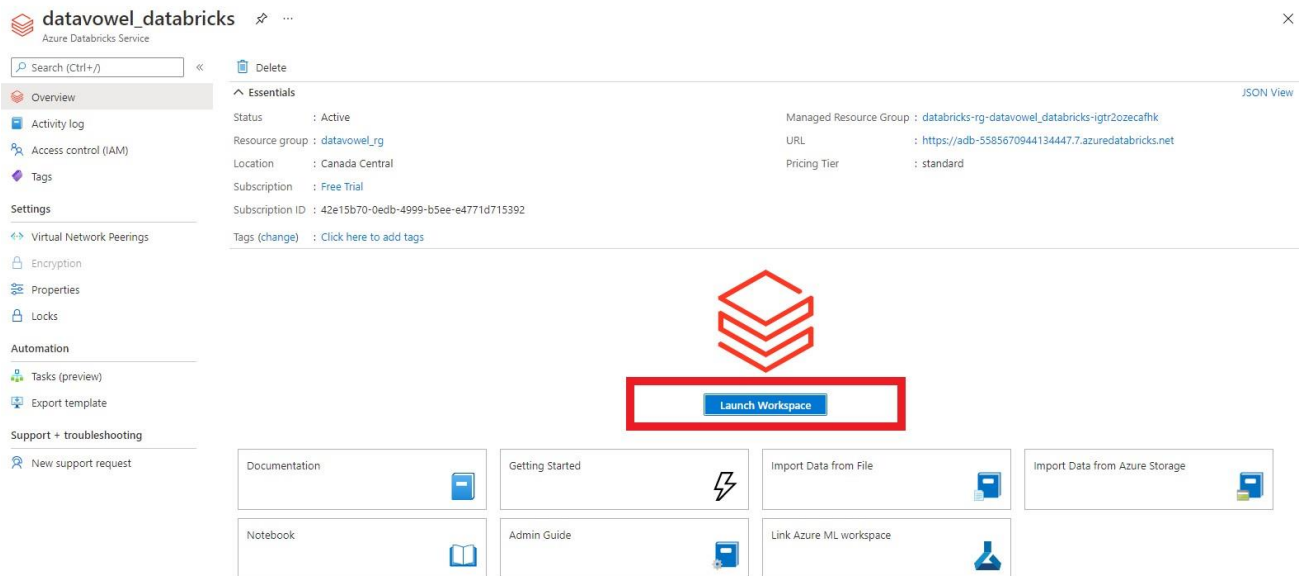


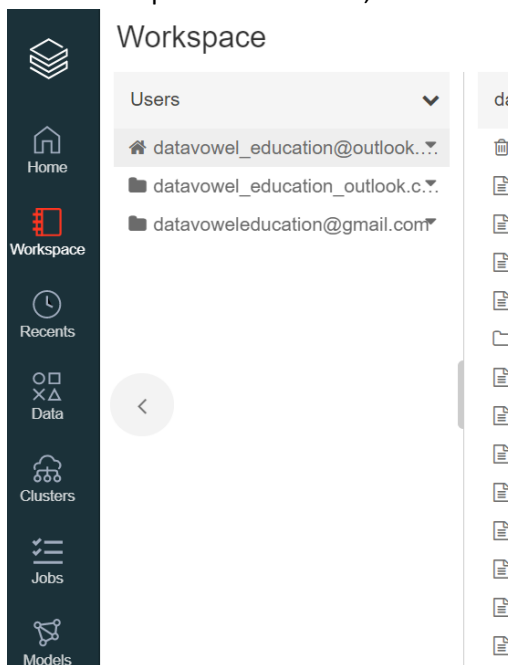
Video for it is in lecture 06 (importing dbc files) and for mounting the storage account refer lecture 13 too. Feel free to approach me in case of more questions.

How to Import DBC files

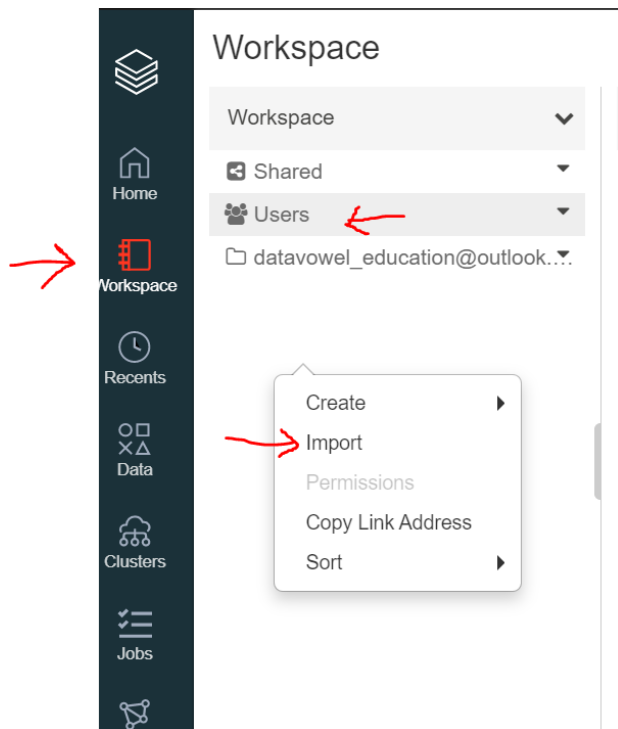
1) Launch your databricks workspace



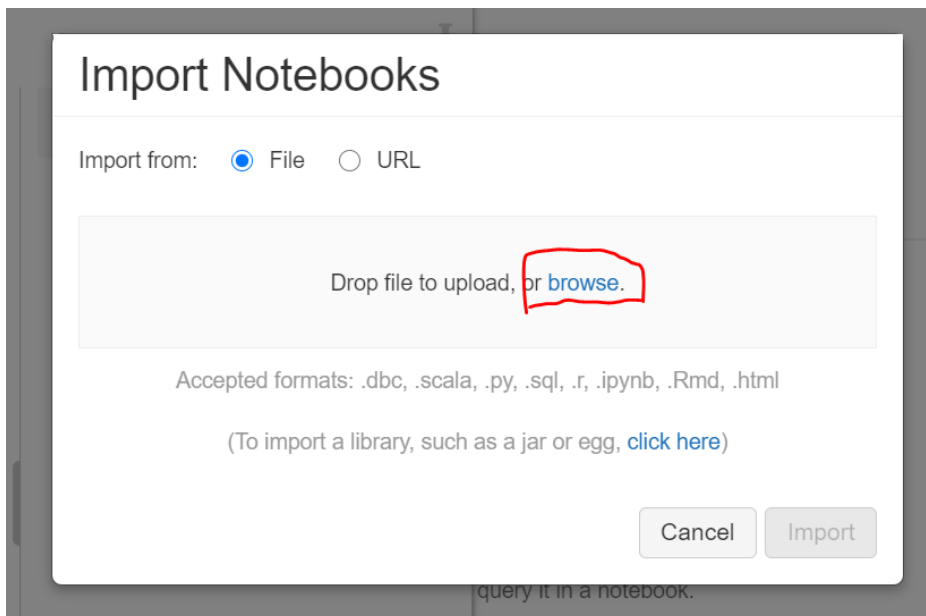
2) Go to workspace and left click, this will bring below screen



- 3) Click on the users and after that right click on blank space it will give a option to import files



- 4) After clicking import, below pop up will come and click browse to import dbc file.



- 5) Now import the DBC file which is attached in the resource section of lecture 6

Mounting of Storage account

-----Script to mount storage account-----

Video simulation for it is in lecture 13. All bold highlighted content needs to be replaced according to your subscription. Container name for our course is **files** and storage account name is **datavowelstorage**. **SAS key generation process is also covered in page 5.**

```
val containerName = "<containername>"

val storageAccountName = "<storageAccount>"

val sas = "<SAS token>"

val url = "wasbs://" + containerName + "@" + storageAccountName + ".blob.core.windows.net/"

var config = "fs.azure.sas." + containerName + "." + storageAccountName +
".blob.core.windows.net"

// COMMAND -----

// COMMAND -----

dbutils.fs.mount(
source = url,
mountPoint = "/mnt/< containername >",
extraConfigs = Map(config -> sas))
```

-----How to create containers-----

datavowelstorage Storage account

Search (Ctrl+/) Open in Explorer Move Refresh Delete Feedback

Overview

- Activity log
- Tags
- Diagnose and solve problems
- Access Control (IAM)
- Data migration
- Events
- Storage Explorer (preview)

Settings

- Access keys
- Geo-replication
- CORS
- Configuration
- Encryption
- Shared access signature
- Networking
- Security
- Static website
- Properties
- Locks

Classic alerts in Azure Monitor is announced to retire in 2021, it is recommended that you upgrade your classic alert rules to retain alerting functionality with the new ARM storage accounts. [See Continue alerting with ARM storage accounts.](#)

Essentials

Resource group (change) : [datavowel_rg](#)

Status : Primary: Available, Secondary: Available

Location : Canada Central, Canada East

Subscription (change) : [Free Trial](#)

Subscription ID : 42e15b70-0edb-4999-b5ee-e4771d715392

Tags (change) : [Click here to add tags](#)

Performance/Access tier : Standard/High

Replication : Read-accessible

Account kind : StorageV2

Containers

Scalable, multi-effective storage for unstructured data

Learn more

File shares

Serverless SMB and NFS file shares

Learn more

Tables

Tabular data storage

Learn more

Tools and SDKs

Storage Explorer (preview) PowerShell Azure CLI .NET Java Python Node.js

Monitoring

Show data for last: 1 hour 6 hours 12 hours 1 day 7 days

Show data for: Account

Click container and give the name of container and in our course name of container is **files**

datavowelstorage Containers

Search (Ctrl+/) + Container Change access level Restore containers Refresh Delete

Search containers by prefix: Show details

Name	Last modified	Public access level
You don't have any containers yet. Click '+ Container' to get started.		

New container

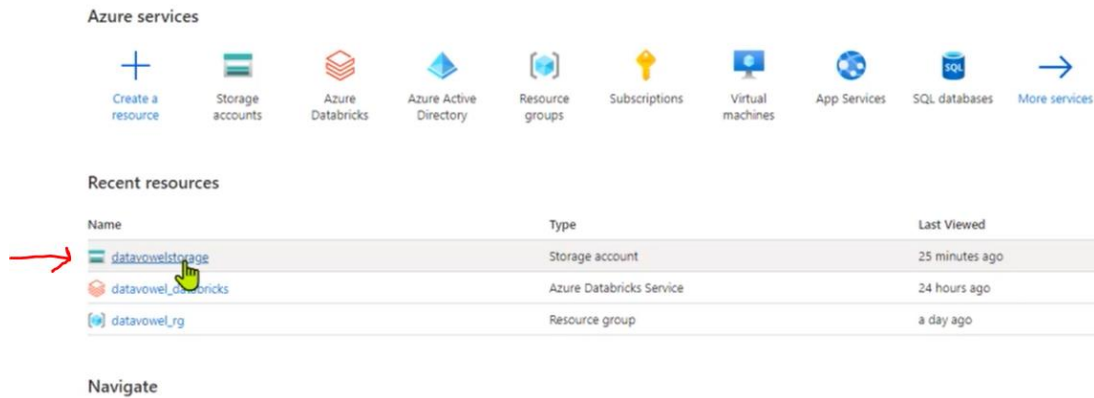
Name *

Public access level: Private (no anonymous access)

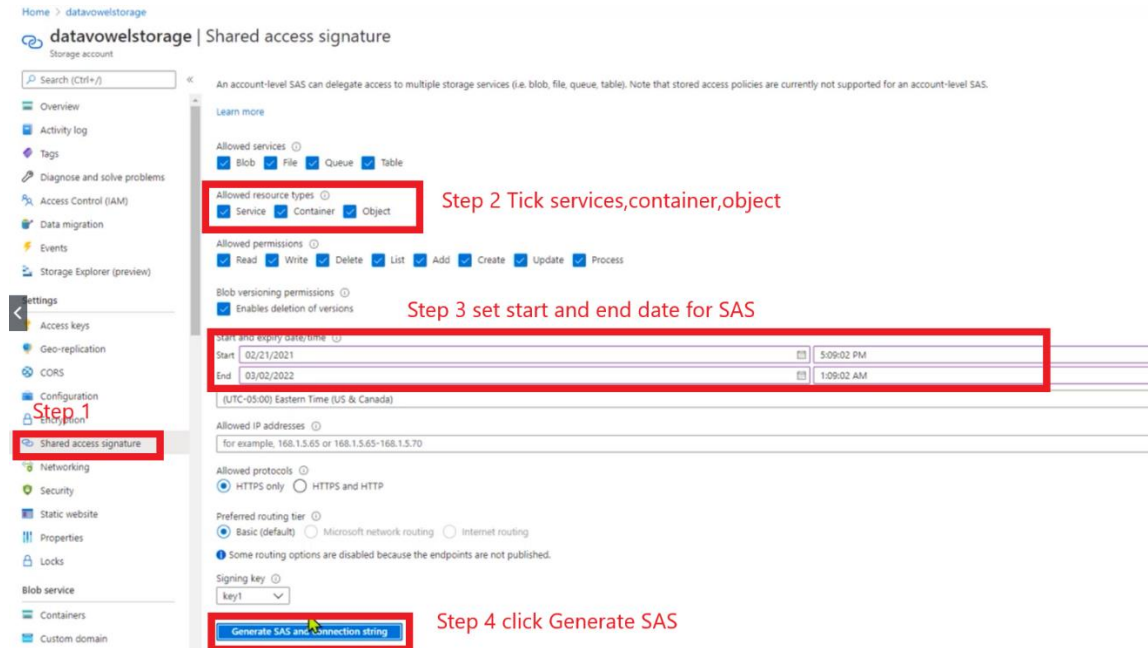
Advanced

In order to generate SAS keys

Go to storage account



You will land up screen like below and after that on left hand side pane click Shared Access Signature and follow steps from 1 to 4



Below Highlighted box is the SAS token

Home > datavowelstorage

datavowelstorage | Shared access signature

Storage account

Search (Ctrl+F)

End 03/02/2022 1:09:02 AM

(UTC-05:00) Eastern Time (US & Canada)

Allowed IP addresses ⓘ
for example, 168.1.5.65 or 168.1.5.65-168.1.5.70

Allowed protocols ⓘ
☒ HTTPS only ☐ HTTPS and HTTP

Preferred routing tier ⓘ
☒ Basic (default) ☐ Microsoft network routing ☐ Internet routing
Some routing options are disabled because the endpoints are not published.

Signing key ⓘ
key1

Generate SAS and connection string

Connection string
https://datavowelstorage.blob.core.windows.net/QueueEndpoint=https://datavowelstorage.queue.core.windows.net/FileEndpoint=https://datavowelstorage.file.core.windows.net/TableEndpoint=https://datavowelstorage.table.core.windows.net

SAS token ⓘ
?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D

Blob service SAS URL
https://datavowelstorage.blob.core.windows.net/?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D

File service SAS URL
https://datavowelstorage.file.core.windows.net/?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D

Queue service SAS URL
https://datavowelstorage.queue.core.windows.net/?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D

Table service SAS URL
https://datavowelstorage.table.core.windows.net/?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D

At last for mounting replace values according to your subscription. For our course values will be like below. Note if you have given any other containername and storage account name replace values accordingly.

```
val containerName = "files"
```

```
val storageAccountName = "datavowelstorage"
```

```
val sas = "?sv=2020-02-10&ss=bfqt&srt=sco&sp=rwdlacupx&se=2022-03-02T06:09:02Z&st=2021-02-21T22:09:02Z&spr=https&sig=wBDVJfKpXSdsnglDaNiVCs%2Bs1epTD7Tvlgvpkv8288c%3D"
```

```
val url = "wasbs://" + containerName + "@" + storageAccountName + ".blob.core.windows.net/"
```

```
var config = "fs.azure.sas." + containerName + "." + storageAccountName +  
".blob.core.windows.net"
```

```
dbutils.fs.mount(  
  source = url,  
  mountPoint = "/mnt/files",  
  extraConfigs = Map(config -> sas))
```

