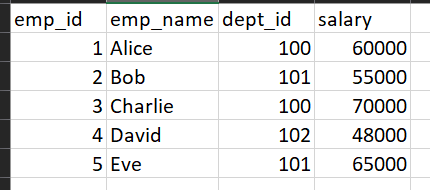
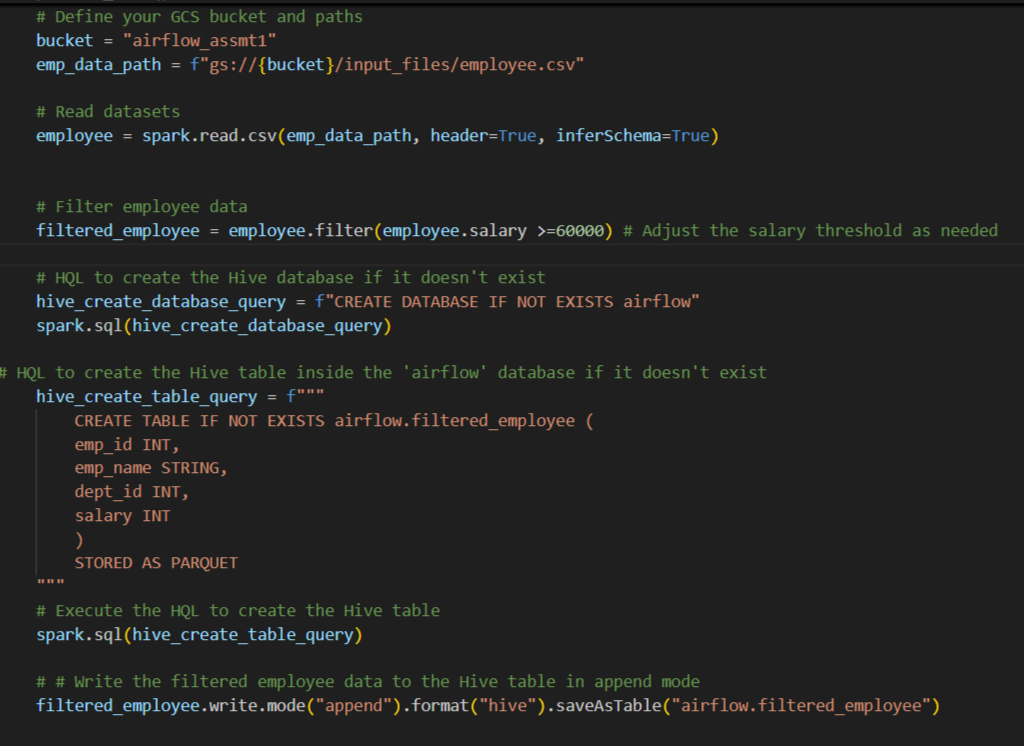
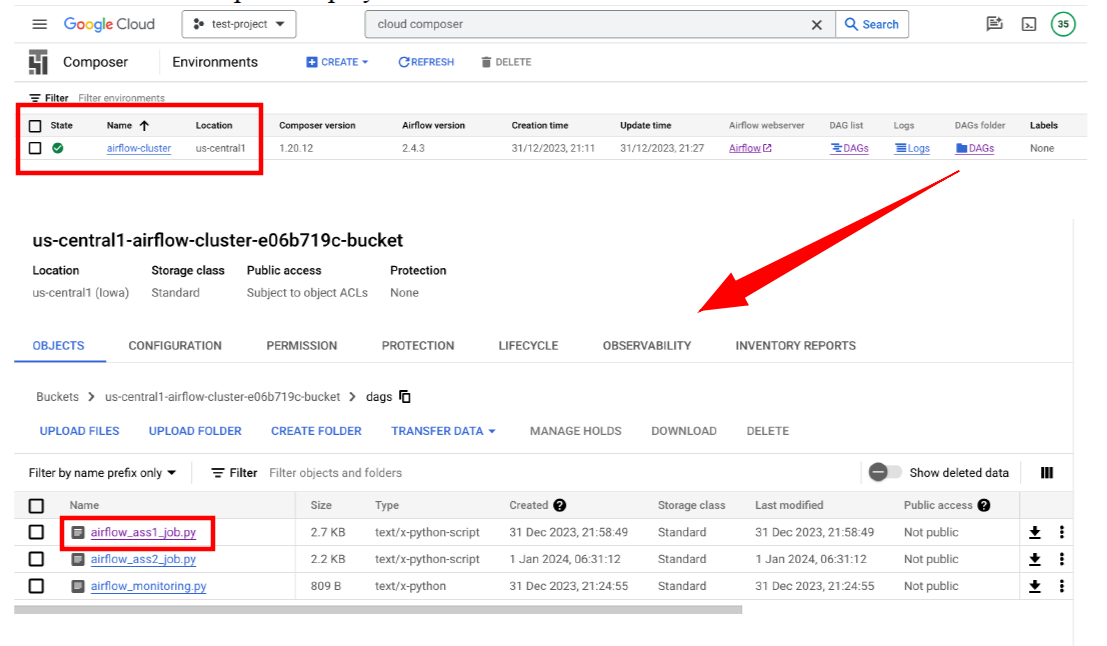
Employees Batch Data Processing

1. automated a workflow using Apache Airflow to process daily incoming CSV files from a GCP bucket using a Dataproc PySpark job and saving the transformed data into a Hive table.
2. Used GCP services- DataProc/GCS and Airflow to schedule the dag job.
3. created a bucket called ‘airflow\_assmt1’ and placed employee.csv under input\_files folder. This file will be picked up by the spark job which is a part of the DAG. Also made sure to place the pyspark job ‘employee\_batch.py’ in the python\_file folder in the same GCS bucket.

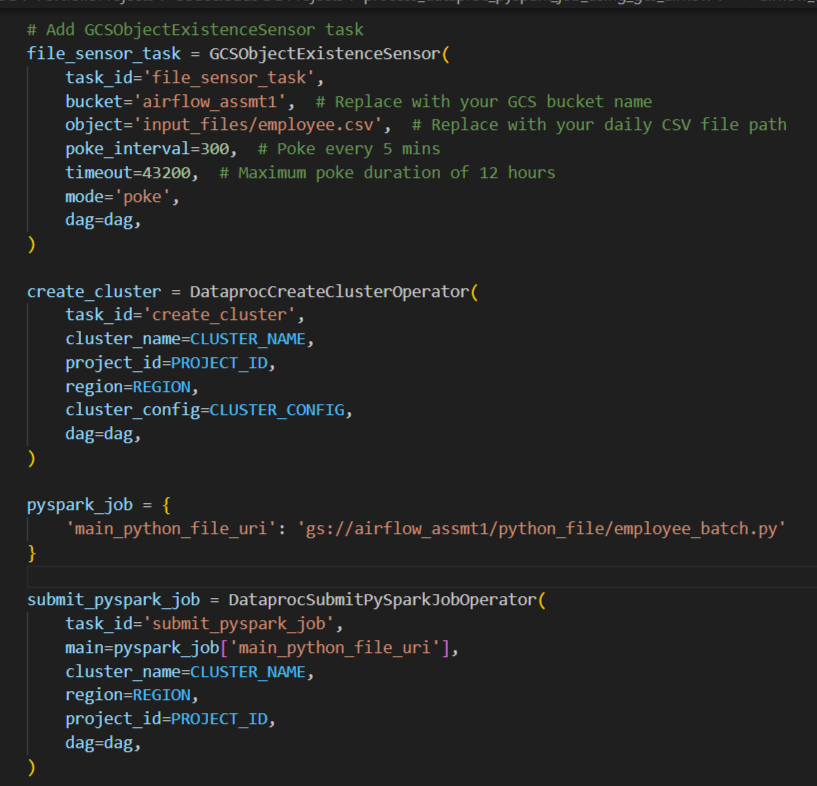




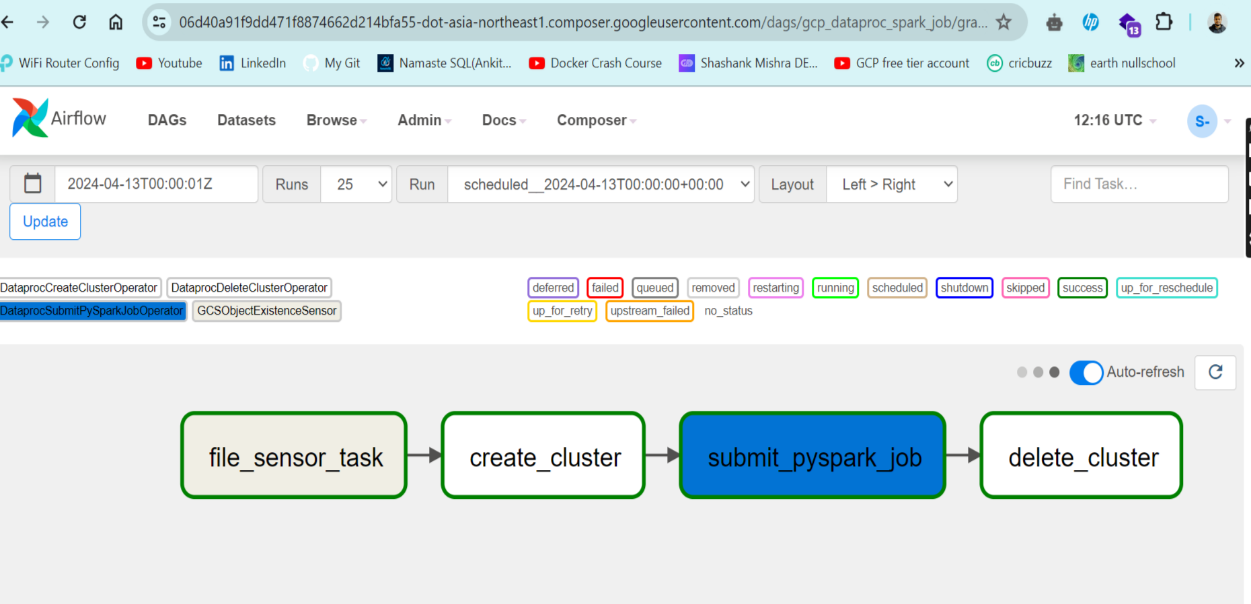
1. I created an airflow cluster and then proceeded to place the ‘airflow\_ass1\_job.py’ file in the DAG list so that it can be picked up by airflow.



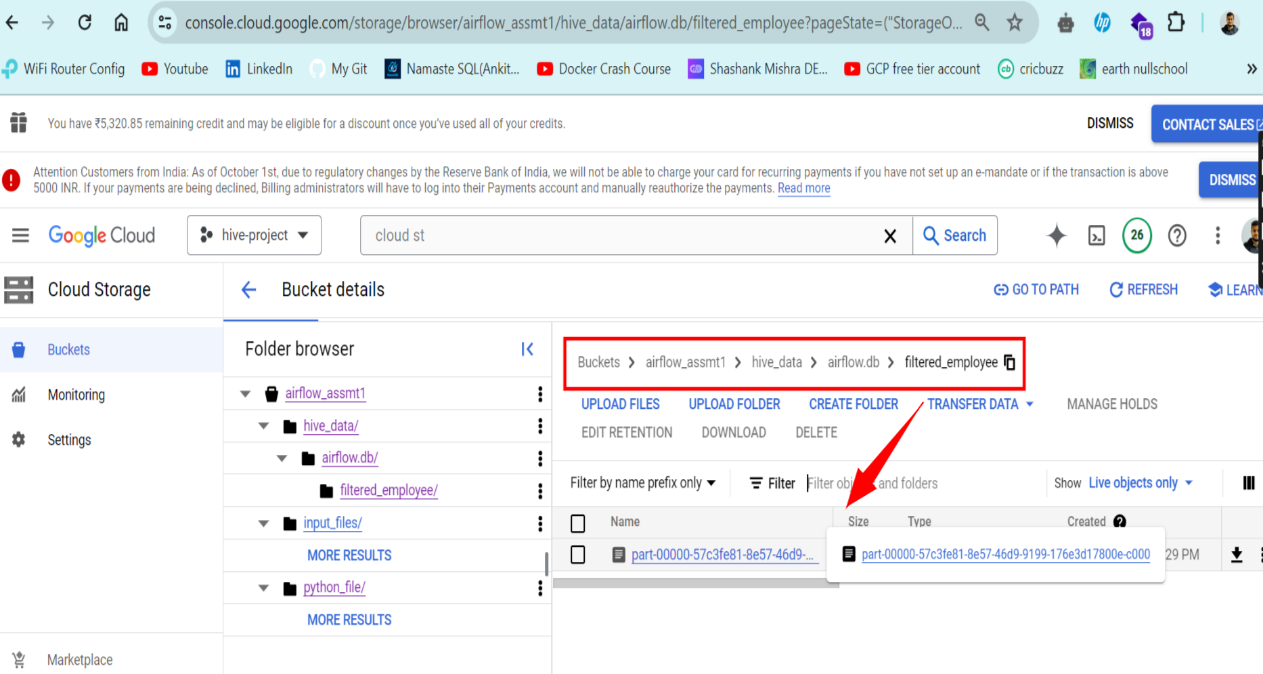
1. Now the various stages can be seen in the DAG. A file sensor checks every 5 mins in the input\_file location, once it detects employee.csv a new cluster was created followed by which the spark job was launched which then filters employees with salary >=60,000 and then placing the output in another GCS location in the bucket airflow\_assmt1 under the hive\_data folder.



1. Then we were able to see that all stages have run successfully followed by deletion of the cluster as well.



1. We can see the resultant data saved in a parquet file format saved in hive\_data folder (over which we were able to build external hive tables to query the data)



Challenges:

1. Issues with the spark job being picked up since we had to define a location to save the output.

2. We can’t put the output on the local of the cluster since with the deletion phase of the cluster this data would disappear as well. Hence it made sense to place the output in a GCS bucket