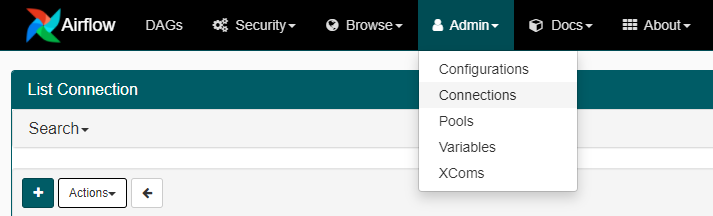
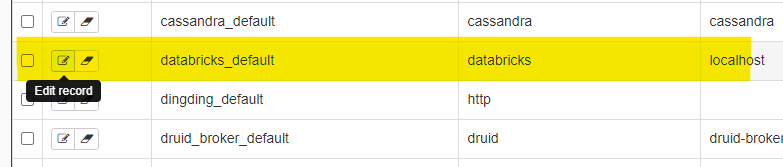
**Run notebooks/scripts/jars in Databricks using Airflow**

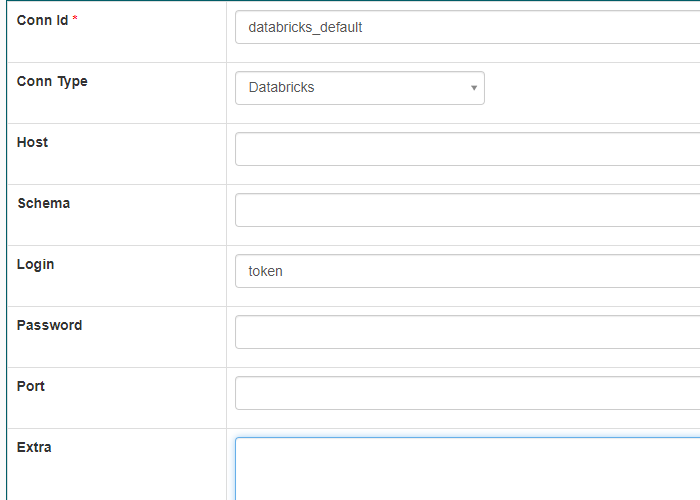
1. Create a Databricks connection in airflow

Go to Admin - > Connections 

Find **databricks\_default** and click edit (A new connection can also be created)



Fill in the required details



Host : **Error! Hyperlink reference not valid.**

Login : token

Password : Generated user token

Extra : {"token": "<user\_token>","host":"<your\_host>"}

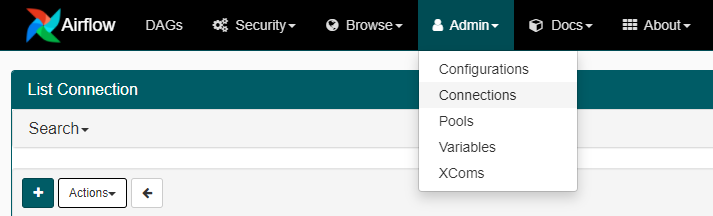
Eg: {"token": "abcdefghijklmn12345","host":"https://eastus2.azuredatabricks.net/"}

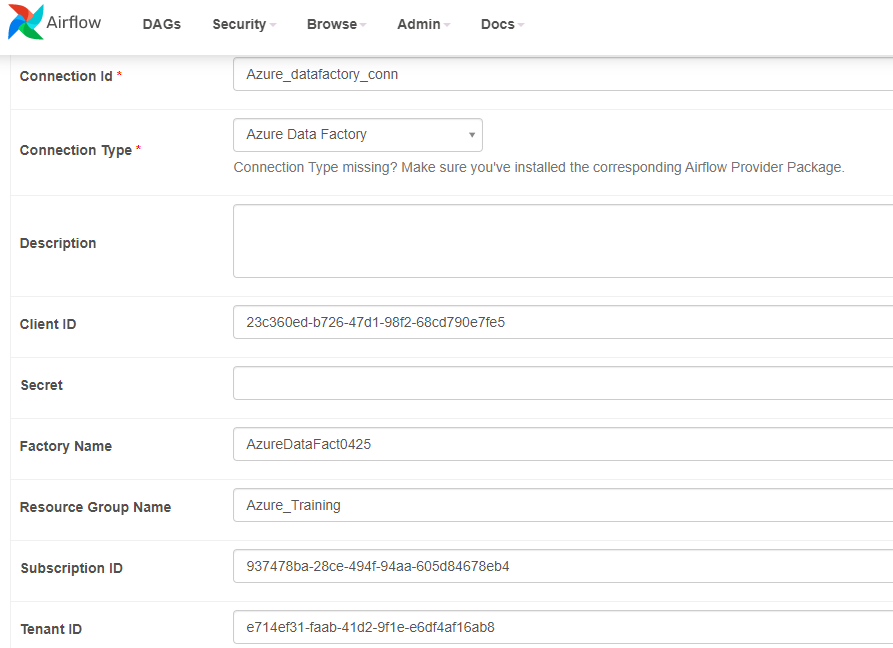
Leave the other fields empty and click save

**Run Pipeline in Azure Data Factory using airflow**

2.Create a Data Factory connection in airflow

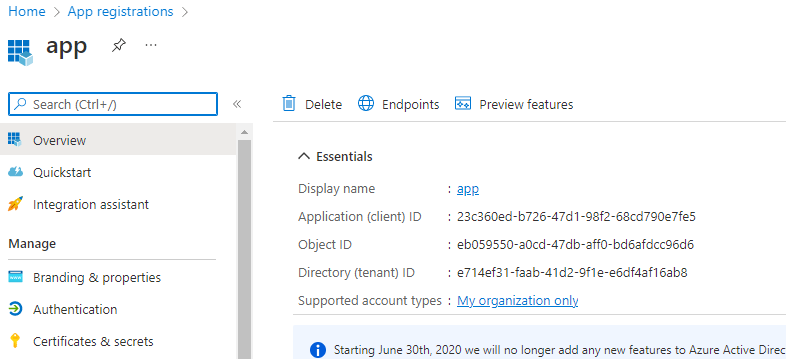
Go to Admin - > Connections



****

**Note:**

**For connecting to ADF we have to register an app on Azure to get the Client ID, Tenent ID as shown below:**

****

**For Secret ID go to Certificates & secrets in app registration where app is created as shown below:**

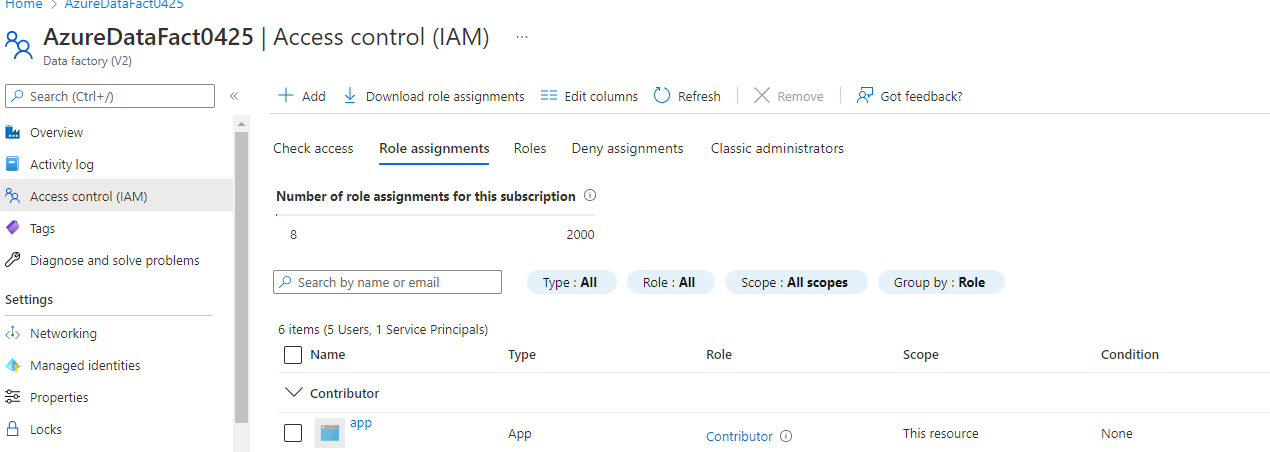
**Graphical user interface, text, application, email

Description automatically generated**

**Copy the value as Secret in Connection.**

**Next step is to add the app to access Data Factory we have add role assignment to Data Factory for the app as show below:**

**Azure Data Factory🡪IAM Roles🡪Add role Assignment🡪(Search the app and add)**

****

* **Use case:**
* **Trigger an ADF pipeline to copy the file from blob to the landing zone**
* **Trigger a databricks notebook on a job cluster.**
* **Pass the location and filename.**
* **Process the data and insert it into a delta table.**
* **On success or failure Send an alert email about the job.**

**Airflow Dag Graph View**

**Diagram

Description automatically generated**

1. Create a DAG with below code to run the Azure Data Factory pipeline.

We have two operators to use:

AzureDataFactoryRunPipelineOperator

Use the [**AzureDataFactoryRunPipelineOperator**](https://airflow.apache.org/docs/apache-airflow-providers-microsoft-azure/stable/_api/airflow/providers/microsoft/azure/operators/data_factory/index.html#airflow.providers.microsoft.azure.operators.data_factory.AzureDataFactoryRunPipelineOperator) to execute a pipeline within a data factory.

Passing parameter to the pipeline as name of raw and landing file name:

run\_pipeline1 = AzureDataFactoryRunPipelineOperator(

        task\_id="Adf\_run\_pipeline\_copyActivity",

        pipeline\_name="pipeline3",

        parameters={"raw\_filename": "testing.csv",

                    "landing\_filename":"testing\_landing.csv"

                    },

        do\_xcom\_push=True,

    )

Above Airflow operator Trigger copy Activity in Data Factory

Graphical user interface, application

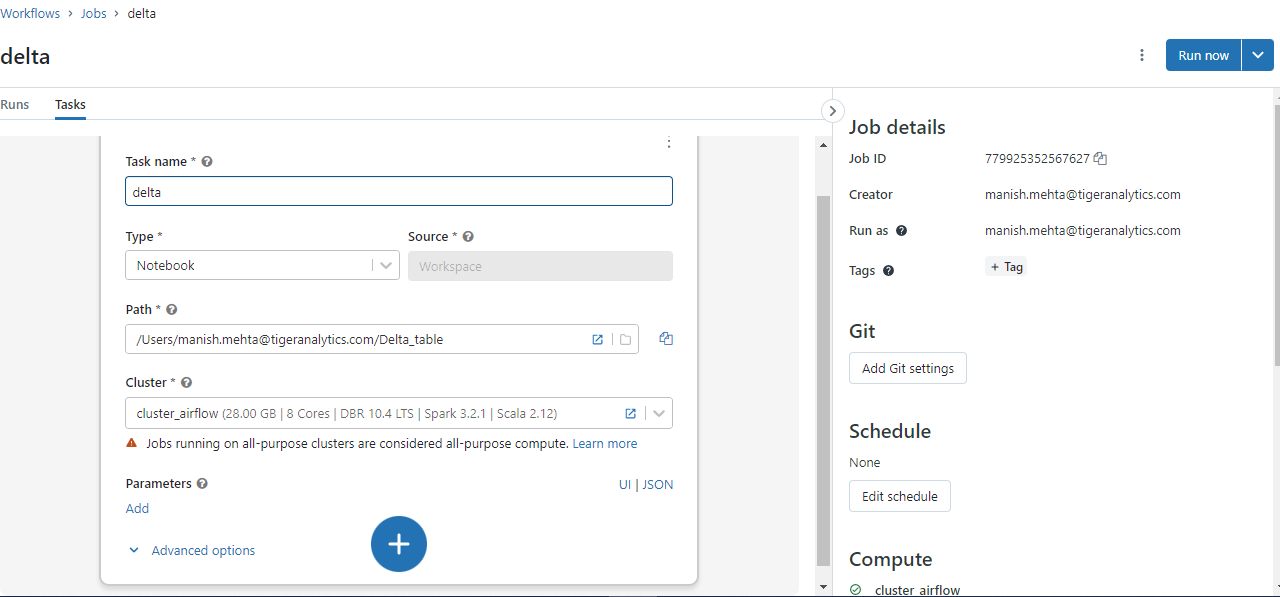
Description automatically generated

After copy from Blob Storage to landing Zone (Data Lake Gen 2):

2.Create a DAG with below code to run the notebook

We have operators to use

**DatabricksRunNowOperator** to run an existing job using the job id.



job\_id=779925352567627

notebook\_params = {

  "filename": "testing\_landing.csv"

}

notebook\_run = DatabricksRunNowOperator(

    job\_id=job\_id,

    task\_id="Databricks\_processing",

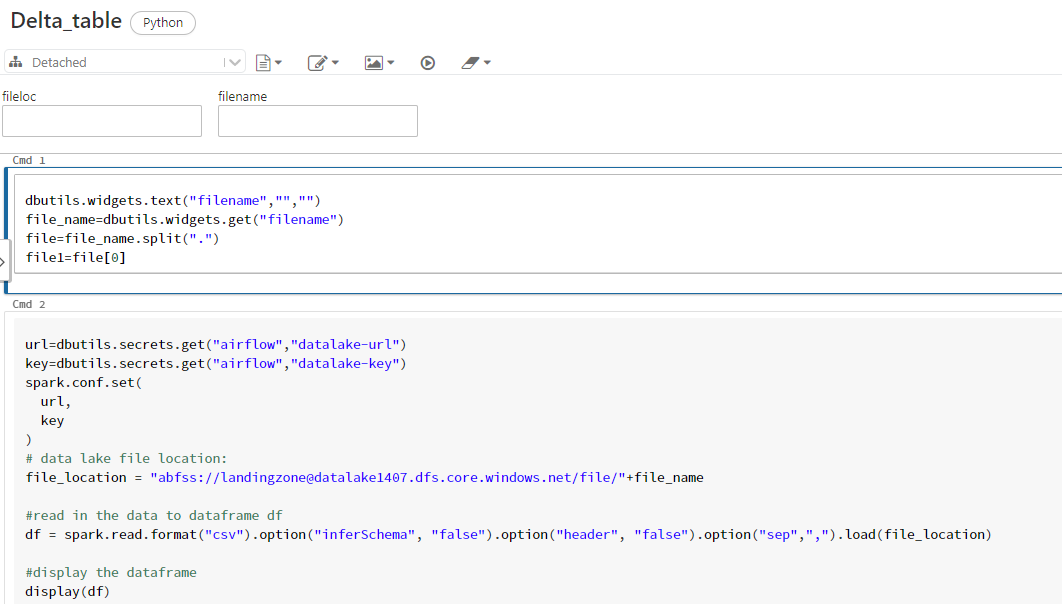
    notebook\_params=notebook\_params,

    dag=dag,

)

The above code is using DatabricksRunNowOperator is used to run the job in databricks

Job Notebook read the copied file from data lake (landing zone) and save as a table in databricks



3.Email Alert

EmailOperator is used to send email on job completion

send\_email = EmailOperator(

task\_id='email\_alert',

to='manishmehta5298@gmail.com',

subject='table created successfully',

html\_content="Date: {{ ds }} {{run\_id}}",

dag=dag,)

Graphical user interface, text, application, email

Description automatically generated

**Final Code**

import os

from datetime import datetime, timedelta

from airflow.models import DAG

try:

    from airflow.operators.empty import EmptyOperator

except ModuleNotFoundError:

    from airflow.operators.dummy import DummyOperator as EmptyOperator  # type: ignore

from airflow.providers.databricks.operators.databricks import DatabricksRunNowOperator

from airflow.operators.bash\_operator import BashOperator

from airflow.providers.microsoft.azure.operators.data\_factory import AzureDataFactoryRunPipelineOperator

from airflow.providers.microsoft.azure.sensors.data\_factory import AzureDataFactoryPipelineRunStatusSensor

from airflow.utils.edgemodifier import Label

from airflow.operators.email\_operator import EmailOperator

DAG\_ID = "AzureDatafactory\_Databricks\_run\_pipeline"

with DAG(

    dag\_id=DAG\_ID,

    start\_date=datetime(2022, 7, 6),

    schedule\_interval="@daily",

    catchup=False,

    default\_args={

        "retries": 1,

        "retry\_delay": timedelta(minutes=3),

        'email': ['manishmehta5298@gmail.com'],

        "azure\_data\_factory\_conn\_id": "Azure\_datafactory\_conn",

        "factory\_name": "AzureDataFact0425",  # This can also be specified in the ADF connection.

        "resource\_group\_name": "Azure\_Training",  # This can also be specified in the ADF connection.

    },

    default\_view="graph",

) as dag:

    begin = EmptyOperator(task\_id="begin")

    end = EmptyOperator(task\_id="end")

    # [START howto\_operator\_adf\_run\_pipeline]

    run\_pipeline1 = AzureDataFactoryRunPipelineOperator(

        task\_id="Adf\_run\_pipeline\_copyActivity",

        pipeline\_name="pipeline3",

        parameters={"raw\_filename": "testing.csv",

                    "landing\_filename":"testing\_landing.csv"

                    },

        do\_xcom\_push=True,

    )

job\_id=779925352567627

notebook\_params = {

  "filename": "testing\_landing.csv"

}

notebook\_run = DatabricksRunNowOperator(

    job\_id=job\_id,

    task\_id="Databricks\_processing",

    notebook\_params=notebook\_params,

    dag=dag,

)

send\_email = EmailOperator(

task\_id='email\_alert',

to='manishmehta5298@gmail.com',

subject='table created successfully',

html\_content="Date: {{ ds }} {{run\_id}}",

dag=dag,)

begin >> run\_pipeline1>>notebook\_run>> send\_email>>end