# ***Steps for Deploying Talend job into Snowflake using Airflow.***

1. **Setting up Snowflake Account:**
    - Created a new snowflake account:
    - Snowflake:
        Account: https://yab67224.us-east-1.snow*******/
        UserName: Khalidairflow
        Password: Prometric@2024
        warehouse= COMPUTE_WH
        database= AIRFLOW_SNF
        schema= AIRFLOW_SCHEMA
        Region: US East (N. Virginia)
        Role: ACCOUNTADMIN

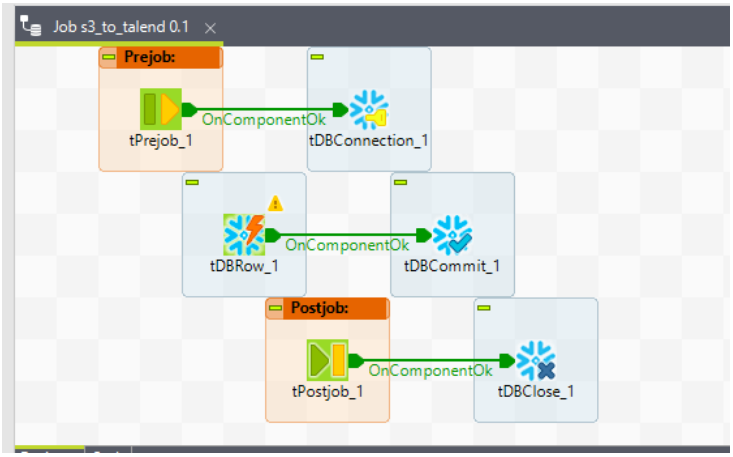
**Created Structure, Stage, and file format in Snowflake:**

```
create or replace TABLE AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE (
        ID NUMBER(38,0),
        NAME VARCHAR(16777216),
        COMPANY VARCHAR(16777216)
);

create or replace file format my_csv_format

type = csv field_delimiter = ',' skip_header = 1
field_optionally_enclosed_by = '"'
null_if = ('NULL', 'null')
empty_field_as_null = true;


CREATE OR REPLACE STAGE AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE_stage
URL = 's3://s3stag/landing_directory/'
CREDENTIALS = (
   AWS_KEY_ID = 'AKIATNIWVCQAYAQTXCPS',
   AWS_SECRET_KEY = 'INY4AN472h********************'
)
FILE_FORMAT = my_csv_format;

show stages;
list @AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE_stage;
select * from AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE;
```

2. **Setting up Talend Job**

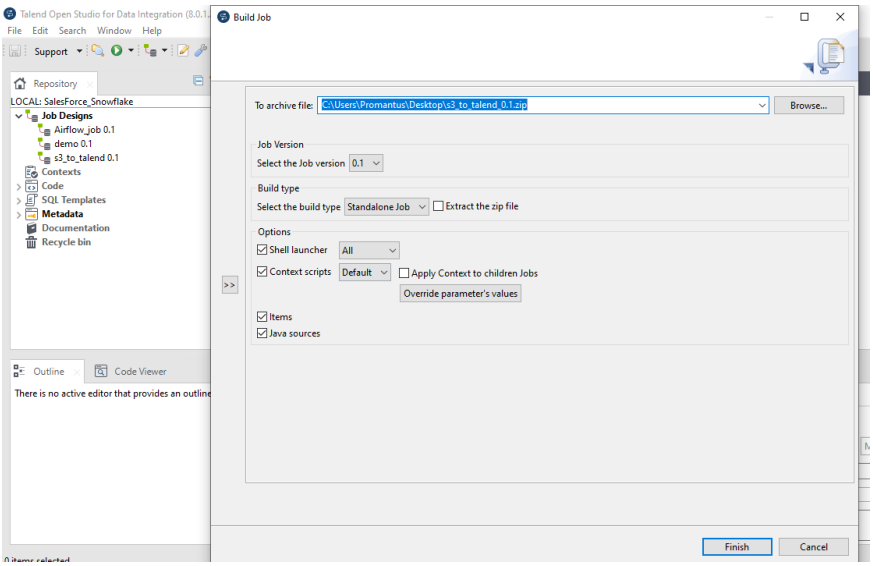
   **Develop Talend Job:**

   - Created a new talend job to deploy into snowflake as shown below.




   - In metadata set connection to snowflake using credentials.
   - tDBRow_1: Type below code to load data into snowflake.

     "copy into AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE
     From @AIRFLOW_SNF.AIRFLOW_SCHEMA.AIRFLOW_TABLE_stage
     FILE_FORMAT = (FORmaT_NAME =my_csv_format)"
   - Build standalone talend job as show in below image.

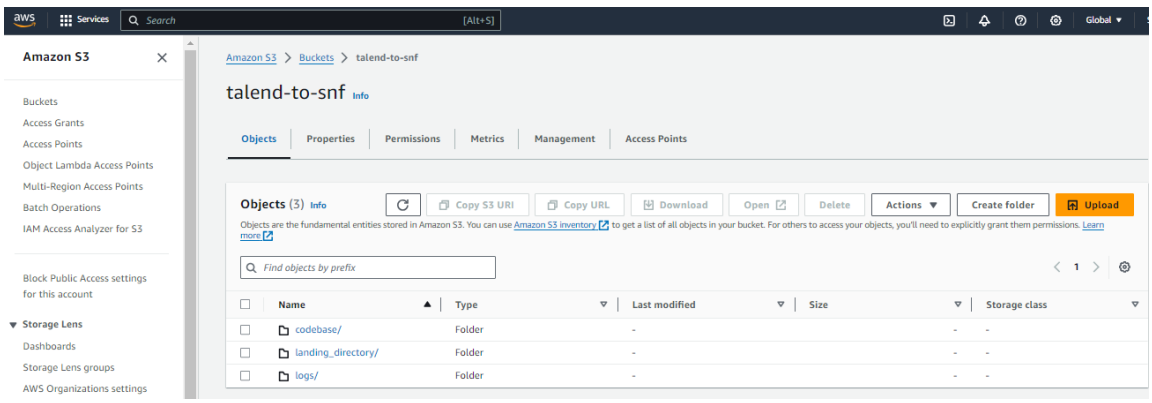



   - placed the job zip file in s3 bucket **talend-to-snf/codebase.**

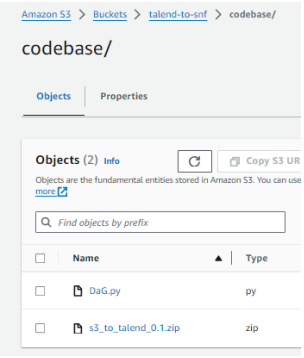
3. **Setting up Apache Airflow on AWS EC2:**

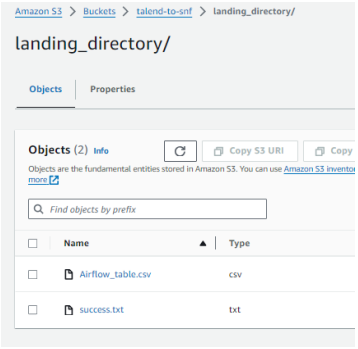
- Launch an EC2 instance on AWS with proper role and required packages to operate Talend, Python operator, snowflake, and airflow as follows.
  - Install openjdk-8-jdk since Talend requires.
  - Install unzip to extract build job-name.zip from Talend.
  - Install virtual machine to configure Airflow to use an external metadata database (e.g., PostgreSQL, MySQL)
  - Install Apache-airflow [Postgres]==2.6.0
  - Set user Id and password for login to airflow using public Ip address of ec2.
  - Install python3.
  - Install Apache-airflow-providers-snowflake. ...etc.
  - The details are attached.
- Create S3bucket give full accesses for the role you created while launching ec2.
  - Create log folder in S3 bucket: where you capture all logs of the packages.
  - Create codebase folder in S3 bucket: where you kept your DAG and job-name.zip.
  - Create landing directory folder in S3 bucket: where you need to put csv and success file.

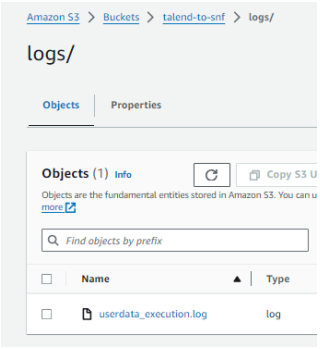
Creating S3 Bucket



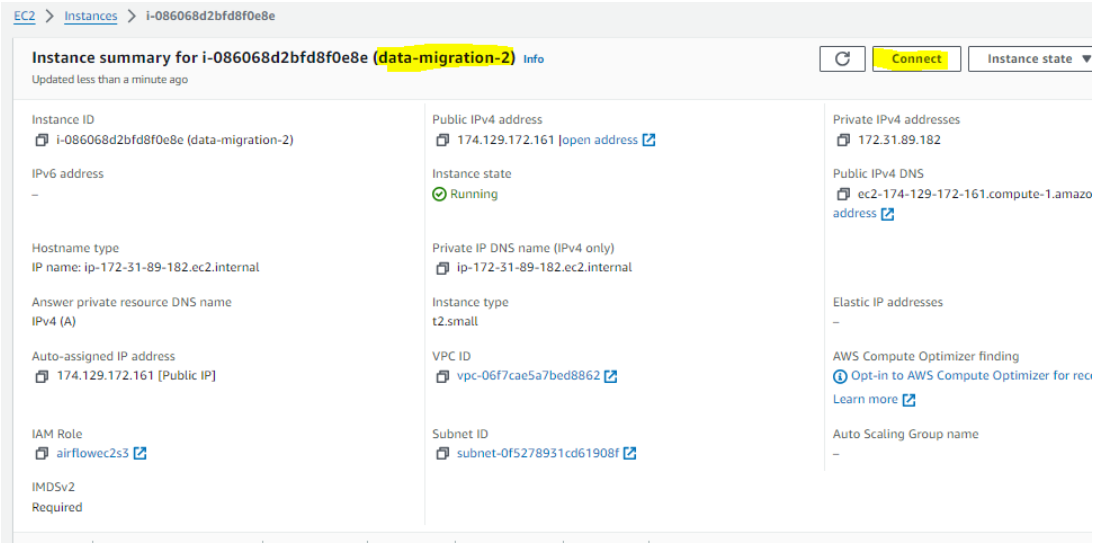

Codebase Landing directory log



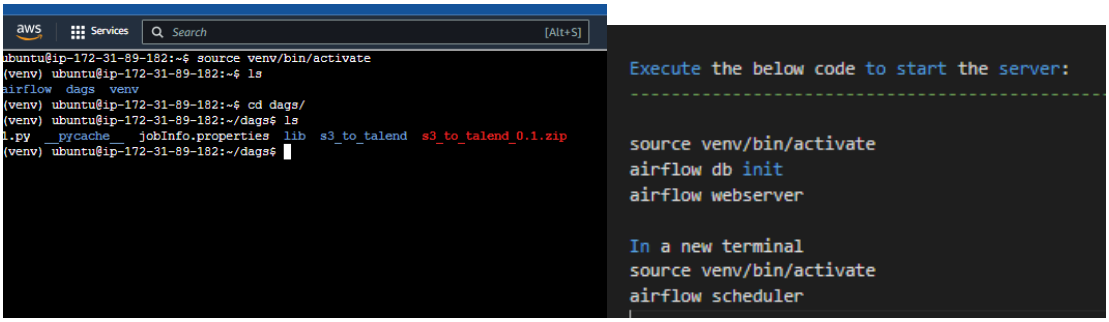



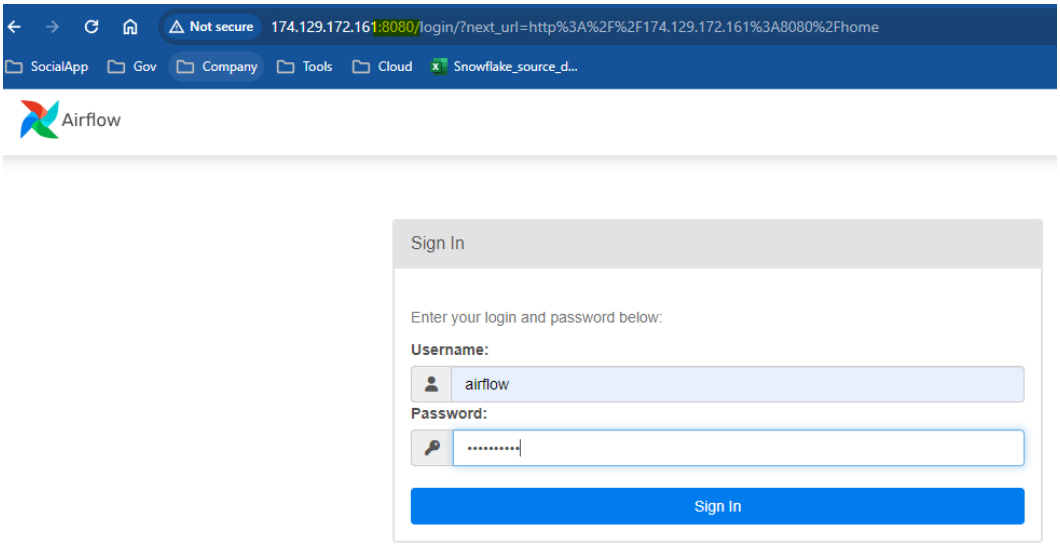

## Launch and Connect Instances




## Connect Terminal and execute the code to connect to airflow.

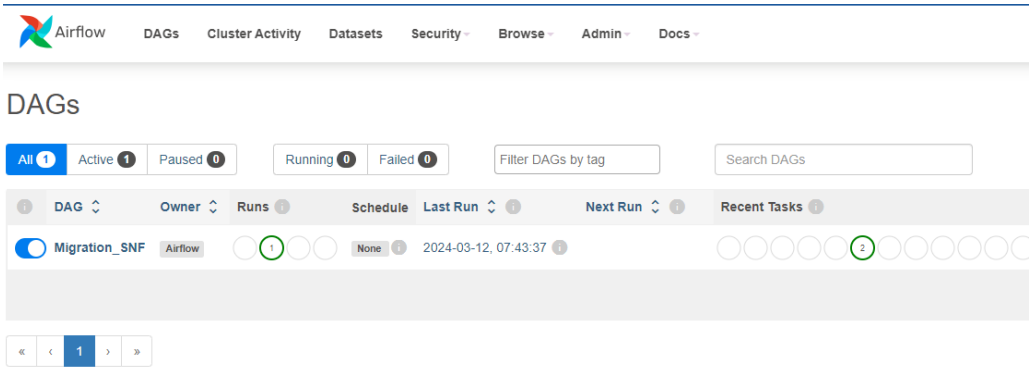



## Giving Public IP address of Ec2 follow by 8080 port of Airflow using Airflow user id and password which we define under packages Login to Airflow

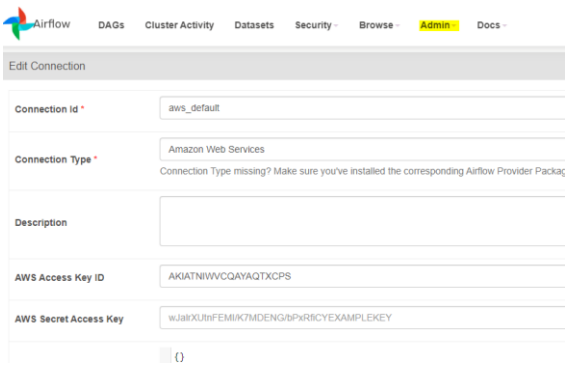



4. **Configuring Airflow Connections:**
    - Launch Airflow using Ip address of ec2.
    - Configure Snowflake connection in Airflow's Connection settings with the Snowflake credentials obtained earlier.
    - Configure other necessary connections for AWS (e.g., S3 staging)

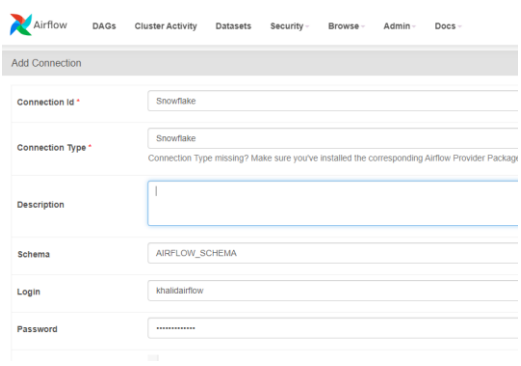
**In Airflow the Dag you loaded will be executed**




**Connect to AWS**




**Connect to Snowflake**



5. **Creating an Airflow DAG:**
    - Define an Airflow Directed Acyclic Graph (DAG) that includes tasks for executing your Talend job.
    - Use the Bash Operator or Python Operator to execute your Talend job as a task within the DAG.
    - Define any necessary dependencies between tasks.
6. **Start Airflow Schedular:** Ensure that the Airflow scheduler is running so that it can schedule and execute your DAGs according to the specified schedule.
7. **Monitor Execution:** Monitor the execution of your DAG in the Airflow UI. Check logs for any errors or issues that may arise during execution.
8. **Maintenance and Update:** Regularly maintain and update your Airflow environment, DAG scripts, and Talend jobs to ensure they continue to run smoothly and efficiently.