

### How to run this ETL:

- 1) run the Snowflake query to create the **stages** for S3 buckets with your S3 credentials
- 2) Put your credentials for S3 and snowflake in the python script named Dublin\_Transportation\_ETL.py (you can get this information in the amazon web portal related cloud services and the snowflake web portal)

S3:

```
99     aws_access_key = "xxxxxx"
100     aws_secret_key = "xxxxxxxx"
101     region = "us-east-2"
102     file_url = CSV_URL["Weather_Data_Met_Eireann"]
103     excel_csv_data = load_excel_from_url(EXCEL_URL["Cycle_Counts"])
104     bucket_name = "s3carloslinaresk81"
105     s3_file_name = f"Weather_Data_Met_Eireann_{formatted_datetime}.csv"
106     s3_excel_file_name = f"Cycle_Counts_{formatted_datetime}.csv"
107     folder_name = "s3_bucket_folder"
108     s3_excel_folder_name = "s3_excel_bucket_folder"
109     s3_folder_file_name = f"{folder_name}/{s3_file_name}"
110     s3_excel_folder_file_name = f"{s3_excel_folder_name}/{s3_excel_file_name}"
111
```

Snowflake:

```
22     SNOWFLAKE_CONFIG = {
23         "user": "xxxxxx",
24         "password": "xxxxx",
25         "account": "xxxx",
26         "warehouse": "COMPUTE_WH",
27         "database": "ETL_DATABASE",
28         "schema": "ETL_SCHEMA",
29         "role": "ACCOUNTADMIN"
30     }
```

- 3) Run the python script named Dublin\_Transportation\_ETL.py

Notes:

A) Access keys and secret key were used for this exercise for ease of use, but there are more secure methods that should be implemented on a prod, QA and develop environment

B) The daily run job in configure in Cron using this script:

Crontab -e to open a crontab in Linux

```

GNU nano 6.2 /tmp/crontab.noq2f8/crontab *
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow command
0 7 * * * /mnt/c/Users/Lenovo/Desktop/Portfolio/Scripts/python.exe /mnt/c/Users/Lenovo/AppData/Roaming/Jetbrains/pycharmCE2024.1/scratches/Dublin_Transportation_ETL.py

```

On the lines that are not commented are the path of your python.exe is specified followed by a space and then the path of the script to be executed.

C) A successful run of the python script looks like this:

```

File 'Weather_Data_Met_Eireann_2024_12_31_11_10_11.csv' uploaded to S3 bucket 's3carloslinaresk01/s3_bucket_folder/Weather_Data_Met_Eireann_2024_12_31_11_10_11.csv'
File 'Cycle_Counts_2024_12_31_11_10_11.csv' uploaded to S3 bucket 's3carloslinaresk01/s3_excel_bucket_folder/Cycle_Counts_2024_12_31_11_10_11.csv'
2024-12-31 11:10:22,594 - Data loaded from URL: https://us.cso.ie/public/api/restful/PxStat_Data_Cube_API_ReadDataset/10411/J0N-stat/1.0/en
2024-12-31 11:10:23,284 - Data loaded from URL: https://us.cso.ie/public/api/isonrc/dqra-576322/isonrc/572-5722_5722_5722method/572-5722PxStat_Data_Cube_API_ReadDataset/572-5722Params/572-5722Less/572-5722Query/572-5722d/572-5722
2024-12-31 11:10:23,944 - Data loaded from URL: https://data.smartbikes.ie/dataset/10411/J0N-stat/1.0/en
C:\Users\Lenovo\Desktop\ram-Interviews\lib\site-packages\snowflake.connector\config_manager.py:351: UserWarning: Bad owner or permissions on C:\Users\Lenovo\AppData\Local\snowflake\config.toml
  warn(f"Bad owner or permissions on {str(filep)} {chmod_message}")
2024-12-31 11:10:25,462 - Snowflake Connector for Python Version: 3.12.4, Python Version: 3.12.2, Platform: Windows-11-10.0.22H31-SP0
2024-12-31 11:10:25,462 - Connecting to GLOBAL Snowflake domain
2024-12-31 11:10:25,462 - This connection is in OCSP Fail Open Mode. TLS Certificates would be checked for validity and revocation status. Any other Certificate Revocation related exceptions or OCSP Responder failures would
2024-12-31 11:10:26,895 - Snowflake connection established.
2024-12-31 11:10:27,283 - Number of results in first chunk: 1
2024-12-31 11:10:27,283 - Table 'staging.Passenger_Numbers_dublin_bikes' created or already exists.
2024-12-31 11:10:27,375 - Number of results in first chunk: 1
2024-12-31 11:10:27,375 - Table 'final.Table.Passenger_Numbers_and_dublin_bikes' created or already exists.
2024-12-31 11:10:27,938 - Number of results in first chunk: 1
2024-12-31 11:10:27,938 - Table 'staging.Weather_Data_Met_Eireann' created or already exists.
2024-12-31 11:10:28,078 - Number of results in first chunk: 1
2024-12-31 11:10:28,078 - Table 'final_table.Weather_Data_Met_Eireann' created or already exists.
2024-12-31 11:10:28,420 - Number of results in first chunk: 1
2024-12-31 11:10:28,420 - Table 'staging.Cycle_Counts' created or already exists.
2024-12-31 11:10:28,527 - Number of results in first chunk: 1
2024-12-31 11:10:28,527 - Table 'final_table.Cycle.Counter_Totem' created or already exists.
2024-12-31 11:10:28,788 - Number of results in first chunk: 1
2024-12-31 11:10:28,788 - Table 'final_table.Cycle.Counter_Eco' created or already exists.
2024-12-31 11:10:30,908 - Data from bus.Passenger.Numbers inserted into the staging table.
2024-12-31 11:10:32,783 - Data from Dublin_Bus.Passenger.Numbers inserted into the staging table.
2024-12-31 11:10:33,228 - Data from Dublin_Bikes inserted into the staging table.
2024-12-31 11:10:33,228 - Copy csv into staging done.
2024-12-31 11:10:34,817 - Number of results in first chunk: 1
2024-12-31 11:10:34,817 - Copy converted excel to csv into staging done.
2024-12-31 11:10:37,438 - Number of results in first chunk: 1
2024-12-31 11:10:38,278 - json Data moved from staging to final table.
2024-12-31 11:10:39,100 - CSV Data moved from staging to final table.
2024-12-31 11:10:42,294 - Number of results in first chunk: 1
2024-12-31 11:10:42,294 - ETL process completed successfully.
2024-12-29 23:15:42,294 - closed
Dublin_Transportation_ETL.py
1:1 CRLF UTF-8 4 spaces Python 3.12 (ram-Interviews)
2024-12-29 23:15:51,817 - ETL process completed successfully.
2024-12-29 23:15:51,817 - closed
2024-12-29 23:15:51,915 - No async queries seem to be running, deleting session
2024-12-29 23:15:52,022 - Snowflake connection closed.

Process finished with exit code 0

```

D) Json files were loaded from the URL without using S3 because it's not a heavy load. On the other hand, the .CSV are a much heavier load so that's why S3 was used with the copy into command that's optimized for these scenarios. Finally, the excel file needed to be converted to .csv because snowflake works better with csv instead of .xlsx format files.

