**Steps to Run the Project**

1. **Install Visual Studio Code**

Download and install [Visual Studio Code](https://code.visualstudio.com/).

1. **AWS IAM Roles** : Create the following IAM roles with appropriate policies:

- EC2\_DefaultRole: For EC2 to perform actions on EMR and S3.  
 - EMR\_EC2\_DefaultRole: For EC2 in EMR to access other AWS services.  
 - EMR\_DefaultRole: For EMR itself to perform actions on EC2 and other services.

- snowflake\_S3: For snowflake to assume this role to perform actions on S3. Note the ARN for this role.

## Create EC2 Instance

- Launch a `t3.large` EC2 instance and SSH into it from VS Code using a .pem key.

## 4. Install Dependencies

- Update and install Docker:  
  
sudo apt-get update -y  
sudo apt-get install docker.io -y  
sudo systemctl start docker  
  
- Install Astronomer:  
  
sudo apt-get install curl -y  
curl -sSL https://install.astronomer.io | sudo bash  
astro version

## 5. Initialize Airflow Project

- Create a directory and initialize the Astro project:  
  
mkdir airflow\_project && cd airflow\_project  
astro dev init

- Place Airflow.py in Dags/ folder

## 4. Prepare S3 Bucket

- Create the following directory structure in your S3 bucket and place the files accordingly:  
```  
bucket/  
├── input/  
│ └── Electric\_Vehicle\_Population\_Data.csv  
├── output/  
│ ├── electricutility\_details/  
│ ├── location\_details/  
│ ├── vehicle\_details/  
│ └── vehicletype\_details/  
└── scripts/  
 ├── src/  
 │ ├── snowflake.sql  
 │ └── transform.py  
 └── tests/  
 └── transform\_test.py  
```

## 5. Create Storage Integration Object in snowflake

- Create a Snowflake account and set up a storage integration object using the IAM role ARN in step 2

- Update the USER\_ID and EXTERNAL\_ID from storage object in the IAM role created for snowflake in step 2 to assume this role to perform actions on S3 bucket.

## 6. Start Airflow

-Build the Docker image and start the Airflow environment:  
  
astro dev start

## 7. Airflow Web Interface

- Access the Airflow web server at `http://localhost:8080`.

## 8. Set Up Connections

- In Airflow, navigate to Admin > Connections and create the following connections:

**1. AWS Default**

* + - * **Connection Id**: aws-default
      * **Connection Type**: Amazon Web Services
      * **AWS Access Key ID**: <Your AWS access key>
      * **AWS Secret Access Key**: <Your AWS secret key>
    1. **Snowflake**
       - **Connection Id**: snowflake\_conn
       - **Connection Type**: Snowflake
       - **Login**: <Snowflake account username>
       - **Password**: <Snowflake account password>
       - **Account**: <Your account ID>
       - **Warehouse**: <Warehouse name>
       - **Database**: <Database name>
       - **Region**: <Region name>
       - **Role**: <Type of Role>

## 9. Trigger the DAG

Trigger the Airflow DAG to execute the entire ETL process.