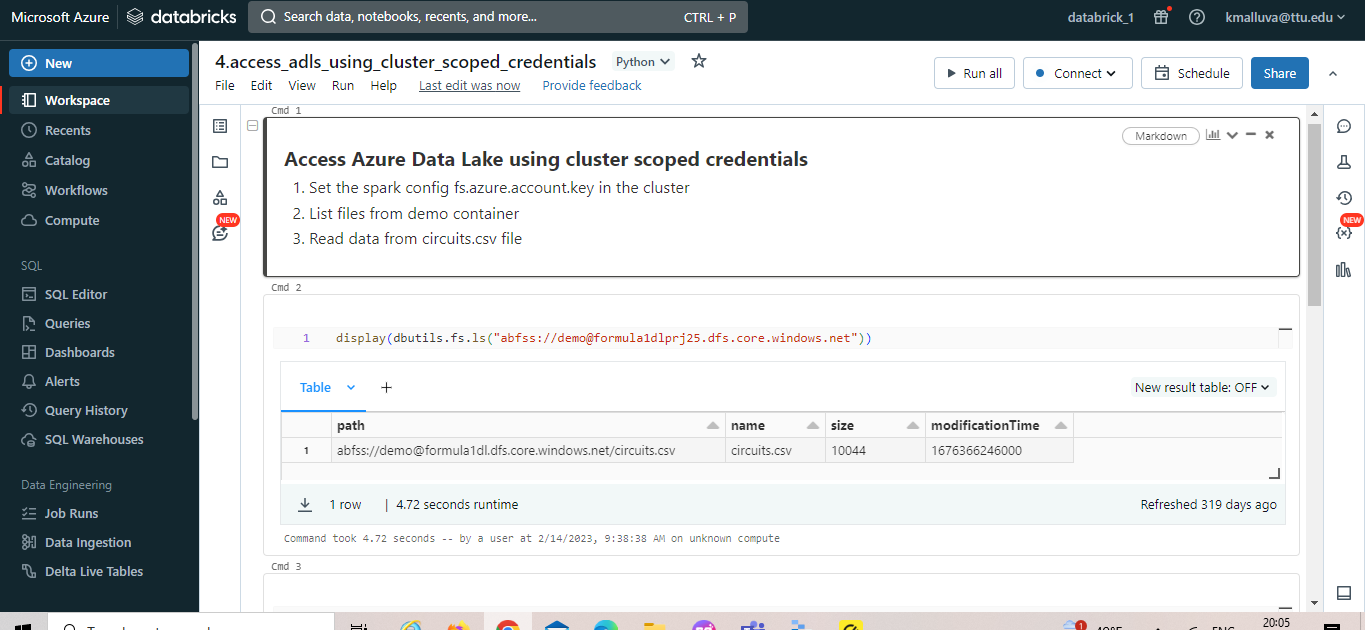


Session scoped authentication:



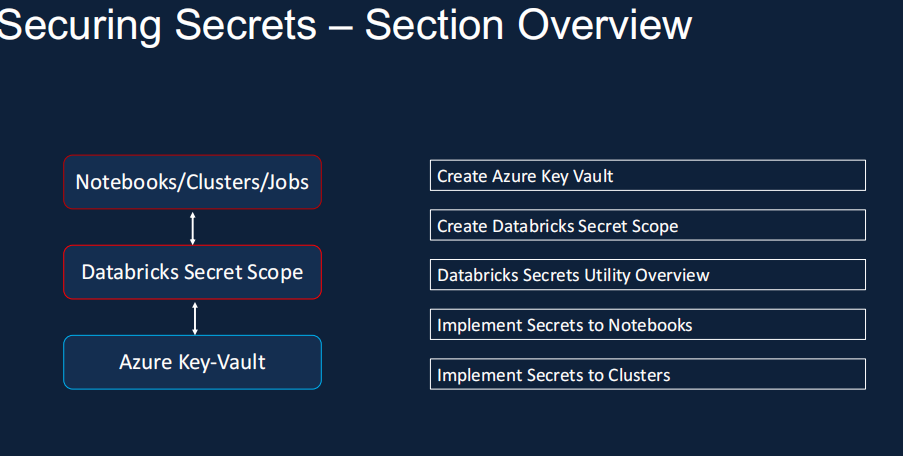
Not a good it to store the data in spark config:



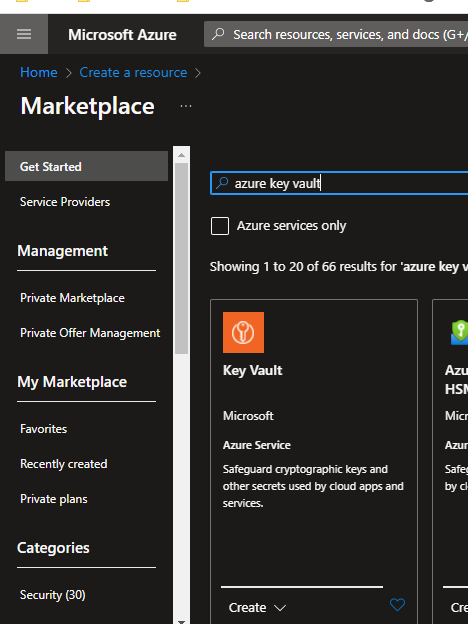


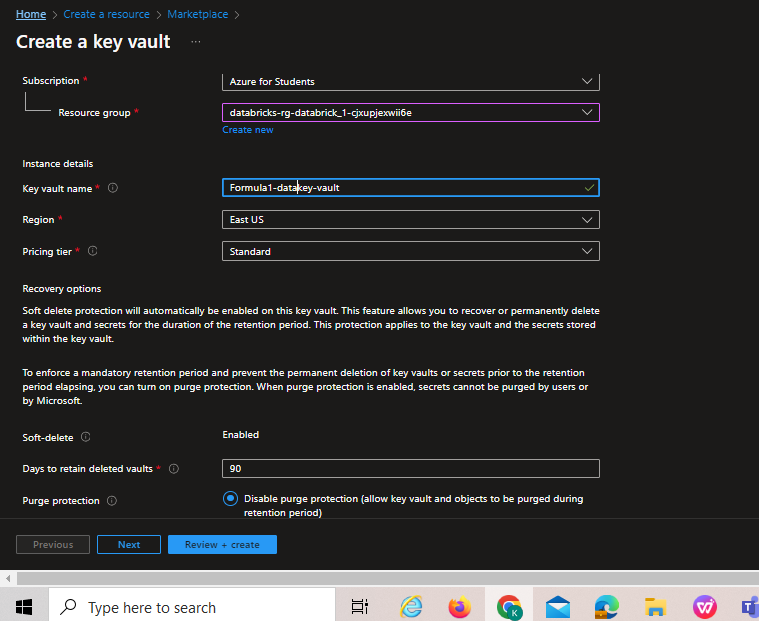
Access azure data lake using credential passthrough:

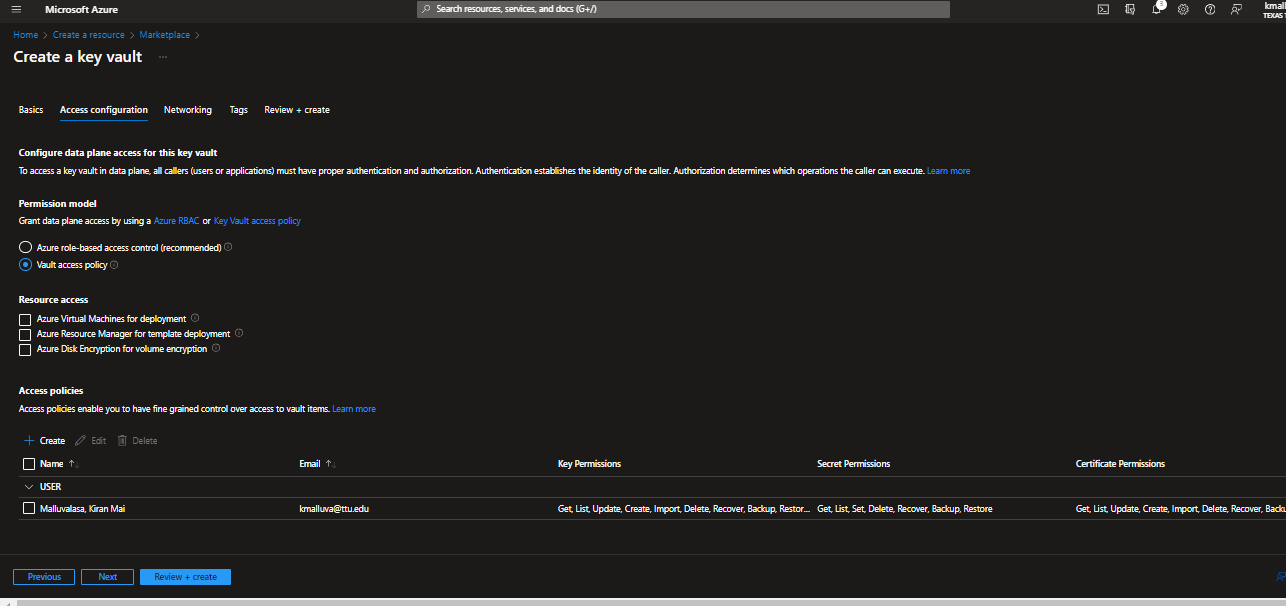


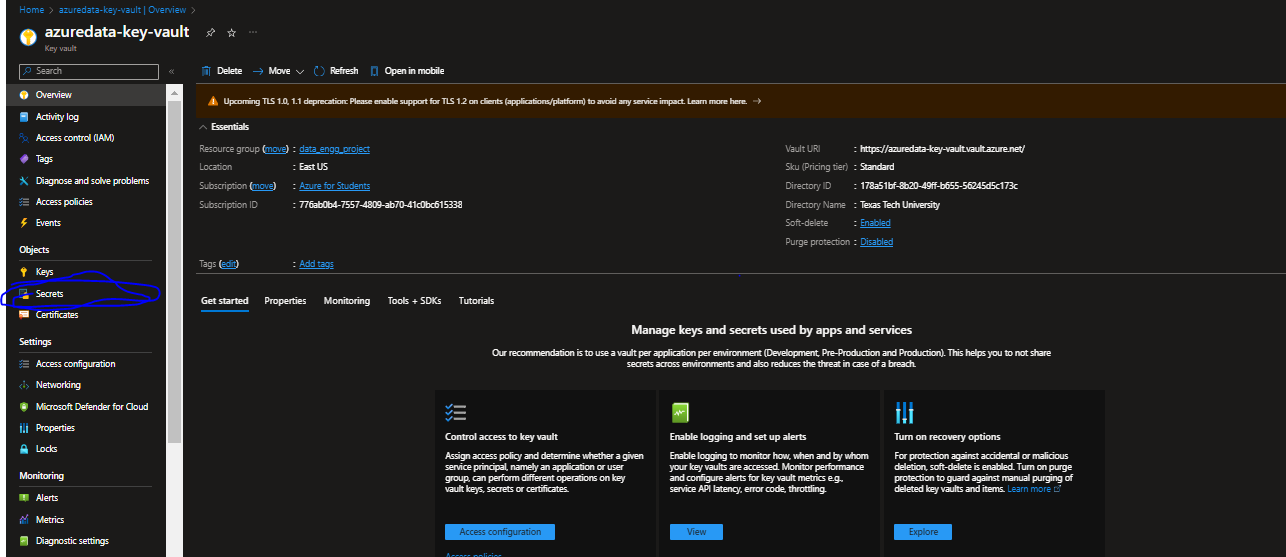


Creating Azure key vault:



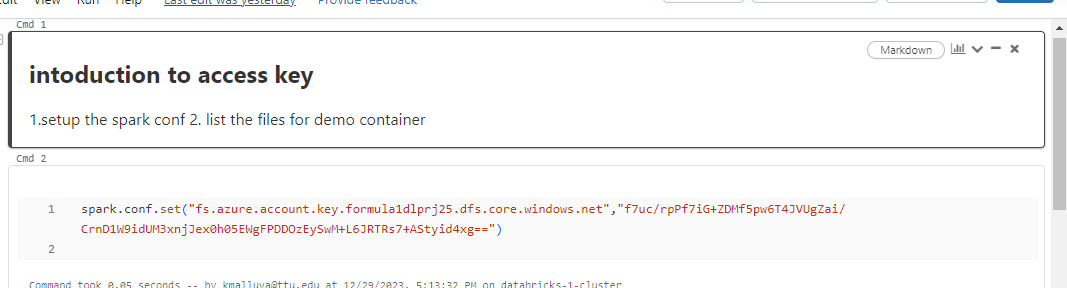


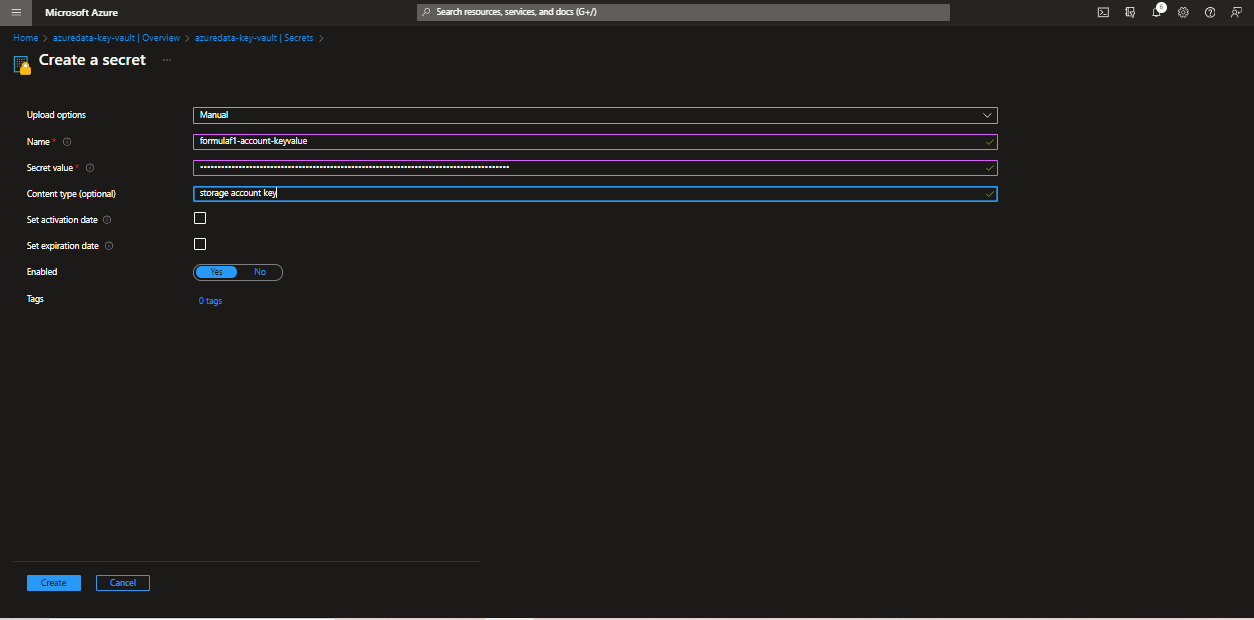






Copy this key into secret value:

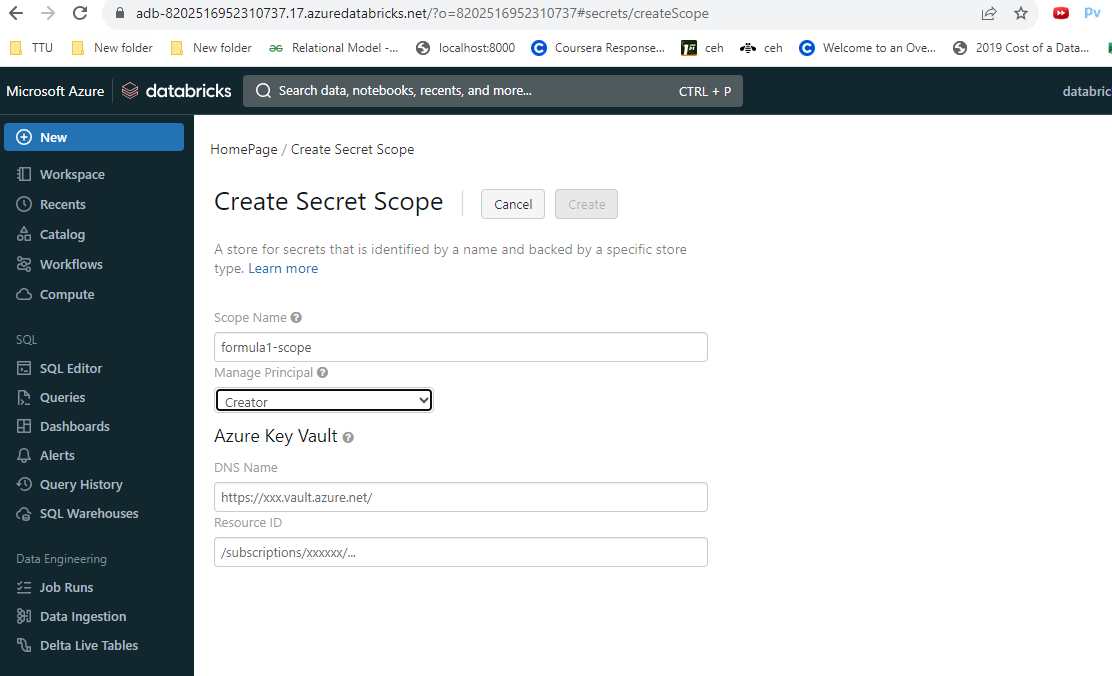




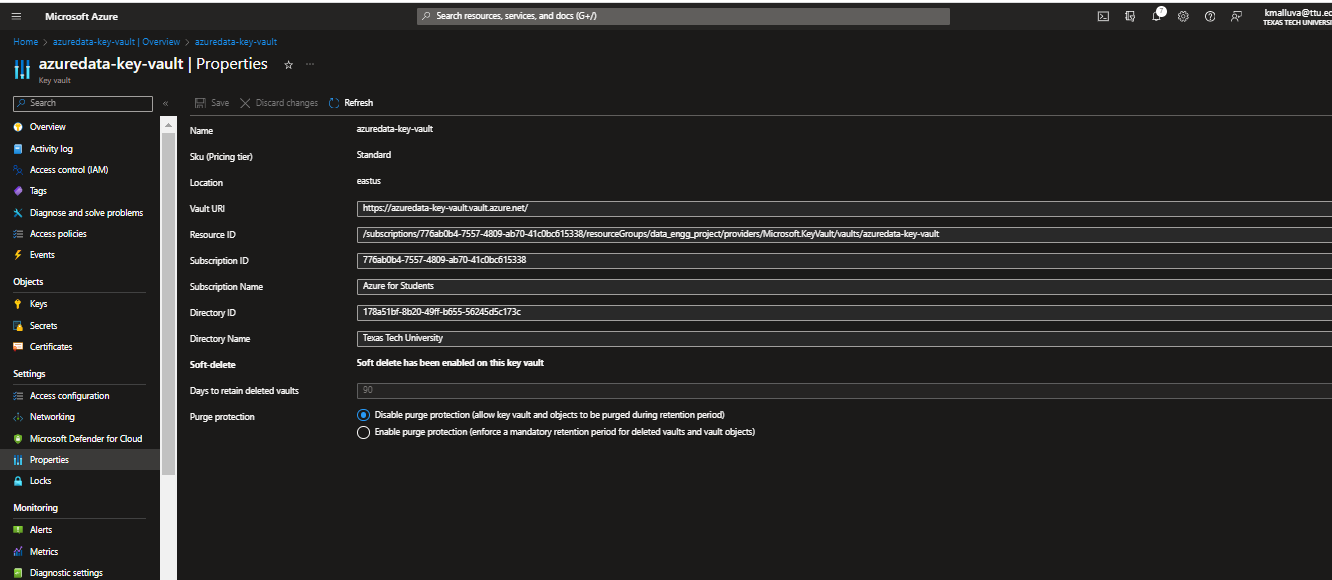
Creating secret scope:

<https://adb-8202516952310737.17.azuredatabricks.net/?o=8202516952310737#secrets/createScope>

Go to home page => and add secrets/createScope to that url



Azure data key vault -> properties -> copy the vault url and resource id:



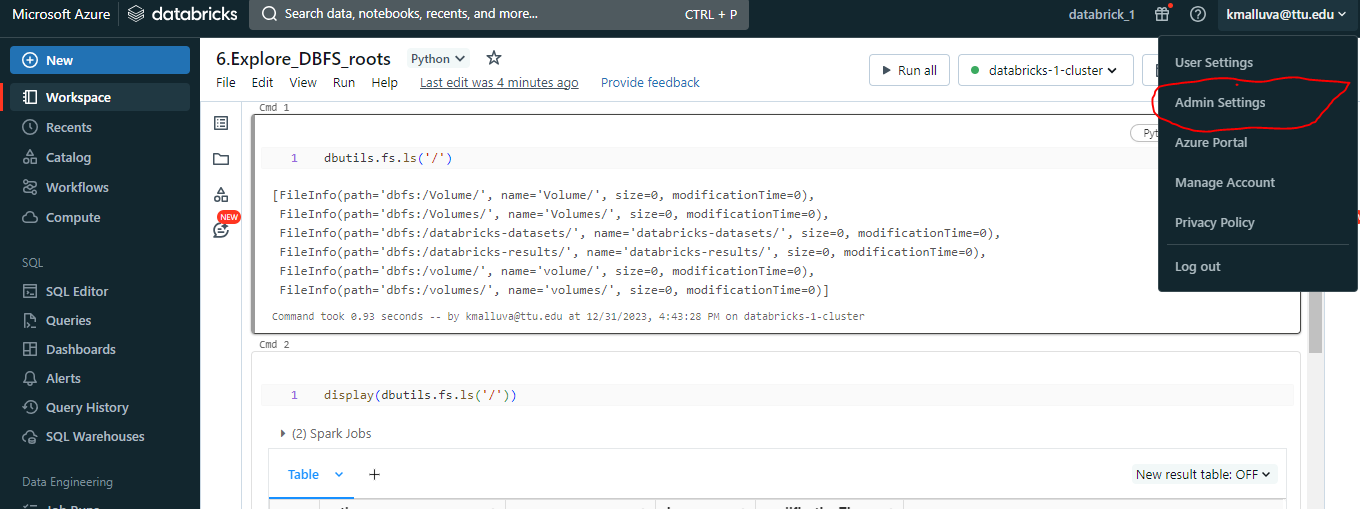


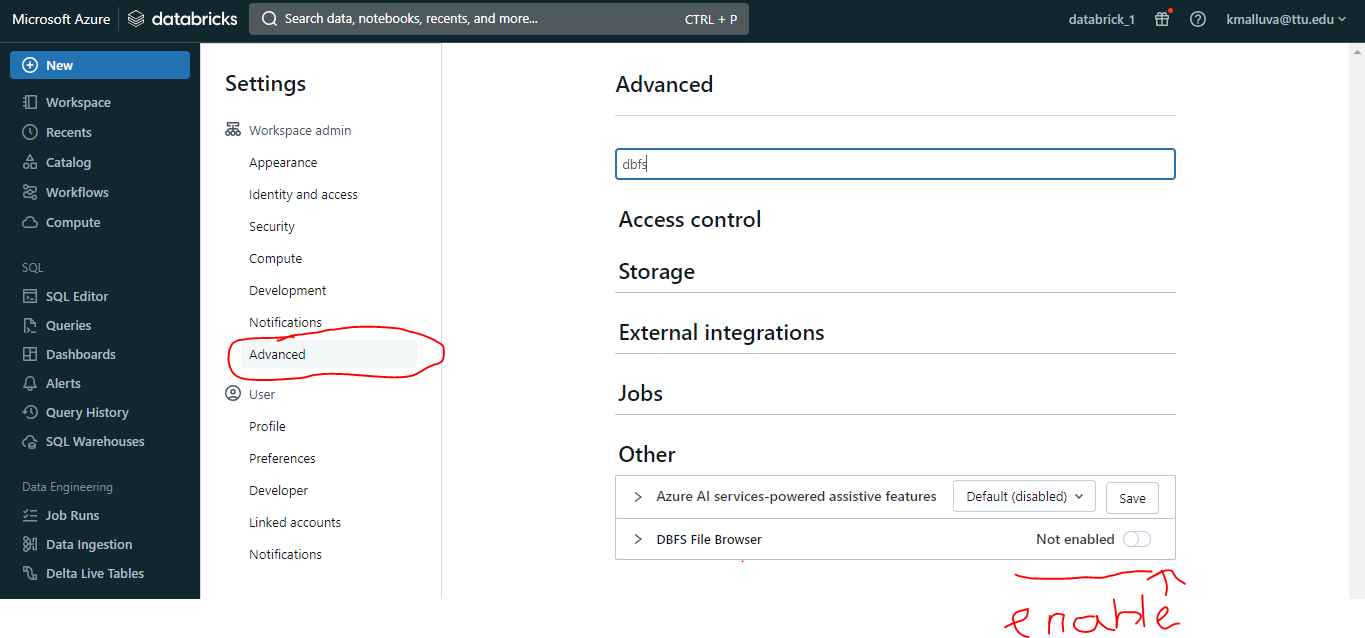
Databricks secret utilities:

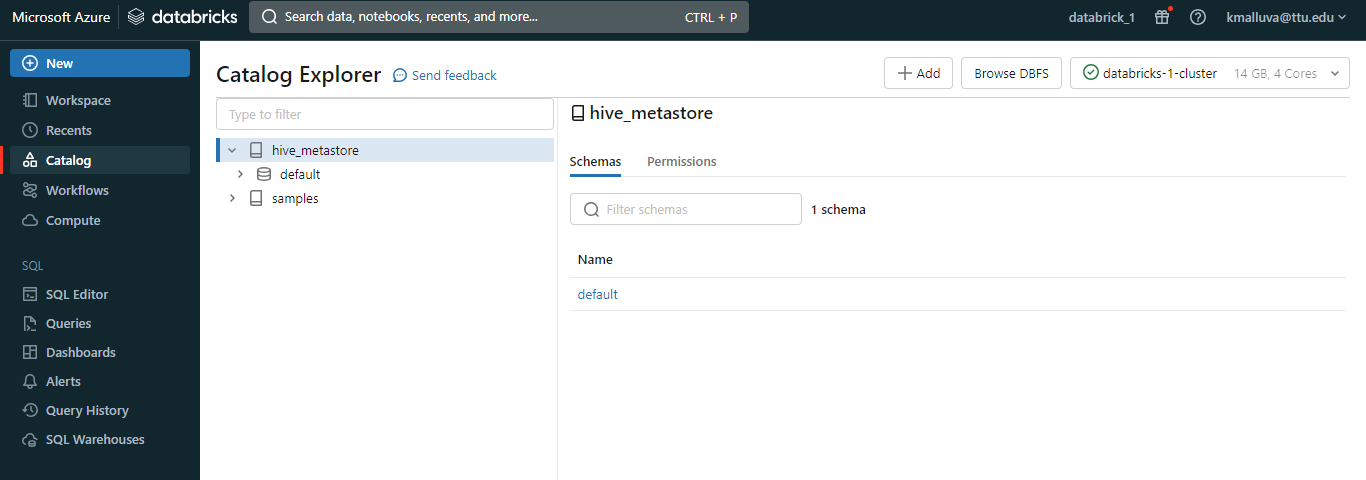


Databricks Mounts:

Databricks file system is distributed file system mounted on the databricks workspace.

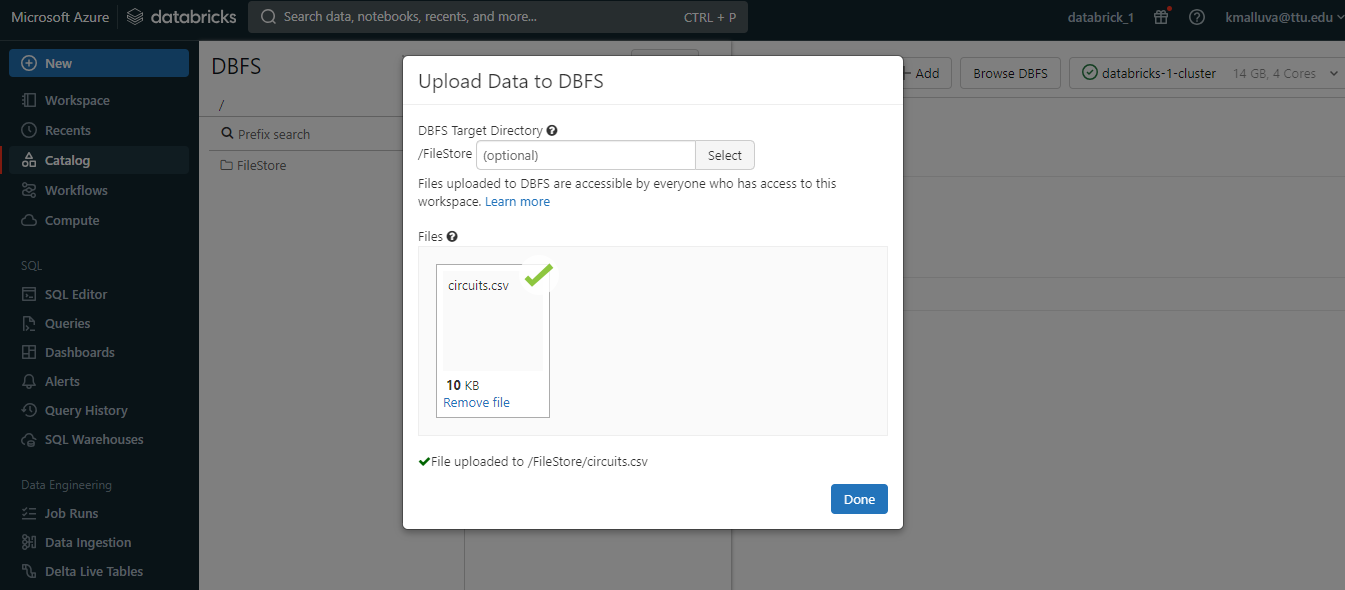




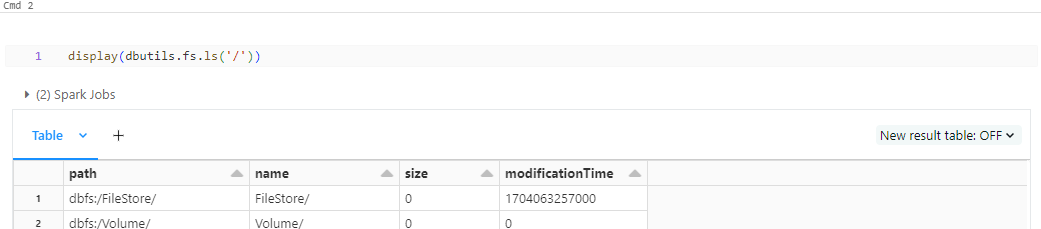


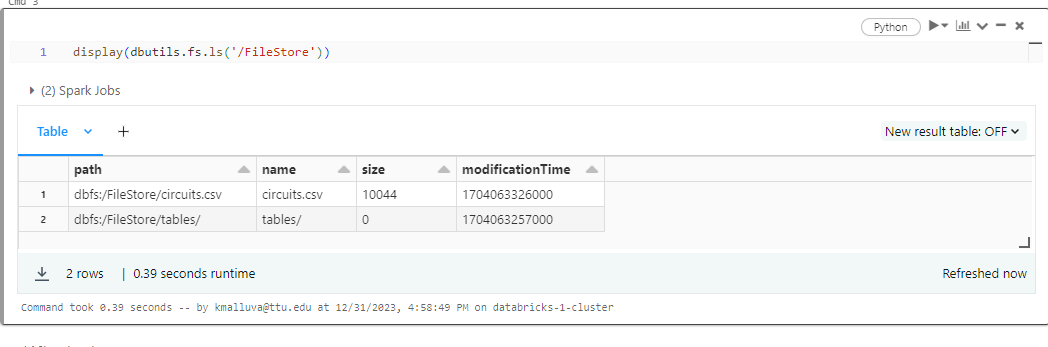
Where at the top you can see DBFS browser

Where you ca insert the files. ,click on that DBFS -> upload file ->circuits.csv

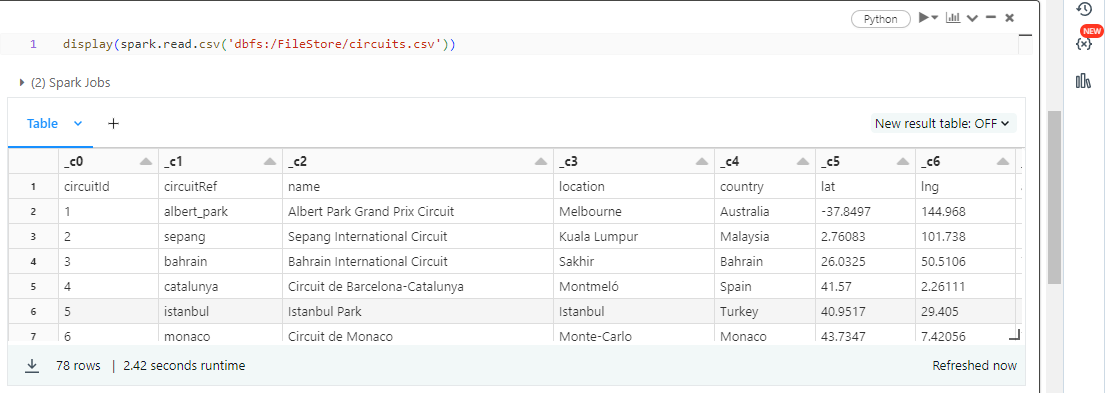


Where you can see the file is added :



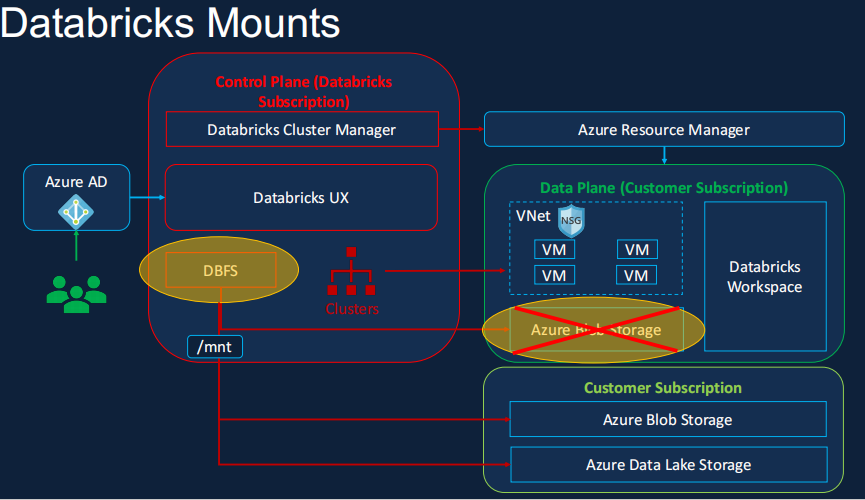


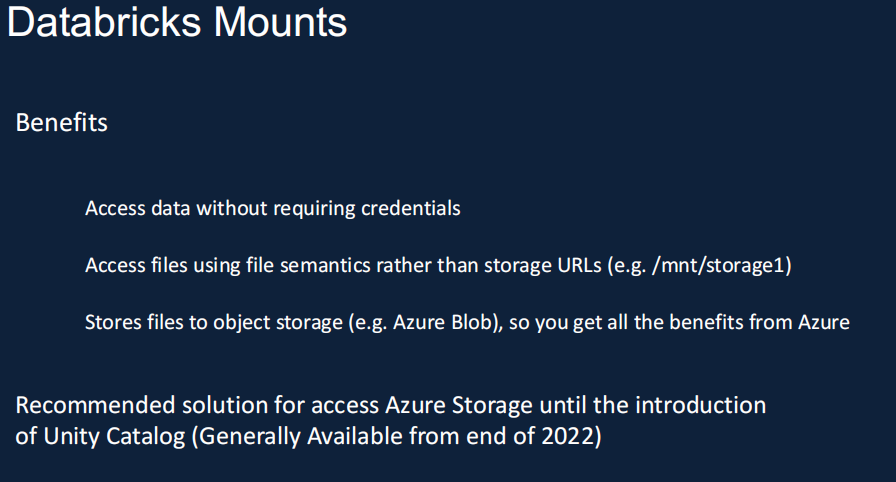
To read the csv file:

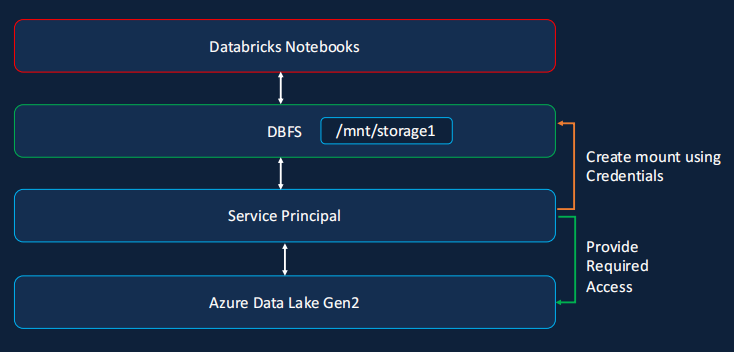


Where DBFS root is not secure for customer data, so we should not use DBFS root

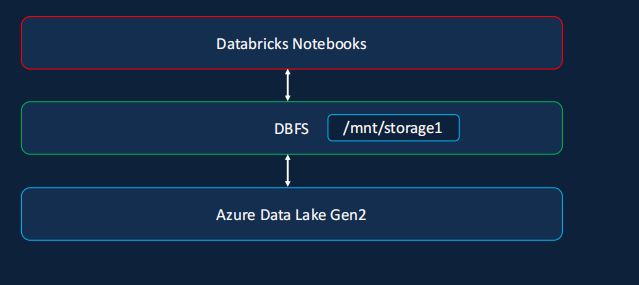
Alternative:







But databrick mount doesnot need any crediancials



Where we don’t show our credientials:

