Batch Data Processing for Rental Marketplace Analytics

Project Overview: A rental marketplace platform (similar to Airbnb) requires an end-to-end data pipeline to enable analytical reporting on rental listings and user interactions. The platform stores application data in an AWS Aurora MySQL database, and the goal is to build a data warehouse using Amazon Redshift for business intelligence and reporting.

You are required to implement a batch data processing pipeline that:

- Extracts rental listing data from AWS Aurora MySQL and loads it into Amazon S3 as an intermediate storage layer.
- Ingests data from S3 into Amazon Redshift for structured processing.
- Implements a multi-layer architecture in Redshift with Raw, Curated, and Presentation layers.
- Uses AWS Glue for extraction, transformation, and loading (ETL).
- Orchestrates the workflow using AWS Step Functions to ensure efficient execution.

Key Business Metrics to Derive:

Once the data is available in Redshift's Presentation Layer, generate the following insights:

Rental Performance Metrics:

- **Average Listing Price:** Compute the average price of active rental listings each week.
- **Occupancy Rate:** Measure the percentage of available rental nights that were booked over a month.
- **Most Popular Locations:** Identify the most frequently booked cities every week.
- **Top Performing Listings:** Track properties with the highest confirmed revenue per week.

User Engagement Metrics:

- **Total Bookings per User:** Count the total number of rentals booked per user every week.
- **Average Booking Duration:** Compute the mean duration of confirmed stays over time.
- **Repeat Customer Rate:** Measure how many users book more than once within a rolling **30-day period**.

Evaluation Criteria

- Efficient ETL implementation using AWS Glue & Step Functions.
- Correct data validation and transformation logic.
- Optimized Redshift schema for analytical queries.
- Well-documented setup and troubleshooting guide.

1. apartment_attributes Table

Column Name	Description
id	Unique identifier for each apartment listing.
category	Type of apartment (e.g., Studio, 1BHK, 2BHK, 3BHK, Penthouse).
body	Description or details of the apartment.
amenities	List of available amenities in the apartment (e.g., Wi-Fi, Gym).
bathrooms	Number of bathrooms in the apartment.
bedrooms	Number of bedrooms in the apartment.
fee	Additional fee associated with the apartment (e.g., maintenance).
has_photo	Boolean indicating whether the apartment has photos available.
pets_allowed	Boolean indicating if pets are allowed in the apartment.
square_feet	The area of the apartment in square feet.
address	Full address of the apartment.
cityname	Name of the city where the apartment is located.
state	Name of the state where the apartment is located.
latitude	Latitude coordinate of the apartment's location.
longitude	Longitude coordinate of the apartment's location.

2. user_viewing Table

Column Name	Description
user_id	Unique identifier for each user.
apartment_id	Reference to the apartment being viewed (links to apartment_attributes).
viewed_at	Timestamp of when the apartment was viewed by the user.
is_wishlisted	Boolean indicating if the apartment was added to the wishlist.
call_to_action	Action prompt for the user (e.g., Contact, Book Now, Save for Later).

---

3. apartments Table

Column Name	Description
id	Unique identifier for each apartment listing.
title	Title of the apartment listing.
source	Platform or source where the apartment listing is posted (e.g., Airbnb).
price	Price of the apartment listing.
currency	Currency in which the price is listed (e.g., USD, EUR).
listing_created_on	Timestamp when the apartment listing was created.
is_active	Boolean indicating if the apartment listing is active.
last_modified_timestamp	Timestamp of the last update to the apartment listing.

4. bookings Table

Column Name	Description
booking_id	Unique identifier for each booking.
user_id	User ID who made the booking (references user_viewing).
apartment_id	Apartment ID for the booked apartment (references apartments).
booking_date	Timestamp of when the booking was made.
checkin_date	Start date of the stay for the booking.
checkout_date	End date of the stay for the booking.
total_price	Total amount paid for the booking.
currency	Currency used for the payment (e.g., USD, EUR).
booking_status	Status of the booking (e.g., confirmed, canceled, pending).
payment_status	Payment completion status (e.g., paid, pending, failed).
num_guests	Number of guests for the booking.