**Assumptions and Solution Notes**

1. **For this weather data processing and answering the questions, I am using aws glue for etl / data pipeline**

Assuming few of the following:

1. Assuming the following s3 bucket & folder have been created before running the script to process the data

**S3 Bucket :** s3://ma-weather-data-bucket/

**Folder for CSV file(s) :** s3://ma-weather-data-bucket/raw\_csv/

**Folder for Parquet file(s) :** s3://ma-weather-data-bucket/parquet/

**Note:** The above things are configured in the script at the beginning. This configuration can also be put as a separate python script and can be imported in the main script. So if there is any change to these can be easily changed at one place in the configuration and whole thing is good to go.

1. Using the Redshift to create a table to load the weather data and query the same out of it to answer the 3 questions.
2. **The data pipeline process flow follows the standard ETL process / stages**

**Stage-1 (Extract)**

This step is meant for Data Extraction.

In this step the given CSV files are read from their configured path on s3 and loaded into a spark dataframe

**Stage-2 (Transform)**

This step is meant for Data Processing.

In this step the data loaded in the spark dataframe in the previous step is cleaned up. In this scenario, the data rows with NULL values in the 3 columns of subject (to answer the questions) are discarded and also the data is converted into parquet and written back to the configured path on s3.

**Stage-3 (Load)**

This step is meant for Data Loading.

In this step the data is converted into parquet format and written back to the configured path on s3.

**Note:** There is also one more part of Data Loading, loading the data into a database table to query to answer the 3 questions. The Data Loading into the Redshift table is done outside of this script with a purpose to show the other method of data loading. An extra module can be added to automate the data loading into a database table.

1. **The Redshift table creation, queries (to answer the 3 questions) and the query outputs are documented in a separate document.**