ETL WITH AIRFLOW

Problem Description

A music streaming service requires an end-to-end data pipeline to analyze user streaming behavior. The data pipeline should integrate data from multiple sources, process it, and generate key performance indicators (KPIs) for business intelligence.

The user and song metadata reside in an **Amazon RDS** database and should be simulated using provided CSV datasets. Streaming data is stored in **Amazon S3** in batch files. The pipeline should extract, validate, transform, and load the data into **Amazon Redshift** for analytical processing.

Objective:

- Build a data pipeline that ingests user and song metadata from RDS (CSV simulation) and streaming data from S3.
- Perform necessary transformations and validations.
- Compute KPIs such as:
  - **Genre-Level KPIs:** Metrics that provide insights into how different music genres perform on the platform.
    - **Listen Count:** Total number of times tracks in a genre have been played.
    - **Average Track Duration:** The mean duration of all tracks within a genre.
    - **Popularity Index:** A computed score based on play counts, likes, and shares for tracks in a genre.
    - **Most Popular Track per Genre:** The track with the highest engagement (plays, likes, or shares) in each genre.
  - **Hourly KPIs:** Metrics that capture platform activity trends on an hourly basis.
    - **Unique Listeners:** The distinct number of users streaming music in a given hour.
    - **Top Artists per Hour:** The most streamed artists during each hour.
    - **Track Diversity Index:** A measure of how varied the tracks played in an hour are, based on the number of unique tracks played compared to total plays.
- Load processed data into **Amazon Redshift** for further analysis.

User Stories:

1. As a **data engineer**, I want to ingest metadata from RDS (CSV) and streaming data from S3 into a processing pipeline.
2. As a **data engineer**, I want to validate that all required columns exist in each dataset to ensure data integrity.
3. As a **data engineer**, I want to transform raw streaming data and compute meaningful KPIs for business analysis.

4. As a **data engineer**, I want to efficiently load transformed data into Redshift using an **Upsert** strategy to handle duplicate and new records.
5. As a **business analyst**, I want to query the processed data in Redshift to gain insights into user behavior and song popularity trends.

Deliverables:

1. **ETL Pipeline Implementation:**
   - An **Apache Airflow DAG** that orchestrates the data pipeline.
   - Python scripts for data extraction, transformation, validation, and Redshift ingestion.
2. **Data Validation Module:**
   - Automated checks to ensure all required columns exist before processing.
3. **Transformation & KPI Computation:**
   - Code to compute genre-based and hourly streaming KPIs.
4. **Redshift Data Loading Module:**
   - Optimized **Upsert** strategy using staging tables to merge data efficiently.
5. **Logging & Error Handling:**
   - Detailed logging and error handling to troubleshoot pipeline failures.
6. **Documentation:**
   - Step-by-step documentation on setting up and running the pipeline.
   - SQL queries to validate results in Redshift.

Evaluation Criteria:

- Correct implementation of **ETL pipeline** using Airflow.
- Proper **data validation** and error handling.
- Efficient computation of **KPIs** and storage in Redshift.
- Code **readability, efficiency, and best practices**.
- Well-structured documentation for ease of use and troubleshooting.