Build a Data Mart in SQL(DLBDSPBDM01)

Phase 3: Finalisation

Student Name: Akawa Johannes Sheepo

Matriculation Number: 92131324

Introduction

The modern hospitality industry heavily relies on data-centric systems to streamline property rental processes and enhance user experiences. This project simulates the development of a relational database inspired by Airbnb, with the objective to store, manage, and process structured data for hosts, guests, properties, bookings, payments, and communications. The system enables efficient interactions among platform participants and supports data integrity, transparency, and business operations like earnings tracking, reviews, and availability.

Database Functionality Overview

The developed database offers full-featured management capabilities for three primary user roles: **Guests**, **Hosts**, and **Administrators**. Each role performs defined operations aligned with real-world booking platforms.

- **Guests** can register, browse listings, filter results, book rooms, communicate with hosts, post reviews, and make payments.
- **Hosts** manage property listings, set availability, define room rules, and monitor earnings and guest feedback.
- **Administrators** oversee transactions, resolve disputes, ensure compliance, and generate platform-wide reports.

The database is normalized to the 3rd Normal Form (3NF) to reduce redundancy and ensure referential integrity. It supports complex queries for operations such as filtering available rooms, calculating host income, and logging user verifications and messages.

Schema Design and Implementation

The **Entity-Relationship (ER) model** is composed of over 20 entities, including:

- user, guest, host, listing, room, photo, address, amenity, listing_amenity, roomrule, roomrule_assignment, booking, payment, transaction, rating, review, calendar, message, verification, and sociallink.

The schema includes:

- **Many-to-many relationships** (e.g., listing_amenity, roomrule_assignment)
- **Recursive relationships** (e.g., message for threaded communication)
- **Triple relationships** (e.g., transaction joins booking, user, and role context)

SQL scripts were written for the creation and population of each table, and rigorous testing was conducted via structured queries and result validation.

Metadata and System Statistics

The database stores extensive metadata to monitor and evaluate system usage:

Metadata Item	Value
Total Number of Tables	20
Average Entries per Table	~25
Total Records	~525 entries
Estimated DB Volume	~3.6 MB (SQL dump, populated)
Number of Relationships	30+ (including junction tables and FK constraints)

Each table was seeded with dummy data ranging from 20 to 30 entries per entity, ensuring the ability to test realistic use-cases, such as overlapping bookings, host-specific listings, and refund triggers.

Key Features Implemented

- **Calendar-driven availability management** for rooms using calendar.
- **Secure payment recording** with detailed tracking of payment method and status.
- **Guest-host communication and review system** through message, rating, and review.
- **Compliance tracking and user trust enhancement** via verification and sociallink.

All SQL commands (DDL and DML), screenshots of database content, and result outputs have been compiled and documented in the development phase presentation.

Conclusion

This project delivers a robust, scalable relational database solution that reflects the functional and data requirements of an Airbnb-like platform. It incorporates core database principles and real-world business logic, ensuring both functionality and maintainability. The final product demonstrates competency in ER modeling, SQL-based implementation, and data architecture suited for real-world deployment.