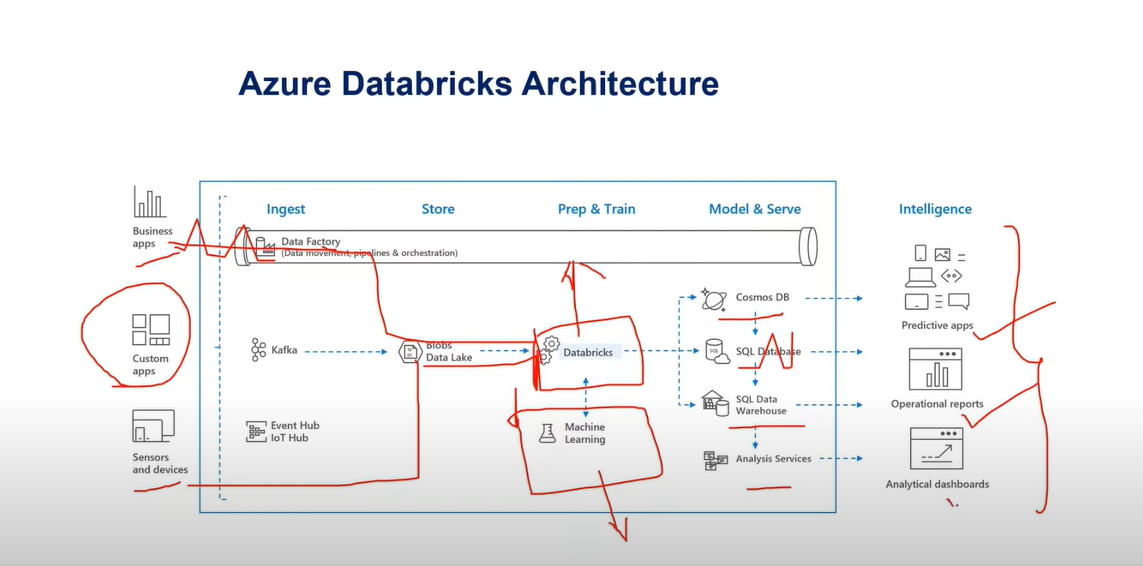
**Azure DataBricks**

**Azure Databricks is a cloud-based analytics and data engineering platform that combines Apache Spark with Microsoft Azure services to enable scalable and collaborative big data processing and machine learning workflows.**

**Azure Databricks is used to analyze and work with large amounts of data. It helps people who work with data to explore and transform it, process it quickly, and use it to build machine learning models. It's like a tool that makes it easier to handle and understand big data.**



In Azure Databricks:

1. Clusters

Notebooks

Tables

Jobs

Delta Lakes

**Dbutils Library**

Dbutils.help()

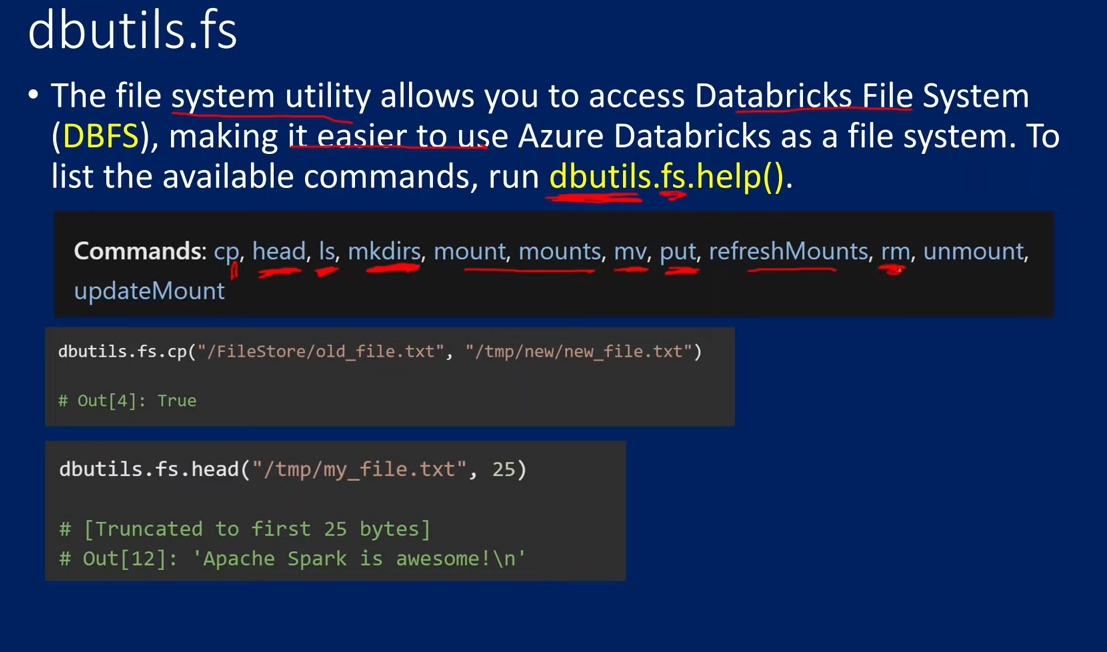
Dbutils.fs.help()

Dbutils.fs.help(‘ls’)

Dbtutils.data.help(‘summarize’)

dbutils.fs.cp('dbfs:/FileStore/tables/movie.csv','dbfs:/FileStore/tables/Copymovie.csv')

dbutils.fs.head('dbfs:/FileStore/tables/movie.csv')



Exit()

Exits a notebook with a value.

Below cells after the exit() will get skipped.

 dbutils.notebook.help()

firstname = 'lucky'

dbutils.notebook.exit(firstname)

Run()

run a notebook and returns its exit value.

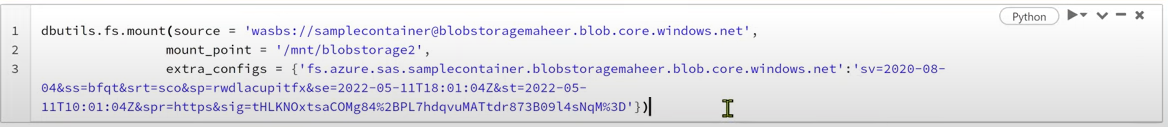
Dbutils.notebook.run(‘path’, timeout)

**Mount()**

The **mount** operation is used in Databricks to connect external data sources, such as Azure Blob Storage, to the Databricks file system (DBFS). It allows you to access and work with data stored in external storage systems as if it were part of the DBFS.

When you mount a directory, it becomes accessible as a new folder within the DBFS. This enables you to read, write, and manipulate the data stored in the external storage system directly from Databricks.

**Connect to blob storage using sas token**



**Cluster**

An Azure Databricks cluster is a set of computation resources and configurations on which you run data engineering, data science, and data analytics workloads, such as production ETL pipelines, streaming analytics, ad-hoc analytics, and machine learning.

It have spark engine and other component installed.

**Types of Clusters**

1. **All-purpose clusters**

All-purpose clusters can be shared by multiple users and are best for performing ad-hoc analysis, data exploration, or development.

1. **Job clusters**

Job clusters terminate when your job ends, reducing resource usage and cost.

Azure Databricks retains cluster configuration information for up to 200 all-purpose clusters terminated in the last 30 days and up to 30 job clusters recently terminated by the job scheduler. To keep an all-purpose cluster configuration even after it has been **[terminated](https://learn.microsoft.com/en-us/azure/databricks/clusters/clusters-manage" \l "cluster-terminate)** for more than 30 days, an administrator can **[pin](https://learn.microsoft.com/en-us/azure/databricks/clusters/clusters-manage" \l "cluster-pin)** a cluster to the cluster list.

https://community.cloud.databricks.com/?o=1540931408951712#secrets/createScope

**Manage Keys and secrets in Azure Databricks**

There can be two ways:

1. Azure key vault (keys are stored in azure key vault and accessed by azure databricks)
2. DataBricks backed scopes (keys are stored in dbfs in encrypted database in azure databricks)

**Azure Keys Vault:**

Step-1 store your secret in key vault

Step-2 open Azure Databricks

Step-3 <https://community.cloud.databricks.com/?o=1540931408951712#secrets/createScope>

Step-4 Fill Credentials and add

Step-5 use the key vault key in dbutils.fs.mount command

dbutils.fs.mount(

    source='wasbs://samplecontainer@luckystorage0001.blob.core.windows.net/',

    mount\_point='/mnt/new-mount',

    extra\_configs={'fs.azure.account.key.luckystorage0001.blob.core.windows.net':dbutils.secret .get(‘testscope’,’blobAccountKey’)

                   })

**Databricks backed storage**

**Configure Azure Databricks CLI**

1. **Install python**
2. **pip install databricks-cli**
3. **databricks --help (to show all commands)**
4. **databricks configure --token**
5. **Go to Azure Databricks -> Setting -> User Setting --> Generate Token**